# 基于数学建模和大数据思想的

# 新冠病毒防控和检测的研究

【摘要】

本文以本次新冠疫情实际情况为背景，着眼于在抗疫物质情况紧张且瘟疫传播途径不明朗的情况下，大规模疫情的防治和潜在感染者的检测，并结合数学建模和大数据的知识对抗疫工作提供建议。

通过模拟本次疫情的真实情况，总结抗议过程中有效的隔离和群组检验措施，并利用数学建模的思想加以改进和分析，有助于在未来的大规模疫情中提供帮助，在医疗物资困难的条件下分析患者的特征，最大化的利用物资。

群组检验的研究针对传染病病毒传播期间所面对的检测问题。常规检测是对疑似患者进行逐一检测，其检测效率低，检测成本高，很难实现应检即检的目的，使得一些患者没有得到应有的治疗，甚至在社会上造成疫情传播。本文提出的检测方案，是将被检测对象多轮分组进行检测。分组检测法是对接受检测人员提取样本，然后将每组被检人员的全部样本放到一起统一进行一次检测。如果是阴性，则表明该组被检人员均不是传染病病毒患者，全部通过检测。如果是阳性，则表明该组被检人员中有传染病病毒患者，然后根据该组人员数量分别考虑，如果人数较多，就进一步分组检测，如果人数较少，就对每一位进行逐一的检测，以确定其中哪一位，或者哪些人是传染病病毒患者。分组检测方式适合于被检人员多，患病比例低的情况，可以显著减少检测次数，提高检测效率，降低检测成本。

此外，我们还考虑了隔离措施。感染 Covid-19 的人可能在几天或更长日子之后才出现症状，有些人更是从来都不会显示症状，但他们仍然可以传播病毒。因此我们还准备聚类分析技术分析无症状感染者的表观特征，借此为隔离提供建议。

【关键字】

**概率统计原理 灰色关联模型 传染病模型浓度模型**

### 目录

### 1. 绪论

#### 研究背景

近些年来，全球范围内多次有强传染性的疾病爆发，而这些疾病不只是具有强传染性，有些还具有强致命性。我国境内也多次爆发过不同类型的强传染性疾病，例如 2003 年爆发的非典（即重症急性呼吸综合征），2004 年的禽流感疾病，2009 年的甲型 H1N1 流感,2016 年的 ZIKA 病毒疫情,除此之外，还有一些传染性较小的疾病，但是对社会的稳定也有较大的影响，例如在社会人群中持续传播并无法治愈的艾滋病。从社会角度来看，如果能快速且准确的在样本群中检测出患病的个体，则能在疾病的传播初期，对疾病的传播进行有效的控制，对社会的稳定和经济的发展都有重要的意义；从医学疾病检测机构角度来看，如果能快速且准确的在样本群中检测出患病的个体，能有效节约检测经费和检测时间，提高检测效率。

新冠肺炎疫情席卷全球，对全球经济和各国人民正常生活造成严重影响。为了让世界回到正轨，缓解全球新冠肺炎疫情，检测是最重要的部分。因此，检验问题已成为我们必须直接面对的困难和挑战。

检测问题面临的最大困难是检测量有限。由于这部分受医疗水平的影响，医疗水平在短期内难以显著提高，因此有必要在短期内改进模型，让有限的检测实现对大多数人的最大覆盖。

因此，我们面临的问题是，面对大量的检验对象，我们如何将他们分组，提高检验量的覆盖面和效率?如何实施我们的检测方法来应对疫情的各种情况?无症状感染者能否得到有效控制?针对这些问题，我们展开了研究和调查。

#### 研究意义

分组检测思想经过几十年的发展，已经在国内外的研究中得到广泛的应用，不仅仅局限于对传染性疾病梅毒的检测，也应用于 HIV 病毒筛查和收集血清样本进行某种疾病筛查的医学检测领域、马铃薯病毒检测的植物病理学领域和药物研究的新药发现领域等诸多领域。根据不同的检测情况，分组检测思想对应的算法，可估计检测出样本组内特异性个体的检测次数，分组检测的最优组大小，或者是简单的将特异性个体从样本组中检测出来

采用多次分组检测方法，就是将疑似病例人员根据要求进行分组进行检测。对接受检测人员提取样本，然后将每组被检人员的全部样本放到一起进行检测，我们称之为混样检测。如果是阴性，则表明该组被检人员均不是传染病病毒患者，全部通过检测。如果是阳性，则表明该组被检人员中有传染病病毒患者，然后就对该组人员分别进行逐一的检测，以确定其中哪一位，或者那些人是传染病病毒 患者。因为学生患有传染病病毒 的比例很低，所以，分组检测后，检测结果呈阳性的比例很低，而检测结果呈阴性的比例很高。也就是说，分组检测后，大部分被检人员在第一次全组检测后就通过了检测，除了是否是传染病病毒患者的疑虑。由于每组被检人员较多，而只通过一次检测，就有明确的结果，检测效率高，检测成本也很低。

### 2. 同类研究工作现状

Dorfman 分组检测就是将所需检测的待测样本汇聚到一个样本池，对样本池进行检测，把检测结果为阳性的样本池里的样本分割成一个个的小样本组，然后对每一个样本组进行检测，若不考虑检测试剂的敏感性和特异性，当组检测为阳性，则需要对组内的个体进行逐个检测，当组检测为阴性，则表明组内无特异性个体，检测停止,这种检测思想称为 Dorfman 检测。通过实验证明和在实际案例中的不断验证，均表明了对实验样本进行分组检测的检测效率明显优于对试验样本进行单体检测的检测效率。

自从分组检测思想提出后，很多学者便对如何进一步提高检测效率这一问题进行了更多的研究，Andrew Sterrett 在 Dorfman 检测思想的基础上提出了 Sterrett 检测思想，他认为与其检测阳性组中的每个样本，不如先进行单体检测，，当检测出第一个阳性个体，把剩余样本视为一个新的待检测组进行一次组检测，如果组检测为阴性，则检测工作已经完成，如果结果为阳性，就继续上一过程，直到发现所有阳性个体，因为应用算法进行检测的样本组是由小患病概率个体组成的，所以样本组内检测出患病个体的可能性较小，经模拟研究，发现 Sterrett 算法的检测效率高于 Dorfman 分组算法的检测效率。Christopher R 在文中提出了一种在考虑协变量信息时，识别积极个体的改进算法，该协变量信息用于构建如何在阳性组中实施重新测试，这种改进算法称为“信息性重新测试“，推导了概率质量的闭式表达式和实现算法函数用于解码阳性组所需的测试次数的函数，应用于内布拉斯加州不孕症预防项目的衣原体和淋病检测，研究结果表明，信息性再测试可以显著减少测试的数量，同时提供与已建立的测试相似的准确性。

在目前的疾病检测算法研究中，研究人员不仅仅关注如何提高对样本的检测效率，如何用降低检测成本也是分组检测研究过程中的一个重要部分，在这方面也有很多研究成果。E.J. McMahon 在“汇集献血者样本以降低 HIV-1 抗体检测的成本”一文中描述了一种用于在 15 个样本池中检测 HIV-1 抗体的改良酶免疫分析(EIA)方法，对于检测弱血清阳性样本和检测 HIV-1 蛋白质印迹阴性或蛋白质印迹不确定结果与按照标准环评检测单个样本一样敏感，通过对献血的检测，减少人类免疫缺陷病毒(HIV-1)传播，降低了检测成本,Behets.F等人为了确定 HIV-1 检测前汇集血清的准确性和成本效率，对 8000 名工人及配偶的血清进行了单独筛查，并在 800 个血清池中对每个血清池进行了 10 次筛查，对阳性血清池中所有血清的进一步检测可以确定艾滋病毒阳性个体，将不同条件下将池的成本效率与单个测试进行比较，结果表明，当血清效价和获取样本的成本足够低时，汇集检测是一种可供发展中国家和工业化国家替代单体检测进行使用的方案，因为对血清池进行测试与单独测试相比，能节省 78%的检测成本。

