

Quantiles And Histograms

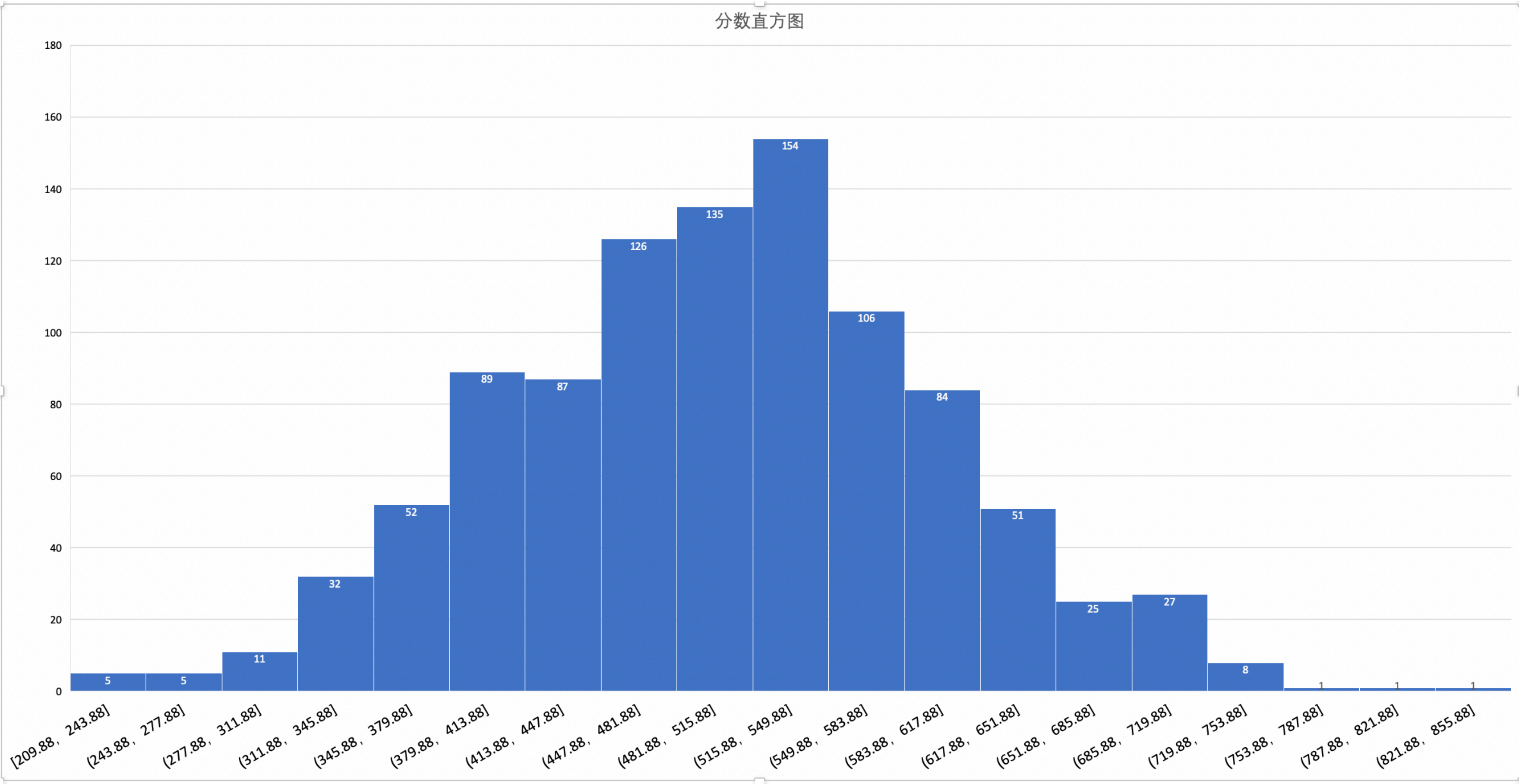
分位点和直方图

田志鹏 20190704

1. 背景概念
2. GK算法
3. Doubles Sketch

什么是直方图

分数	排序后
583.38	209.88
430.72	216.23
393.98	217.83
678.00	223.75
464.72	233.03
326.20	258.91
399.29	267.94
514.51	268.27
369.88	271.6
577.91	275.39
531.11	290.61
409.91	296.09
410.50	297.88
558.07	298.66
464.21	298.75
604.29	300.8
424.15	306.45
477.79	306.83
487.45	308.87
586.04	311.06
579.00	311.51
493.73	312.75
326.72	314.87
525.72	315.81
464.30	318.91
614.52	319.62
398.39	320.49
520.14	321.49
330.54	324.18
418.35	324.89
483.42	326.2