而在对样本检测时，可能存在检测误差，尤其在分组较多的情况，检测次数也会增加，产生误差的可能性也随之增大，这样会导致最后检测结果偏差较大，并且浪费检测资源。通过较少检测次数来减少检测误差，则需对样本进行最优分组，徐明民和邵均利在“分组验血算法的数学证明及最佳分组人数的一个新公式”一文中，对分组验血法作了比较严密的数学证明，提出了一个计算最佳分组人数的新公式，由公式计算可计算出不同患病概率所对应的最佳分组人数，这样能使得对样本组的检测次数降低，进而降低检测误差。而产生误差的客观原因方面，主要是检测试剂的敏感性和特异性的精确度是无法达到100%，所以提高检测试剂的敏感度能大大减少客观误差的产生。Diwakar Gupta 和 ReginaMalina 修改了多尔夫曼和斯特雷特的小组检测方案，使其适用于检测血液样本中是否存在艾滋病毒抗体，建议通过改变反应阈值和非反应阈值，将试剂盒灵敏度提高到接近 100%。随后，使用组和重复测试，仔细选择组的大小和重复测试的次数，以最大限度地提高效率，同时保持假阳性预测值(FPPV)在一个特定的范围内。

### 3.符号说明

|  |  |
| --- | --- |
| 符号 | 定义 |
| N | 组大小 |
| a | 每一组的数目 |
|  | 感染率 |
| b | 感染人数 |
| X | 测试的总数 |
| c | 有受感染的人群 |
|  | 假阳性率 |
|  | 假阴性率 |
|  | 实验仪器的误差概率 |
|  | 相关系数 |
| Z | 综合方法的要点 |
| r(t) | 总人口数是负的 |
|  | 每个无症状人群的概率 |
| A | 每天可以测试的样品数量 |
| M | 总人数 |
|  | 被检测为阳性的概率 |
| S | 无症状人群 |
|  | 每组的人数 |
|  | 每个人的测试次数 |
|  | 检测结果呈阳性的概率 |
|  | 测试结果为阴性的概率 |
|  | 剩下的人数 |
|  | 检测总数 |

\*注：表中未出现的部分符号以文中具体说明为准。

### 研究目标和基本方法

#### 4.1 主要目标

本文以本次新冠疫情实际情况为背景，着眼于在抗疫物质情况紧张且瘟疫传播途径不明朗的情况下，大规模疫情的防治和潜在感染者的检测，并结合数学建模和大数据的知识对抗疫工作提供建议。通过模拟本次疫情的真实情况，总结抗议过程中有效的隔离和群组检验措施，并利用数学建模的思想加以改进和分析，有助于在未来的大规模疫情中提供帮助，在医疗物资困难的条件下分析患者的特征，最大化的利用物资。

#### 4.2 基本方法

1. 采用基本的分组方法，确定分组阶段数和每个阶段的群组大小。决策有关的目标是：测试总数， 假阳性率和假阴性率。通过修改假阳性率和假阴性率和感染率，进行灵敏度分析。
2. 改变分组方案，确定分组阶段数和每个阶段的群组大小。通过修改假阳性率和假阴性率和感染率，进行灵敏度分析。将不同的分组方案的结果进行对比。给出建议。
3. 面对未知的感染情况和感染率，动态的分析和调整方案，给出一套综合的分组测试建议。以此为基础，开发一款处理测试结果数据并给出下一步提示的软件程序。
4. 利用机器学习的技术分析病患和潜在患者群体的表观特征，给出合理的隔离建议，编入软件。
5. 构建一种科学的评价体系，为检测工作的效率和准确性进行评价。

#### 4.3 主要方案

1. 对于分组检验，我们有二分分组方案，多层分组方案和立方分组方案等必选方案；
2. 对于隔离，我们给出了积极隔离，消极隔离等方案。

#### 4.4 拟解决的关键问题

1. 在抗疫物质情况紧张且瘟疫传播途径不明朗的情况下，大规模疫情的防治和潜在感染者高效检测；
2. 在抗疫物质情况紧张且瘟疫传播途径不明朗的情况下，以检测结果为基础的隔离方案；

### 分组测试模型

#### 5.1问题分析

为了提高同等资源下的资源利用效率，我们考虑采用群检测方法。问题1要求我们对测试的总数、假阳性率和假阴性率做出决定，建立数学模型，并根据每个阶段的阶段数和组数进行初步计算和检验。[1]

首先考虑决策目标。为了更快地进行冠状病毒的检测，必须考虑检测总数、假阳性率、假阴性率三个因素。对于检测总次数来说，检测总次数越少，可以节省的资源就越少，在资源有限的前提下可以检测更多的人。关于假阳性率，即阴性被判定为阳性的百分比。由于分组的样本是阳性的，重组的所有成员都被确定为阳性，并进行进一步的测试，直到确定一个阳性的个体。假阳性率越低，可以隔离人数越少，实现资源的有效利用。对于假阴性率，即阳性被判定为阴性的百分比，即对于实验仪器的误差，减少测试次数可以控制实验的误差。

其次，考虑两个变量——阶段的数量和每个阶段的群体规模。阶段数要结合实际情况考虑。我们首先查阅相关信息，得到日测试阶段的上限作为相位变量个数的取值范围。对于每个阶段的分组大小，应对整体样本进行分组检验。因为每个样品都是统一的，所以是不可分割的。为了计算方便，我们将试验样本的因素考虑为组数。

第三，利用统计概率样本检测相关知识，利用总体样本的特征估计部分样本。我们使用香港的诊断率来估计样本中病人的数量。由于患者的数量将随机分布在每组样本中，所以患者的数量将随机分布在不同数量的组中，以确定总检验次数，所以我们使用期望来表示平均检验次数。采用随机抽样的原理，通过多次试验，试验总数稳定。

最后，对这三个决策目标进行了综合评价。由于三个决策目标之间存在联系，我们采用灰色关联法来获得综合指标。通过调整阶段数、组数和组内样本数，减少综合评价指标，最终得到最优的组检测方案。[3]

#### 5.2模型设计

**5.2.1 变量分析**

**步数(步)**

官方资料显示，一次核酸检测检测样本的时间为6小时，而医院诊所的正常工作时间为10小时。假设疫情期间医院实行轮班制，我们有理由认为，我们认为一天内可以执行的检测周期为两次，即阶段数取值范围为1-2。由于阶段数等于1意味着所有的样本将被测试一次，所以我们主要考虑阶段数为2的情况。

**组的大小(N)**

每个阶段的分组规模(N)是指该阶段需要检验的样本总数。我们想把整个样品分组进行测试。考虑到每个样本是一个整体，不能被整除，为了计算方便，我们只考虑组数是样本总数的因子的情况。式中a为第二阶段的分组数，n为第二阶段的分组数，其关系如下:

**5.2.2三个决策目标**

**检测时间**

检测次数的计算步骤如下:

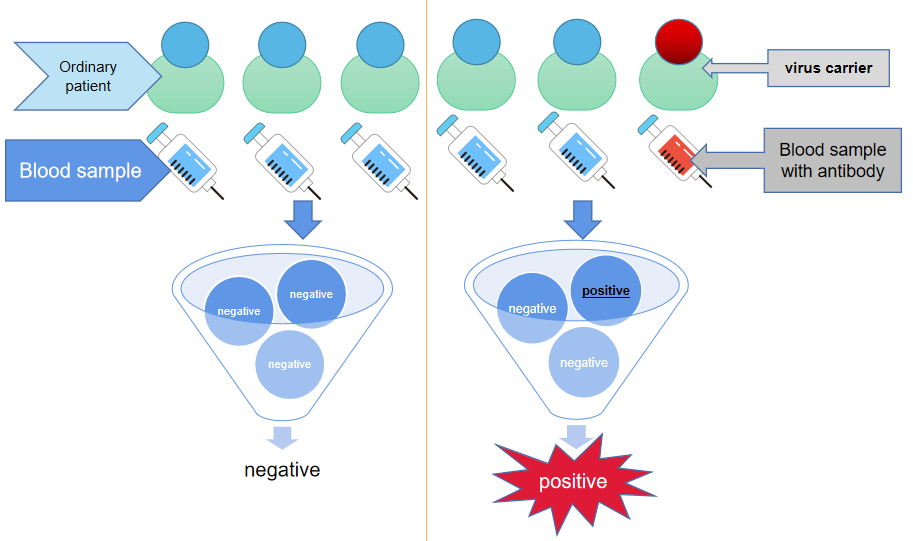
**1.总样本中感染人数的确定**

由于样本中受感染人数的不同，将受感染人数随机分配到不同组的概率也会不同，从而导致测试总数的不同。设感染率为，感染人数为:

**2.考虑阶段数为2的试验总数**

考虑阶段数为2的情况。第一阶段,首先将样本总数在第一阶段分成组,一个是n的因素因此,检测的数量在第一阶段是a。由于感染者的数量是随机分配到不同的组,每组,只要有一个感染样本组中,这意味着这群人被感染了随机分布示例图如下:

图1：随机分配模型



然后我们讨论了感染样本分组后的分组数。如果b组感染人数大于a组感染人数，则最多为一组阳性;如果b组感染人数少于a组，则b组最多为正。[2]

假设a组和c组有感染者，试验总次数的计算公式如下:

（3)

由于c的不同，检测的数量也不同，将被感染的人数随机分配到不同数量的组中。从问题的含义来看，一个群体中受感染的人数表示该群体的所有成员都被感染。下一步是检查单个样本是否被感染。因此，我们只需要考虑感染群体的数量，所以我们参考造假的概率。假币概率的原理是，一堆硬币中有一些是假币。**假币的质量比真币的质量低。**这堆硬币是随机分组的。其中，有一批伪币，其质量比所有真币都要差。假币的数量有一定的概率组。使用概率找到预期的检测数表示不同组的平均检测数。预期公式如下:[4]

（4）

因此，我们必须找到期望最小的分组方法作为阶段2的最优分组检测方案。

**假阳性率**

对于假阳性率，即判断阴性为阳性的百分比，由于分组的样本是阳性的，因此重组的所有成员都是阳性的，并进一步检测，直到确定一个阳性的个体。[1]假阳性率越低，可以减少的隔离人数就越少。设假阳性率为，计算公式为:

(5）

**假阴性率**

关于假阴性率，即被判定为阴性的阳性的百分比，即实验仪器的误差。[3]设实验仪器的误差概率为，假阴性率的计算公式为:

(6）

**3.综合评价模型**

从决策目标的角度，我们选择检测总数、假阳性率、假阴性率作为指标来评价群体检测方法的优劣。由于各指标之间相互联系，又具有反映不同方面的性质，因此必须建立科学合理的综合评价模型。我们选择灰色关联模型。具体模型如下:

确定总体样本量。不同组的检测总数、假阳性率、假阴性率不同，如下表所示:

|  |  |  |  |
| --- | --- | --- | --- |
| 组的数量 | 考验的时刻 | 假阳性率 | 假阴性率 |
|  |  |  |  |
| … | … | … | … |
|  |  |  |  |

图2:确定的值

**选择父索引**

在过程2中，我们找到了关于测试总数、假阳性率和假阴性率的公式。我们可以发现假阴性率和假阳性率与总检测数有关，而总检测数最直观地反映了群体检测的利弊，所以选择总检测数的父指标。

**计算相关系数**

定义父指标与第1点的相关系数为:

(7）

其中可计算为:

(8）

a和b的公式为:

（9)

(10）

取值范围是，然后取。

**计算关联程度**

(11）

**建立综合评价模型**

根据上一步得到的相关性，计算各指标的权重:

（12）

作为各指标的权重，得到一个综合指标。

(13）

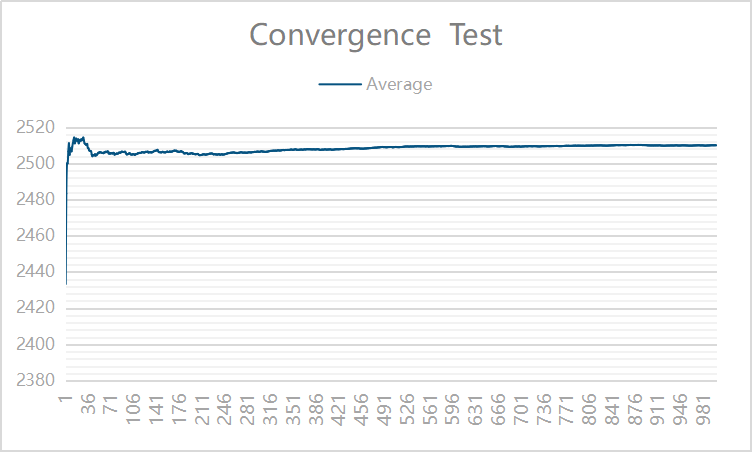
最后，综合指标最小对应的分组数为总体样本数N最好的分组方法。

#### 4.3模型的解决方案

关于感染率的值，我们利用整体样本的特征来估计部分样本，也就是用香港确诊率来估计样本中的患者人数。根据国家统计局的数据，目前香港有7万人感染，其中香港总人口为7345 670人。感染率是千分之一，所以我们取a =0.001。

在计算测试次数时，由于分组后一组阳性的概率很容易计算，只要组中有阳性的个人，就意味着该组是阳性的。计算有多少组是正样本的概率太复杂了，但它们必须存在，所以我们选择一个大的随机模拟得到的概率和期望。下面我们展示了随机时间和期望的图像:

图3



从图像上看，当随机数达到350次时，检测次数的期望值基本趋于一致，在一个很小的范围内波动。为了减少计算误差，我们在计算期望检测次数时取10000的随机数。

利用模型中的假阳性率和假阴性率公式计算不同组下的值。以样本总数为20000为例，不同分组情况下的数据值如下表所示:

|  |  |  |  |
| --- | --- | --- | --- |
| 受感染的人数 | 考验的时刻 | 假阳性率 | 假阴性率 |
| 20. | 1251.763979 | 0.031328 | 0.000285 |
| 40 | 1764.176201 | 0.042819 | 0.000396 |
| One hundred. | 2772.505343 | 0.065685 | 0.000624 |
| 200 | 3891.00172 | 0.08849 | 0.00087 |