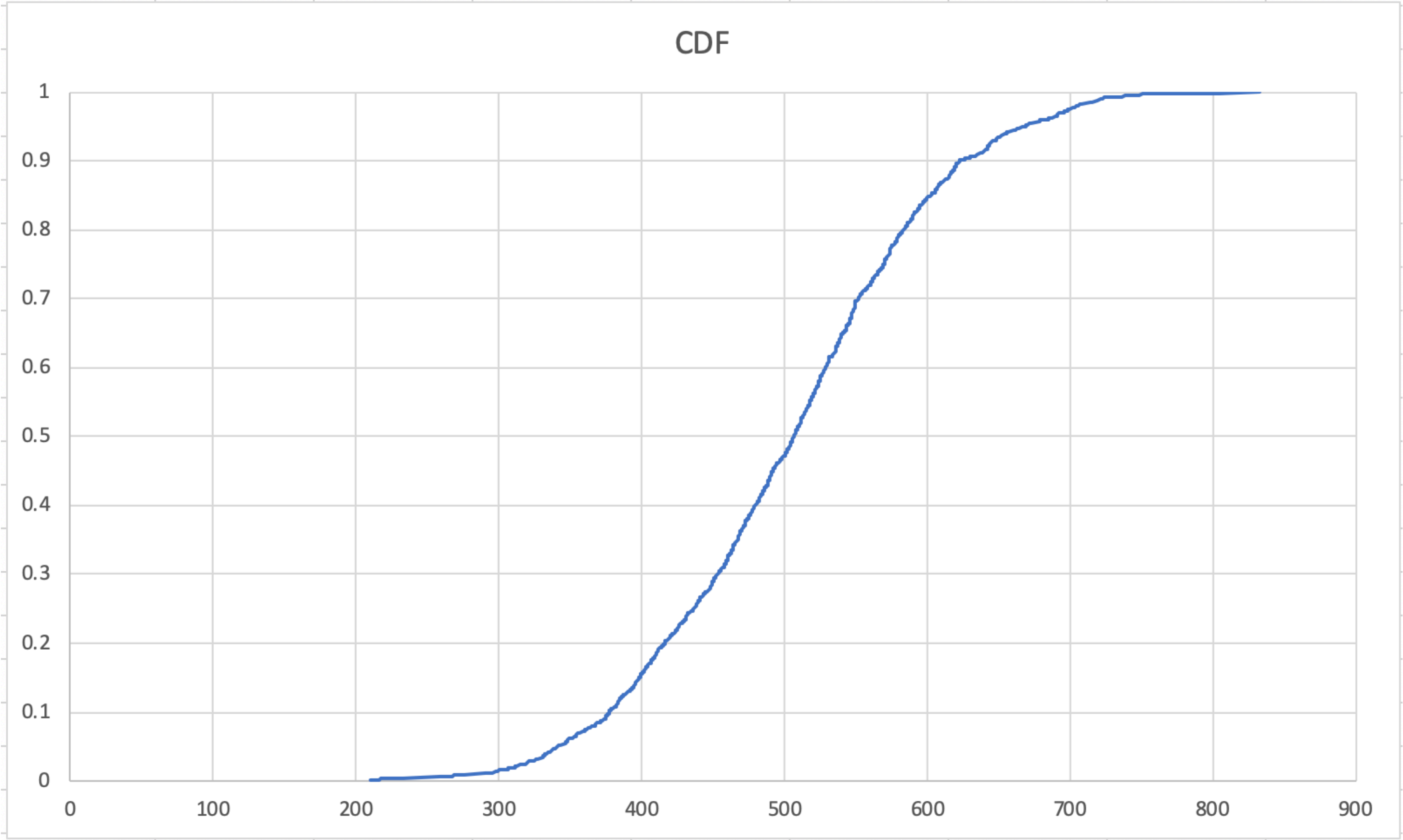


随机生成的某高中学生1000人的高考分数, 最低209, 最高832

如图横轴表示分数段, 纵轴是频次, 就是一个直方图

什么是分位点 ϕ -quantile

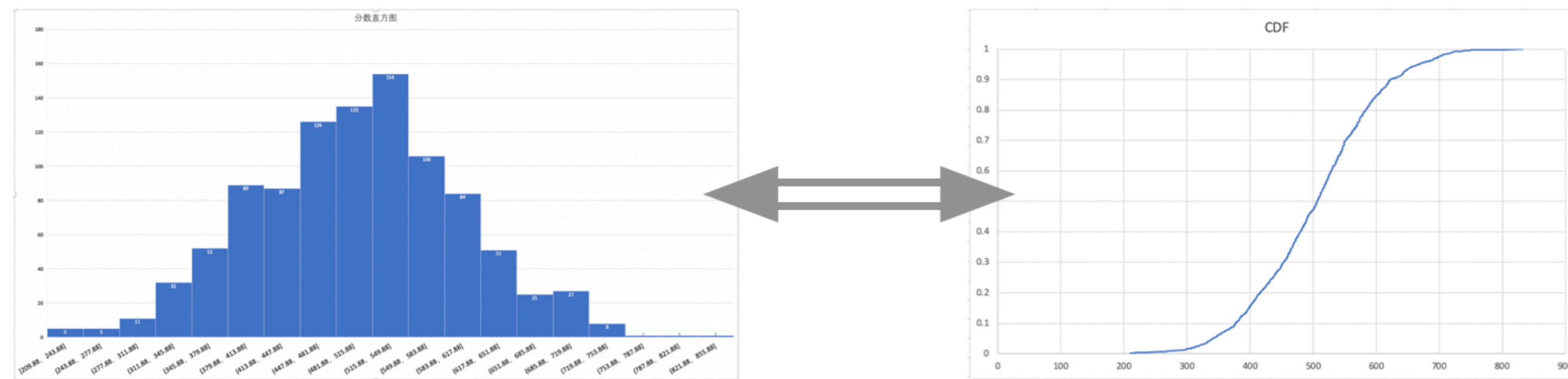
0.2	414分
0.5	506分
0.8	581分
0.9	622分



- 中位数 0.5-quantile
- 数据从小到大排序, $\text{rank} = \phi * n$ 的数据值称为 ϕ -quantile
- 所以问题就是, 求任意一个排名的分数

问题场景

- 直观的解决办法: 排序
- 时间/空间/准确性
- 大数据 and/or 实时计算 场景下如何解决
- 分位点和直方图的相似性(课后作业)



问题场景 琐碎的想法

全部数据太大 -> 允许误差 -> 只保留部分值

和Distinct Count不同, 还是需要保持具体的值的
(所以最后肯定没法优化到与n无关的时间复杂度?)

随机抽样 -> 不行

关键是rank和分布

排序, 隔几个抽一个 ✓

定义误差 ϵ :

在分位点中, 总数据量1000, 误差0.01,

允许的误差rank 20

0.5分位点的值的实际rank可能在490

510之间

同样的误差率, 总量n越大, 允许的误差的绝对值越大

最简单的情况, 在排好序的数据里

每20个抽中间那个,

就能满足误差率下的所有分位点查询

问题场景

流式数据场景特点和要求:

- 数据根本不是按顺序来的, 每个数据只读一次
- 总量 n 也不确定
- 确保误差率 ϵ
- 可以分布式算, 可以merge
- 空间复杂度可控(sublinear)

Greenwald and Khanna 2001 数据结构

元组: $\{v_i, \min_i, \max_i\}$ v_i : 代表值, \min : rank的下界, \max : rank的上界

示例: $\{ 267.94, 1, 7 \} \{ 308.87, 8, 15 \} \{ 338.44, 16, 25 \} \dots$ $\epsilon = 0.01$

提问: 倒数第20名(0.02-quantile)多少分?

回答: 338.44分, 最大的偏差是25名

结论: 只要元组长度不那么长, 就能满足偏差的要求
每个元组也不能太短, 组太多浪费空间

但是这样的结构维护的时候插入困难, GK描述了一种新的结构

GK算法 数据结构

元组: $\{v_i, g_i, \Delta_i\}$

v_i : 代表值, \min_i : rank的下界, \max_i : rank的上界

value

g_i : $\min_i - \min_{i-1}$

gap

Δ_i : $\max_l - \min_i$

delta

示例:

$\{ 267.94, 1, 7 \} \{ 308.87, 8, 15 \} \{ 338.44, 16, 25 \} \dots$

$\{ 267.94, 1, 6 \} \{ 308.87, 7, 7 \} \{ 338.44, 8, 9 \} \dots$

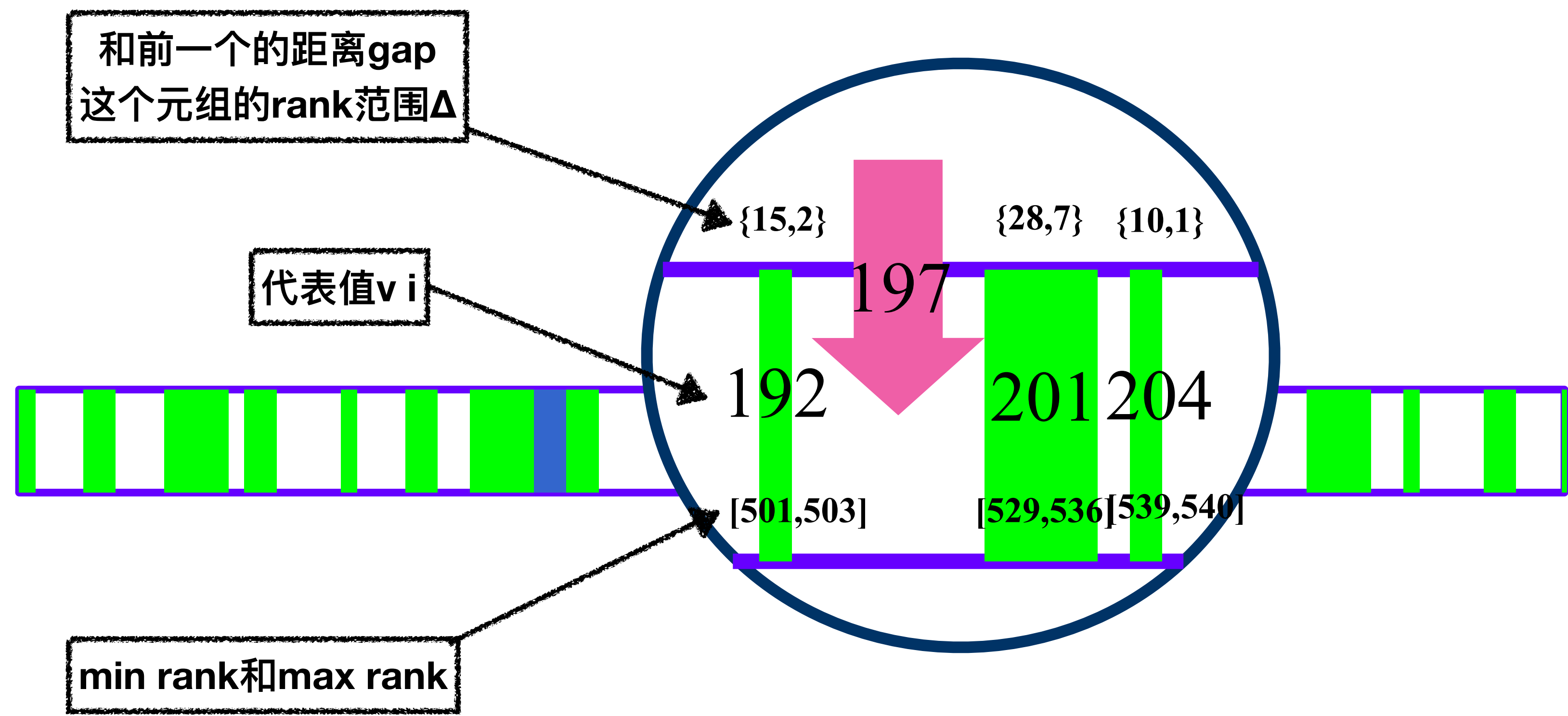
$$(g_i + \Delta_i) \leq 2\epsilon N$$

GK算法 插入操作

如何维护上述结构呢？

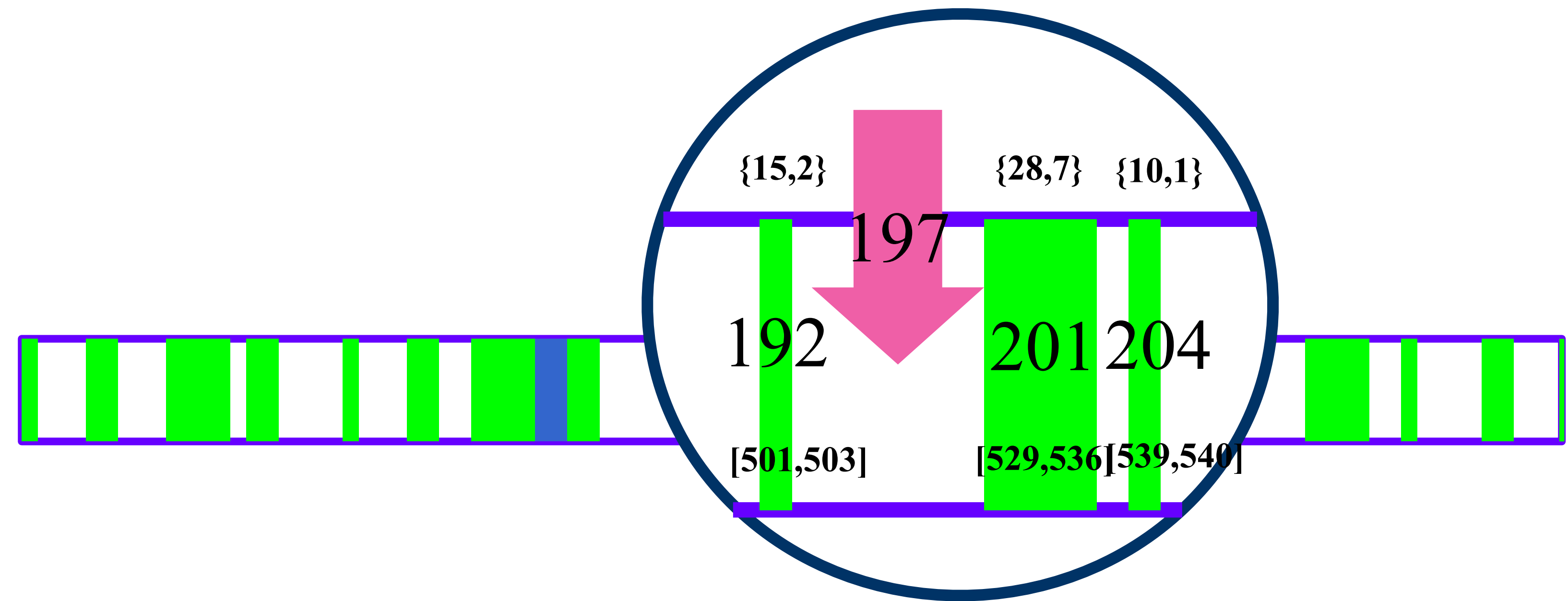
- 插入
- 删除(合并)