图4:不同分组情况下的数据值

根据灰色关联模型，三个指标与父指标的相关性如下表所示:

|  |  |  |  |
| --- | --- | --- | --- |
| 受感染的人数 | 考验的时刻 | 假阳性率 | 假阴性率 |
| 相关 | 0.999999999 | 0.910954924 | 0.910954837 |

图5:三个指标和父指标

各指标权重如下表所示:

|  |  |  |  |
| --- | --- | --- | --- |
| 受感染的人数 | 考验的时刻 | 假阳性率 | 假阴性率 |
| 重量 | 0.745446 | 0.139456 | 0.115098 |

图6:各指标权重

综合评价指标为:

(14）

综合评价指标值越小，分组方案越有效。

各组计算的综合指标如下表所示:

|  |  |  |
| --- | --- | --- |
| 组的数量 | 综合指标 | 排名 |
| 1020 | 0.972138929 | 1 |
| 1024 | 0.97210998 | 2 |
| 1071 | 0.972096827 | 3. |

图7:各组计算的综合指标

得到种群为20,000的最优组为200组。

重复上述步骤，得到不同样本数情况下的最优分组方案。结果如下表所示:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 总检测 | 患病率 | 的患者数量 | 数量的组 | 时间的测试 | 假阳性率 | 假阴性率 | 分数 |
| 4000 | 20. | ０．００５ | 271.2 | 553.954888 | 0.065689 | 0.000624 | 0.054968158 |
| 4000 | 40 | 0．01 | 379.8 | 777.215848 | 0.089354 | 0.000877 | 0.070881357 |
| 10000 | 20. | 0．002 | 439.8 | 881.818882 | 0.042202 | 0.00039 | 0.037706762 |
| 10000 | One hundred. | 0．01 | 935.6 | 1944.595555 | 0.0909 | 0.000891 | 0.071062632 |
| 20000 | 20. | ０．００１ | 605.2 | 1251.763979 | 0.031328 | 0.000285 | 0.027861071 |
| 20000 | 40 | 0．002 | 867.8 | 1764.176201 | 0.042819 | 0.000396 | 0.03767999 |
| 40000 | 40 | ０．００１ | 1212.5 | 2503.446565 | 0.031328 | 0.000285 | 0.02438544 |

图8:不同样本的结果

**4.4灵敏度分析**

在上述模型中，我们使用香港的诊断率作为检验样本中的患病率，但患病率会因地区而异。在流行率高的地区，流行率将会上升;在安全地区，患病率相对较小。为了应对不同区域的检测，我们调整感染率，得到不同感染率对应的最优分组策略。下面我们对4000、10000和20000的整体样本进行了敏感性分析

下表显示了4000人总检测人数和不同感染率对应的分组策略:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 总检测 | 患病率 | 的患者数量 | 数量的组 | 时间的测试 | 假阳性率 | 假阴性率 | 分数 |
| 4000 | 4 | ０．００１ | 122.6 | 250.35529 | 0.030939 | 0.000282 | 0.01545349 |
| 4000 | 10 | 0.0025 | 198 | 393.948366 | 0.046487 | 0.000433 | 0.031818349 |
| 4000 | 20. | ０．００５ | 271.2 | 553.954888 | 0.065689 | 0.000624 | 0.054968158 |
| 4000 | 40 | 0．01 | 379.8 | 777.215848 | 0.089354 | 0.000877 | 0.070881357 |

图9:总人数和不同的感染率

当测试总人数为10000时，结果如下:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 总检测 | 患病率 | 的患者数量 | 数量的组 | 时间的测试 | 假阳性率 | 假阴性率 | 分数 |
| 10000 | 20. | 0．002 | 439.8 | 881.818882 | 0.042202 | 0.00039 | 0.037706762 |
| 10000 | 50 | ０．００５ | 683.2 | 1385.714887 | 0.065251 | 0.00062 | 0.054973315 |
| 10000 | One hundred. | 0．01 | 935.6 | 1944.595555 | 0.0909 | 0.000891 | 0.071062632 |

图10:总人数和不同的感染率

### 6.识别模型

#### 6.1问题分析

第二个问题要求我们检测无症状感染者，建立数学模型，找到合理的方法减少人群中无症状感染者的数量。

首先，要确定减少无症状感染者对人群影响的方法。通过查阅文献，有两种方法可以减少无症状感染对人群的影响。一是开展大规模核酸检测，寻找无症状人群。中国新型冠状病毒携带者;另一个是限制未接受检查的居民旅行的政策。

其次，为了整合上述两种解决方案，本文考虑将整个人群划分为无症状人群，确定病毒携带者和不携带病毒的健康人，基于传染病模型中的SIR模型，建立了无症状感染的处理模型，该模型依赖于三个种群之间的转换关系。改变转换关系中的参数，模拟两个方案对无症状人群的控制和治疗，并以无症状人群数量处于小状态时的时间长短作为方案质量的标准。

最后，两种方案的参数变化率分别为30%和60%。参数变化比越大，对程序的影响越大。因此，比较不同的项目优势在实施前后的效果，并解释我们应该如何处理这种效果。无症状感染者持续蔓延。

#### 6.2问题模型

通过查阅文献，有两种方法可以减少无症状感染对人群的影响:[5]

**1.开展大规模核酸检测，在无症状人群中寻找新型冠状病毒携带者。**

**2.对未接受检查的居民实行旅行限制政策。**

**5.2.1基于sir的无症状传播模型**

在建立模型之前，我们首先确定人群类型和无症状人群接受核酸检测的概率。具体讨论如下:[8]

**人群类型**

在流行病的传播中，不同类型的人对流行病的传播有不同类型的影响。通过查阅资料，我们发现该模型可以与SIR模型进行比较，SIR模型将人群分为易感人群、感染人群和治愈人群。同样，我们将总体分为三部分:S, I, R，具体含义如下:

|  |  |  |
| --- | --- | --- |
| 象征 | 意义 | 缩写 |
| 年代 | 无症状感染者(感染者、健康人) | 无症状的 |
| 我 | 被诊断为阳性的人 | 积极的 |
| R | 被诊断为阴性的人 | 负 |

图11:具体含义

负总体占总总体的比例为r(t)，初值记为，。由于总人口保持不变，三种人口的比例明显有以下公式:

(15）

**无症状人群进行核酸检测的概率**

政府进行大规模核酸检测时，各无症状人群被检测的概率为:[6]

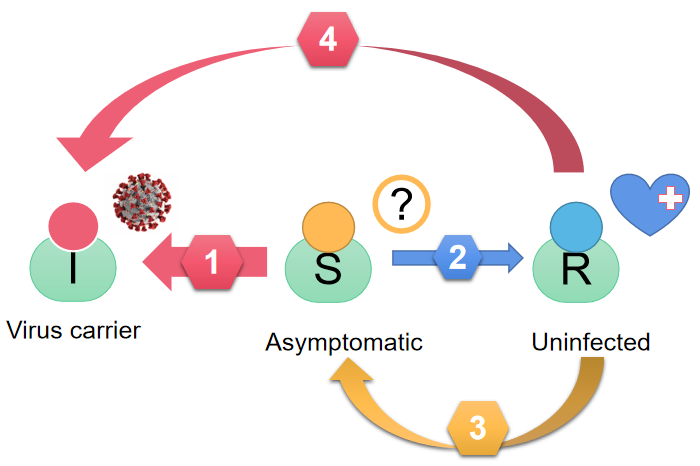
(16）

其中A为每天可检测的样本数量，M为总人数。换句话说，上述公式的概率就是可检测样本的数量占无症状总体的比例。如果检测量扩大，意味着A增加，每个无症状样本被检测的概率更高。

**5.2.2建立无症状传播模型**

首先考虑无症状总体S、正总体I和负总体r之间的关系，通过分析得到三者之间的转换关系。转换关系示意图如下:

图12



1.S试验后无症状阳性者变为I型阳性者。

2.无症状人群经S检验诊断为阴性，R检验为阴性。

3.阴性人群I易被未感染人群S中的病毒携带者感染，但并不表明症状变成无症状人群S。

4.阴性人群I易被非感染人群S中的病毒携带者感染，出现症状就医后转变为阳性人群S。

其次，分析这四种变换关系，得到微分方程。具体步骤如下:

考虑情况1、4，根据概率检验阳性人群的增加来自无症状人群S，结果为阳性，阴性人群I感染转为阳性人群。假设被检测为阳性的概率为，感染症状的概率为上述信息，得到的变化率为:

（17）

从情况1、2来看，对于无症状人群来说，无症状人群会分别减少正、负人群的流入。设检验为负的概率为，则得到的变化率为:

（18）

式中，简化变化率为:

（19）

从2、3、4的情况来看，对于阴性人群来说，增加的人数来自于检测后呈阴性的无症状人群。人数的减少是在两个方面。一种是被感染的人，当他们有症状时变成阳性。同理，变化率是

(20）

在上述讨论的基础上，得到了基于SIR的无症状传播模型

(21）

**5.3无症状传播模型求解**

利用Matlab对模型进行求解，通过文献数据对模型中的参数赋值，赋值结果如下:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | 价值 | 参数 | 价值 | 参数 | 价值 |
|  | ０．６ | 米 | 7000000 |  | 0.99 |
|  | 0．1 | 一个 | 10000 |  | 0．63 |
|  | 0．3 |  | 0．01 |  | 0.03 |

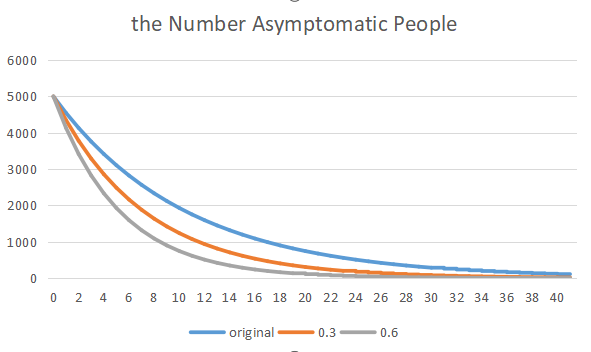
图13:分配结果

以上两种干预方法是调整参数A和参数。关于参数A的分析，由于方案中核酸检测面积大，每天检测的人数需要增加。在参数分析方面，方案要求对未检测人员进行隔离，以减少无症状感染者中病毒向阴性人群传播。调整这两个参数，调整比例如下:

|  |  |  |
| --- | --- | --- |
| 参数 | 变化 | 变化 |
| 一个 | 增加30% | 增加60% |
|  | 削减30% | 削减60% |

图14:调整比率

图15



结果图如下:

我们发现，无症状人群在变化的情况下减少到零的时间比不变的情况下要短。同时，变化率越大，所需时间越短。

综上所述，我们发现大面积的核酸检测和控制未检测人员外出可以应对病毒携带者在未感染人群中向整个人群传播。

### 7.浓度模型

#### 7.1问题分析

问题3要求我们提出一种新的组检测方案，该方案不需要受到组的约束，需要用具体的数值例子来说明该方法下检测的效率。我们考虑的是先确定分组的必要性，然后通过缩小阳性感染者的样本范围，建立更有效的检测方案。

首先考虑我们的目标。根据课题的要求，我们需要开发一个更高效的模型。因此，我们的模型以检测次数为标准，在样本和感染率不变的情况下，通过减少检测次数来达到高效率。

其次，我们考虑过程。我们的想法是使用的想法集中凝结感染者到一个更小的正样本有限样本范围比总样本范围,增加样本范围内的感染率,然后凝结后者模型替换成集团检测模型来计算总数量的检测。

最后,我们分析结果,建立了一个模型,凝聚有限数量的正样本到一个更小范围的样本,并在多个数据集的值,计算的总数检测在多个数据集,并把它与第一个模型的检测数量。有效性。

#### 7.2模型建设

**7.2.1因素**

groups()的数量:

从参考中我们知道，如果样品不分组进行检测，则每个人的检查次数为1次。[4]如果要进行分组检测，那么分组后人均检测次数小于1次，则认为分组检测的效果要比不进行检测的效果好。分组检测，否则不分组检测效果更好。因此，我们用计算来验证分组是否能有效提高效率。[9]

在分组的情况下，假设每组中有人数，变量表示每个人的测试次数，表示阳性测试结果的概率，表示阴性测试结果的概率。在哪里。

如果一组混合样本的检测结果为阴性，则该组被检测为阴性的概率可计算为。因为不需要执行单独的测试，所以组中每个人的平均测试次数为。

如果一组混合样本的检测结果为阳性，由于一组阴性和阳性的样本只有两种可能的检测结果，所以阴性检测结果的概率为，所以该组阳性检测的概率为。小组中的每个人都需要做单独的测试，所以每个人的平均测试次数是。

通过数学知识列出测试次数的分布规律

|  |  |  |
| --- | --- | --- |
| 许多测试 |  |  |
| 概率 |  |  |

图16:试验次数分布规律

找出人均检查次数

(22）

我们将按人均检查次数确定分组的问题转化为通过与1的关系确定分组的必要性的问题。[4]经过一系列的数学计算，如推导，发现当小于0.31时，分组就会有效，而由香港的感染率推算出的香港流行率约为0.001，远小于0.31的计算值。

因此，通过计算，我们可以知道，在我们的模型条件下，分组是有利于检测方法的，所以我们可以得出需要分组的结论。

**7.2.2处理过程**

通过步骤1中严格的计算过程，我们知道分组是必要的，因此我们建立了集中优化的分组检测模型。压缩模型的核心思想是使用组检测筛选出那些积极的测试结果,消除负面结果的人分组,减少的总数和样品数量有限的阳性样本,增加感染率在剩余的样品,一个组中检测模型和重用问题。**我们考虑的最大数量的检测模型,因为当特定值的探测效率下检测的最大数量的可能性高于组的效率问题1的检测模型,然后就可以知道模型的效率将高于1的一组问题。**检查模型。具体步骤如下[7]

由于样本中受感染人数不同，被随机分配到不同群体的受感染人数的概率也会不同。我们考虑的是检测次数最多的情况，即每一个感染者属于一个群体，即每一个群体中只有一个人。受感染人数为:

（23）

其中，为受感染人数，即群体人数的下限，为感染率，为样本总数。我们将遍历满足条件的组的数量。从所有的组中，选择需要检测总数最少的组。

**7.2.3计算过程**

使用步骤1中的遍历方法，找出最大检测次数下的组数。有感染者的群体数量为。然后对各组混合样本进行检验和分析，计算混合样本阴性的患者人数为

(24）

其中，为在试验次数最多的情况下有感染者的组数，即感染者的组数，为样本总数，为组数。

因此，我们可以很容易地知道，剩下的需要检测的人数是，

(25）

然后我们使用组问题1的检测模型计算的总数检测剩余的部分,并将计算结果添加到浓缩检测的数量,这是团体的数量,计算的总数浓缩优化模型的检查:[6]