GK算法 操作



也就是192这个值, 可能的rank范围是501到503
和上一个元组的gap是15, 自己的rank范围是2

GK算法 插入操作



插入值自己变成一个新元组:

gap就是1

{1, 34}

Δ 是max-min = 34

公式

$$g_{\text{new}} = 1; \Delta_{\text{new}} = g_i + D_i - 1$$

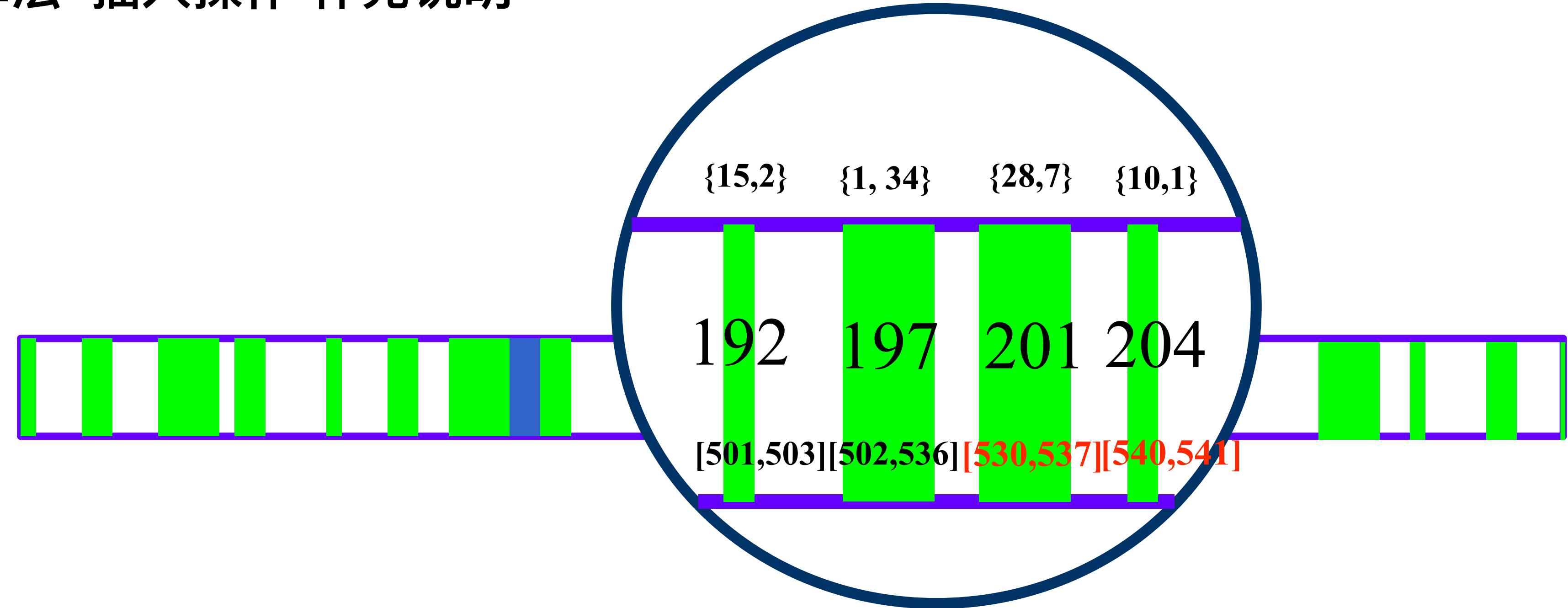
197

可能的下界是192的下界+1 = 502

[502,536]

可能的上界是201的上界 (并不减一, 而是201的上界加1)