（26）

其中，为总检测数，为代入模型1后计算的剩余人检测总数。

**7.3结果比较**

我们将具体值代入问题一的群检测模型和问题三的压缩优化模型中。通过对各种不同情况的效率比较，证明富集优化模型的效率高于群检测模型。具体的替换值和枚举遍历法计算的组数如下:

|  |  |  |  |
| --- | --- | --- | --- |
| 数字号码 |  |  |  |
| 1 | 400 | ０．００１ | 40000 |
| 2 | 200 | ０．００１ | 20000 |
| 3. | One hundred. | ０．００１ | 10000 |
| 4 | 500 | 0．002 | 30000 |
| 5 | 625 | 0．002 | 40000 |

图17:具体数值

利用上述值，我们分别替换问题1中优化的浓度模型和基本组检测模型进行合理计算，得到各模型的如下期望。A.数值越大，需要的检测次数越多，对应的模型效率也越低。我们比较两者的效率，并通过图表直观地反映两者之间的关系:

|  |  |  |
| --- | --- | --- |
| 数字号码 | 集中优化结果 | 组检测模型 |
| 1 | 1160.252 | 2504.995 |
| 2 | 586.6901 | 1252.5 |
| 3. | 290.0631 | 626.25 |
| 4 | 1375.4 | 2624.283 |
| 5 | 1831.566 | 3499.044 |

图18:两个模型的效率

从图中可以看出，问题3中的集中优化模型的检测次数最大，计算出的结果远小于问题1中的基本群检测模型，由此可知，集中优化模型的效率远远高于群检模型的效率。

### Reference

* [1] [Venkatesh Saanika,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Venkatesh%20Saanika) [Bahsoun MohamadAli,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Bahsoun%20MohamadAli) [Arezza Giordano,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Arezza%20Giordano) [Digregorio Justin,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Digregorio%20Justin) [Schweizer Tom A.,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Schweizer%20Tom%20A.) [Churchill Nathan W,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Churchill%20Nathan%20W) [Munoz David G.,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Munoz%20David%20G.) [Fischer Corinne E.,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Fischer%20Corinne%20E.) [Khademi April](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Khademi%20April), “White matter texture abnormalities are associated with delusional severity in a cognitively mixed sample of older adults”, The Alzheimer's Association, 2020
* [2] N/A, “Sensitivity evaluation of 2019 novel coronavirus (SARS-CoV-2) RT-PCR detection kits and strategy to reduce false negative.” Public Library of Science , 2020
* [3] Xingfa Meng, Qunying Xu, Yueqiang Fang, “Research and Application of Calculation Model of Optimal Content of Mixed Samples”, Chinese Journal of Preventive Medicine, 1993
* [4] Qing Ma, Ling Chen, Qingtang Hou, Yuling Yang, “Preliminary Study on the Calculation Method and Application of the Component Content of Mixed Biological Sample”, Chinese Journal of Forensic Medicine, 2013
* [5] [Kaili Cui,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Kaili%20Cui) [Junke Kou](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Junke%20Kou), “Convergence Rates of Wavelet Density Estimators for Strongly Mixing Samples”, Bulletin of the Iranian Mathematical Society, 2020
* [6] [Gerstgrasser Marcel,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Gerstgrasser%20Marcel) [Cloots Michael,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Cloots%20Michael) [Stirnimann Josef,](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Stirnimann%20Josef) [Wegener Konrad](https://schlr.cnki.net/home/search?ad=1&sw-input-ath=Wegener%20Konrad), “Focus shift analysis, to manufacture dense and crack-free SLM-processed CM247LC samples”, Journal of Materials Processing Technology, 2020
* [7] Nan Guo, “Focus shift analysis, to manufacture dense and crack-free SLM-processed CM247LC samples”, China Practical Medicine, 2020
* [8] Xiaofei Li, Yingrong Du, Songqin Lv, Wanlan Li, Liping Xu, Juan Zhang, Yongrui Yang, “Preliminary study on the clinical application of novel coronavirus antibody detection”, Chinese Journal of Virology, 2020
* [9] Tingting Yu, Xiaofei Yu, Hongying Wang, Runwu Zhang, Caimei Ding, Lihua Li, Jing Bai, Dongling Li, Dong Pu, “Application of Viral Load Test in Diagnosis of 102 Cases of Uncertain HIV Antibody Samples”, Journal of Kunming Medical University, 2020

### Appendix

N40000

r=[0.0001 0.0002 0.0003 0.0004 0.0005 0.0006 0.0007 0.0008 0.0009 0.0010 0.0015 0.0020 0.0025 0.0030 0.0035 0.0040 0.0045 0.0050 0.0055 0.0060 0.0065 0.0070 0.0075

];

E=[998.95 1197.91 1394.17 1588.79 1774.07 1943.84 2096.91 2241.50 2375.87 2503.43 3061.86 3529.82 3942.70 4312.27 4652.55 4971.15 5265.39 5546.13 5811.91 6065.44 6309.18 6542.47 6765.70

];

a=[800.20 800.80 801.80 817.60 874.20 971.80 1037.80 1095.00 1192.40 1250.80 1511.00 1713.40 1924.40 2135.00 2284.00 2420.00 2558.40 2713.80 2854.80 2968.00 3066.00 3215.00 3325.20

];

plot(r,a,'k-');

N4000

r=[0.001 0.002 0.003 0.004 0.005 0.006 0.007 0.008 0.009 0.01 0.015 0.02 0.025 0.03 0.035 0.04 0.045 0.05 0.055 0.06 0.065 0.07 0.075 0.08 0.085 0.09 0.095 0.1

];

E=[250.3553 352.67072 430.5417 496.42652 553.48684 605.1929 653.3514 697.07838 738.76368 777.26912 946.1744 1087.40578 1209.7616 1319.88988 1420.50166 1513.45622 1599.61382 1680.94978 1756.3382 1828.7859 1898.50662 1964.98666 2027.12618 2088.84272 2146.60632 2204.91974 2258.81296 2311.48544

];

a=[122.6 177.8 214 245.6 266.6 292.4 325 335 360 371.2 450.4 521.6 565.8 609.2 666.8 719.8 755.2 788.2 823 859 871.2 908.6 913.4 956.4 967.2 988.4 1032 1039.8

];

plot(r,a,'k-');

r1e3

N=[1000.0 2000.0 4000.0 7000.0 10000.0 12000.0 14000.0 16000.0 18000.0 20000.0 22000.0 24000.0 26000.0 28000.0 30000.0 32000.0 34000.0 36000.0 38000.0 40000.0 43000.0 46000.0 50000.0

];

E=[999.0 1197.9 1394.2 1588.8 1774.1 1943.8 2096.9 2241.5 2375.9 2503.4 3061.9 3529.8 3942.7 4312.3 4652.6 4971.2 5265.4 5546.1 5811.9 6065.4 6309.2 6542.5 6765.7

];

a=[30.8 63.6 125.0 220.0 310.8 367.4 438.2 485.0 563.8 600.6 666.6 751.4 807.8 865.0 932.6 989.8 1066.0 1110.8 1178.6 1245.8 1351.0 1434.2 1541.8

];

r\_b=0.03577;

A=E./r\_b;

plot(N,E,'ko');

R5e4

N=[2000.0000 4000.0000 6000.0000 8000.0000 10000.0000 12000.0000 14000.0000 16000.0000 18000.0000 20000.0000 22000.0000 24000.0000 26000.0000 28000.0000 30000.0000

];

a=[43.4000 89.0000 129.0000 181.6000 225.6000 264.0000 305.8000 348.2000 395.8000 436.2000 493.2000 524.4000 559.2000 618.0000 669.8000

];

E=[88.5992 177.3872 266.3366 354.4862 443.1809 531.3124 621.0274 709.2072 798.2149 886.3788 975.9661 1064.5808 1153.2044 1241.2974 1330.7903

];

r\_b=0.03577;

A=E./r\_b;

plot(N,a,'k-');

Calculate data

#include<bits/stdc++.h>

using namespace std;

typedef long long ll;

double E[200050];

int cnt[200050];

double N,n,r;

const int repN=100;

double testN,testr;

double fakene\_rate=0.00883;

void init(){

memset(E,0,sizeof(E));

}

double qsm(double x,ll ex){

double ans=1;

double tmp=x;

while(ex){

if(ex&1)ans\*=tmp;

tmp\*=tmp;

ex>>=1;

}

return ans;

}

double comb(int A,int B){

double ans=1;

for(int i=1;i<=A;i++){

ans=ans\*(B-i+1)/i;

}

return ans;

}

const int TOTturn=5;

int main(){

freopen("需要测试的N和r.txt","r",stdin);

freopen("不同N和r的测试结果.txt","w",stdout);

default\_random\_engine DRE;

init();

int num;

scanf("%d",&num);

for(int i=0;i<num;i++){

scanf("%lf%lf",&testN,&testr);

N=testN;

r=testr;

n=N\*r;

double resA=0,resE=0,po=0,ne=0;

printf("N=%.0lf\nr=%.5lf\n",N,r);

for(int i=1;i<=TOTturn;i++){

int mina;

double minE=1e6;

init();

for(int a=N/50;a<=N/3;a++){ //改这里可以加速

double p=N/a;

uniform\_int\_distribution<unsigned>choose(0,a);

for(int i=0;i<repN;i++){

double e=0;

for(int i=1;i<=a;i++)cnt[i]=0;

int tot=0;

while(tot<n){

int tmp=choose(DRE);

if(cnt[tmp]<p){

cnt[tmp]++;

tot++;

}

}

tot=0;

for(int i=1;i<=a;i++)if(cnt[i])tot++;

E[a]+=(double)(a+p\*tot)/repN;

}

// printf("a= %d E= %.4lf\n",a,E[a]);

if(E[a]<minE){

minE=E[a];

mina=a;

}

}

printf("%d\t\t%.4lf\n",mina,minE);

resA+=mina;

resE+=minE;

}

resA/=TOTturn;

resE/=TOTturn;

double resNE=(resE-resA)\*fakene\_rate/N;

double resPO=(resE-resA-n)/N;

printf("final:\n");

printf("a=%.4lf\nE=%.6lf\npo=%.6lf\nne=%.6lf\n\n",resA,resE,resPO,resNE);

}

return 0;

}

Format conversion

#include<bits/stdc++.h>

using namespace std;

typedef long long ll;

int main(){

int n=28;

double x[99];

double E[99];

double a[99];

memset(x,0,sizeof(x));

memset(a,0,sizeof(a));

memset(E,0,sizeof(E));

freopen("N=40000,r的比较结果（没测试完）.txt","r",stdin);

freopen("output1.txt","w",stdout);

for(int i=1;i<=28;i++){

string s;

cin>>s;

scanf("\nr=%lf\n",&x[i]);

for(int j=0;j<5;j++){

double e,tmp;

scanf("%lf%lf",&tmp,&e);

E[i]+=e;

a[i]+=tmp;

}

a[i]/=5;

E[i]/=5;

}

for(int i=1;i<=28;i++)printf("%.5lf\n",x[i]);

printf("\n");

for(int i=1;i<=28;i++)printf("%.5lf\n",a[i]);

printf("\n");

for(int i=1;i<=28;i++)printf("%.5lf\n",E[i]);

printf("\n");

return 0;

}

Model 1:

|  |  |  |
| --- | --- | --- |
| N | r | EX |
| 40000 | 0.001 | 2504.994858 |
| 20000 | 0.001 | 1252.5 |
| 10000 | 0.001 | 626.25 |
| 30000 | 0.002 | 2624.282631 |
| 40000 | 0.002 | 3499.043508 |

Model 2:

|  |  |  |
| --- | --- | --- |
| N | r | EX |
| 40000 | 0.001 | 1160.252 |
| 20000 | 0.001 | 586.6901 |
| 10000 | 0.001 | 290.0631 |
| 30000 | 0.002 | 1375.4 |
| 40000 | 0.002 | 1831.566 |