GK算法 插入操作 补充说明

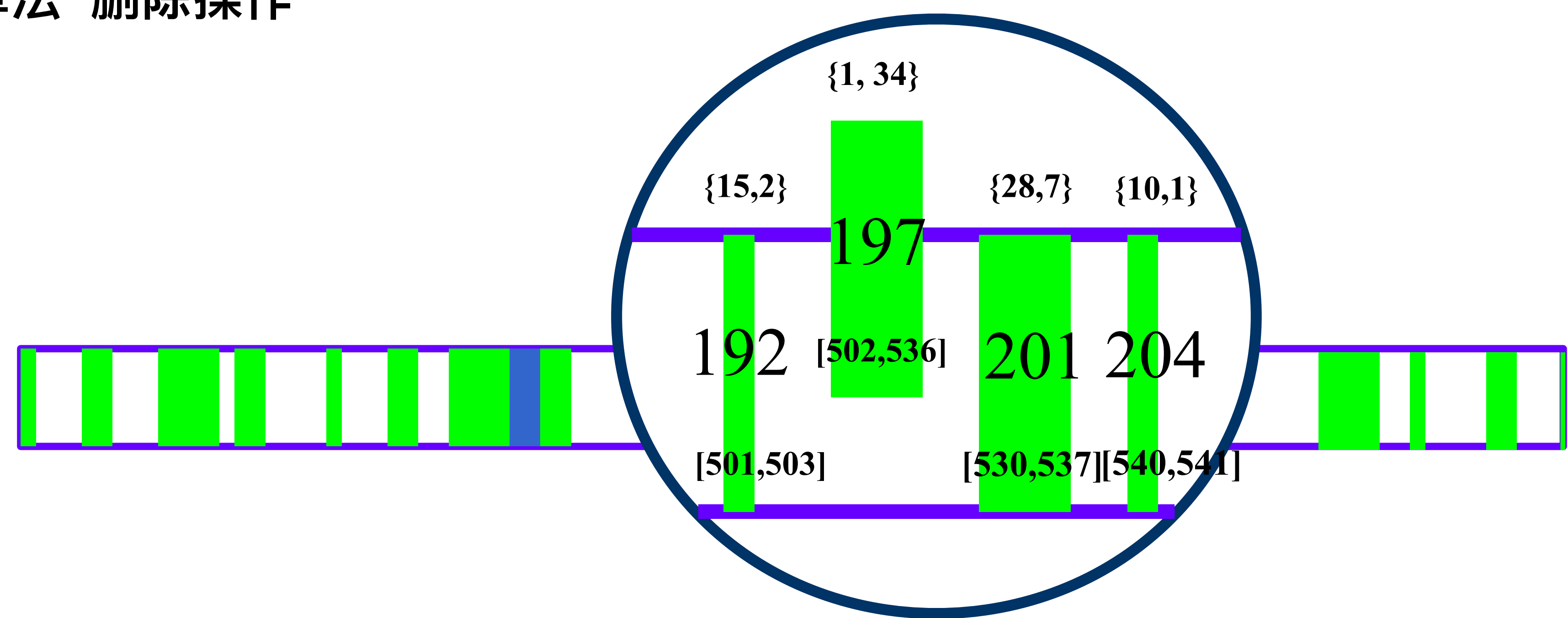


当然插入197之后:

- 排在他后面的元组所有排名都要+1
- 不要由于我们只维护gap的delta, 所有实际并不需要真的去把每个+1
- gap和delta也都不用变!