Dividing group number:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | N | r | EX | condense |
|  | **40000** | **0.001** | 2505 |  |
| 50 | 32000 | 0.00125 | 2231.63354 | 2281.63354 |
| 64 | 25000 | 0.0016 | 1963.831388 | 2027.831388 |
| 80 | 20000 | 0.002 | 1749.521754 | 1829.521754 |
| 100 | 16000 | 0.0025 | 1558.599371 | 1658.599371 |
| 125 | 12800 | 0.00313 | 1389.582739 | 1514.582739 |
| 160 | 10000 | 0.004 | 1221.886773 | 1381.886773 |
| 200 | 8000 | 0.005 | 1088.544314 | 1288.544314 |
| 250 | 6400 | 0.00625 | 969.7532935 | 1219.753293 |
| 320 | 5000 | 0.008 | 853.3802358 | 1173.380236 |
| 400 | 4000 | 0.01 | 760.2522781 | 1160.252278 |
| 500 | 3200 | 0.0125 | 677.2872187 | 1177.287219 |
| 625 | 2560 | 0.01562 | 603.282896 | 1228.282896 |
| 800 | 2000 | 0.02 | 530.9692212 | 1330.969221 |
| 1000 | 1600 | 0.025 | 473.0254382 | 1473.025438 |
| 1250 | 1280 | 0.03125 | 421.4049634 | 1671.404963 |
| 1600 | 1000 | 0.04 | 370.8352108 | 1970.835211 |
| 2000 | 800 | 0.05 | 330.3665845 | 2330.366585 |
| 2500 | 640 | 0.0625 | 294.3142318 | 2794.314232 |
| 4000 | 400 | 0.1 | 230.7319467 | 4230.731947 |
| 5000 | 320 | 0.125 | 205.5525553 | 5205.552555 |
| 8000 | 200 | 0.2 | 161.1459321 | 8161.145932 |
| 10000 | 160 | 0.25 | 143.5603461 | 10143.56035 |
| 20000 | 80 | 0.5 | 100.2642509 | 20100.26425 |
|  |  |  | 0 | 0 |
|  | **20000** | **0.001** | 1252.5 | 1252.5 |
| 25 | 16000 | 0.00125 | 1115.81677 | 1140.81677 |
| 32 | 12500 | 0.0016 | 981.9156942 | 1013.915694 |
| 40 | 10000 | 0.002 | 874.7608769 | 914.7608769 |
| 50 | 8000 | 0.0025 | 779.2996856 | 829.2996856 |
| 80 | 5000 | 0.004 | 610.9433867 | 690.9433867 |
| 100 | 4000 | 0.005 | 544.2721568 | 644.2721568 |
| 125 | 3200 | 0.00625 | 484.8766467 | 609.8766467 |
| 160 | 2500 | 0.008 | 426.6901179 | 586.6901179 |
| 200 | 2000 | 0.01 | 380.1261391 | 580.1261391 |
| 250 | 1600 | 0.0125 | 338.6436093 | 588.6436093 |
| 400 | 1000 | 0.02 | 265.4846106 | 665.4846106 |
| 500 | 800 | 0.025 | 236.5127191 | 736.5127191 |
| 625 | 640 | 0.03125 | 210.7024817 | 835.7024817 |
| 800 | 500 | 0.04 | 185.4176054 | 985.4176054 |
| 1000 | 400 | 0.05 | 165.1832923 | 1165.183292 |
| 1250 | 320 | 0.0625 | 147.1571159 | 1397.157116 |
| 2000 | 200 | 0.1 | 115.3659733 | 2115.365973 |
| 2500 | 160 | 0.125 | 102.7762776 | 2602.776278 |
| 4000 | 100 | 0.2 | 80.57296607 | 4080.572966 |
| 5000 | 80 | 0.25 | 71.78017306 | 5071.780173 |
| 10000 | 40 | 0.5 | 50.13212545 | 10050.13213 |
|  |  |  | 0 | 0 |
|  | **10000** | **0.001** | 626.25 | 626.25 |
| 16 | 6250 | 0.0016 | 490.9578471 | 506.9578471 |
| 20 | 5000 | 0.002 | 437.3804385 | 457.3804385 |
| 25 | 4000 | 0.0025 | 389.6498428 | 414.6498428 |
| 40 | 2500 | 0.004 | 305.4716933 | 345.4716933 |
| 50 | 2000 | 0.005 | 272.1360784 | 322.1360784 |
| 80 | 1250 | 0.008 | 213.345059 | 293.345059 |
| 100 | 1000 | 0.01 | 190.0630695 | 290.0630695 |
| 125 | 800 | 0.0125 | 169.3218047 | 294.3218047 |
| 200 | 500 | 0.02 | 132.7423053 | 332.7423053 |
| 250 | 400 | 0.025 | 118.2563596 | 368.2563596 |
| 400 | 250 | 0.04 | 92.7088027 | 492.7088027 |
| 500 | 200 | 0.05 | 82.59164614 | 582.5916461 |
| 625 | 160 | 0.0625 | 73.57855794 | 698.5785579 |
| 1000 | 100 | 0.1 | 57.68298667 | 1057.682987 |
| 1250 | 80 | 0.125 | 51.38813882 | 1301.388139 |
| 2000 | 50 | 0.2 | 40.28648304 | 2040.286483 |
| 2500 | 40 | 0.25 | 35.89008653 | 2535.890087 |
| 5000 | 20 | 0.5 | 25.06606273 | 5025.066063 |
|  |  |  | 0 | 0 |
|  | **30000** | **0.002** | 2624.282631 | 2624.282631 |
| 75 | 24000 | 0.0025 | 2337.899057 | 2412.899057 |
| 80 | 22500 | 0.00267 | 2262.417032 | 2342.417032 |
| 100 | 18000 | 0.00333 | 2013.338188 | 2113.338188 |
| 120 | 15000 | 0.004 | 1832.83016 | 1952.83016 |
| 125 | 14400 | 0.00417 | 1795.183426 | 1920.183426 |
| 150 | 12000 | 0.005 | 1632.81647 | 1782.81647 |
| 200 | 9000 | 0.00667 | 1407.156882 | 1607.156882 |
| 240 | 7500 | 0.008 | 1280.070354 | 1520.070354 |
| 250 | 7200 | 0.00833 | 1253.052371 | 1503.052371 |
| 300 | 6000 | 0.01 | 1140.378417 | 1440.378417 |
| 375 | 4800 | 0.0125 | 1015.930828 | 1390.930828 |
| 400 | 4500 | 0.01333 | 982.4198054 | 1382.419805 |
| 500 | 3600 | 0.01667 | 875.3998928 | 1375.399893 |
| 600 | 3000 | 0.02 | 796.4538318 | 1396.453832 |
| 625 | 2880 | 0.02083 | 779.7336163 | 1404.733616 |
| 750 | 2400 | 0.025 | 709.5381573 | 1459.538157 |
| 1000 | 1800 | 0.03333 | 611.3012593 | 1611.301259 |
| 1200 | 1500 | 0.04 | 556.2528162 | 1756.252816 |
| 1250 | 1440 | 0.04167 | 544.6382413 | 1794.638241 |
| 1500 | 1200 | 0.05 | 495.5498768 | 1995.549877 |
| 1875 | 960 | 0.0625 | 441.4713477 | 2316.471348 |
| 2000 | 900 | 0.06667 | 426.9709397 | 2426.97094 |
| 2500 | 720 | 0.08333 | 380.3598098 | 2880.35981 |
| 3000 | 600 | 0.1 | 346.09792 | 3346.09792 |
| 3750 | 480 | 0.125 | 308.3288329 | 4058.328833 |
| 5000 | 360 | 0.16667 | 265.6554941 | 5265.655494 |
| 6000 | 300 | 0.2 | 241.7188982 | 6241.718898 |
| 7500 | 240 | 0.25 | 215.3405192 | 7715.340519 |
| 10000 | 180 | 0.33333 | 185.5342688 | 10185.53427 |
| 15000 | 120 | 0.5 | 150.3963764 | 15150.39638 |
|  |  |  | 0 | 0 |
|  | **40000** | **0.002** | 3499.043508 | 3499.043508 |
| 100 | 32000 | 0.0025 | 3117.198743 | 3217.198743 |
| 125 | 25600 | 0.00313 | 2779.165479 | 2904.165479 |
| 160 | 20000 | 0.004 | 2443.773547 | 2603.773547 |
| 200 | 16000 | 0.005 | 2177.088627 | 2377.088627 |
| 250 | 12800 | 0.00625 | 1939.506587 | 2189.506587 |
| 320 | 10000 | 0.008 | 1706.760472 | 2026.760472 |
| 400 | 8000 | 0.01 | 1520.504556 | 1920.504556 |
| 500 | 6400 | 0.0125 | 1354.574437 | 1854.574437 |
| 625 | 5120 | 0.01562 | 1206.565792 | 1831.565792 |
| 800 | 4000 | 0.02 | 1061.938442 | 1861.938442 |
| 1000 | 3200 | 0.025 | 946.0508764 | 1946.050876 |
| 1250 | 2560 | 0.03125 | 842.8099267 | 2092.809927 |
| 1600 | 2000 | 0.04 | 741.6704216 | 2341.670422 |
| 2000 | 1600 | 0.05 | 660.7331691 | 2660.733169 |
| 2500 | 1280 | 0.0625 | 588.6284635 | 3088.628464 |
| 4000 | 800 | 0.1 | 461.4638933 | 4461.463893 |
| 5000 | 640 | 0.125 | 411.1051105 | 5411.105111 |
| 8000 | 400 | 0.2 | 322.2918643 | 8322.291864 |
| 10000 | 320 | 0.25 | 287.1206922 | 10287.12069 |
| 20000 | 160 | 0.5 | 200.5285018 | 20200.5285 |