这里有同学问我, 不维护下面的min/max排名, 那图这些排名咋知道的:
把前序所有元组的gap加起来就是min
再加自己的delta就是max

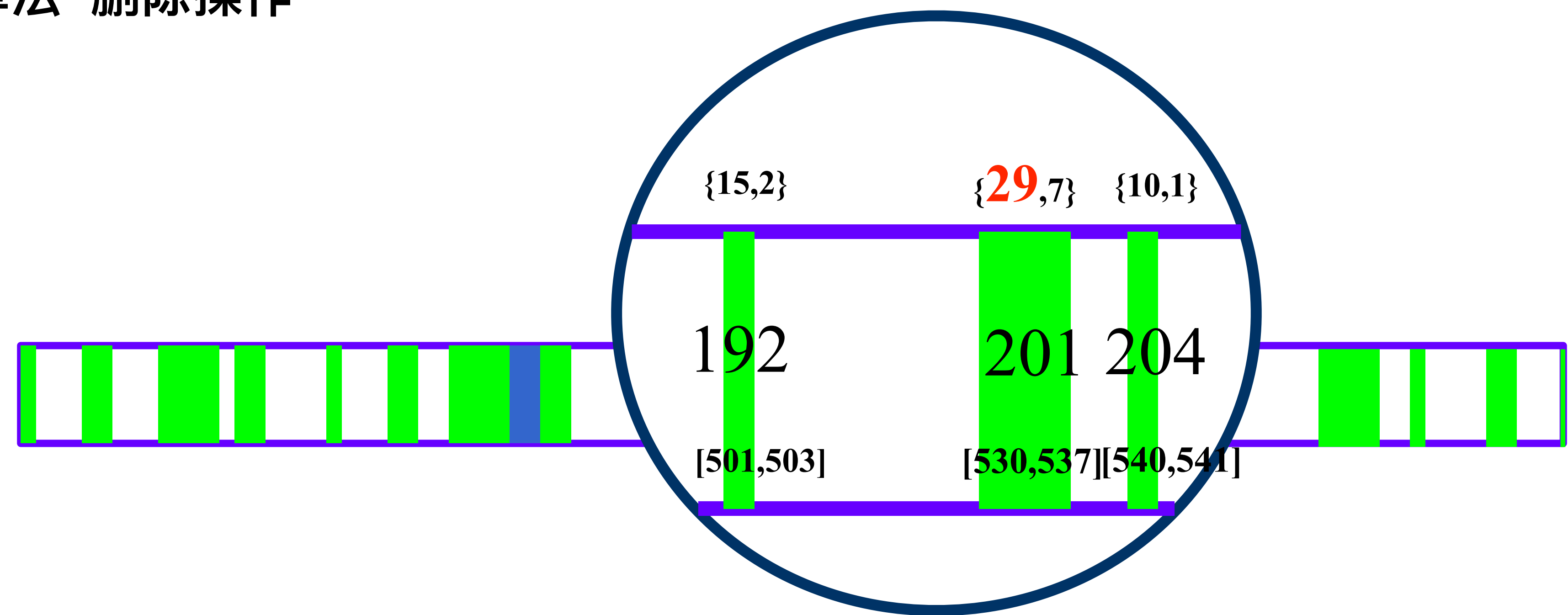
GK算法 删除操作



把197合并到201上:

- 对201这个值来讲, 他的可能排名的上下界并不会变, Δ 也不会变
- 201这个元组的gap需要变(gap是和前一个元组的距离, 前一个元组删了)

GK算法 删除操作



把197合并到201上:

- 对201这个值来讲, 他的可能排名的上下界并不会变, Δ 也不会变
- 201这个元组的gap需要变
 - (gap是和前一个元组的距离, 前一个元组删了, 把他的gap加上)

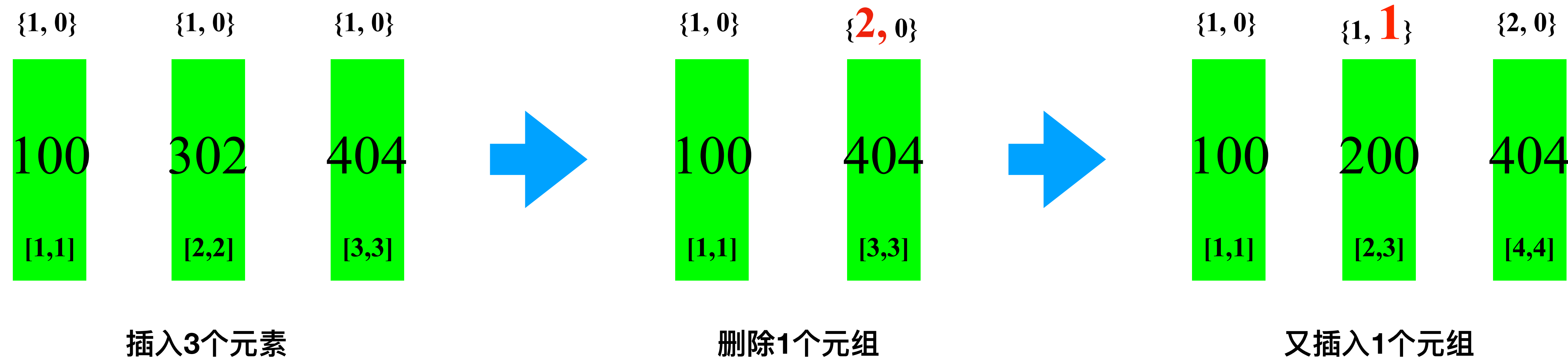
公式

$$g_i = g_i + g_{i-1}$$

GK算法 初始插入操作

200 302 404 100

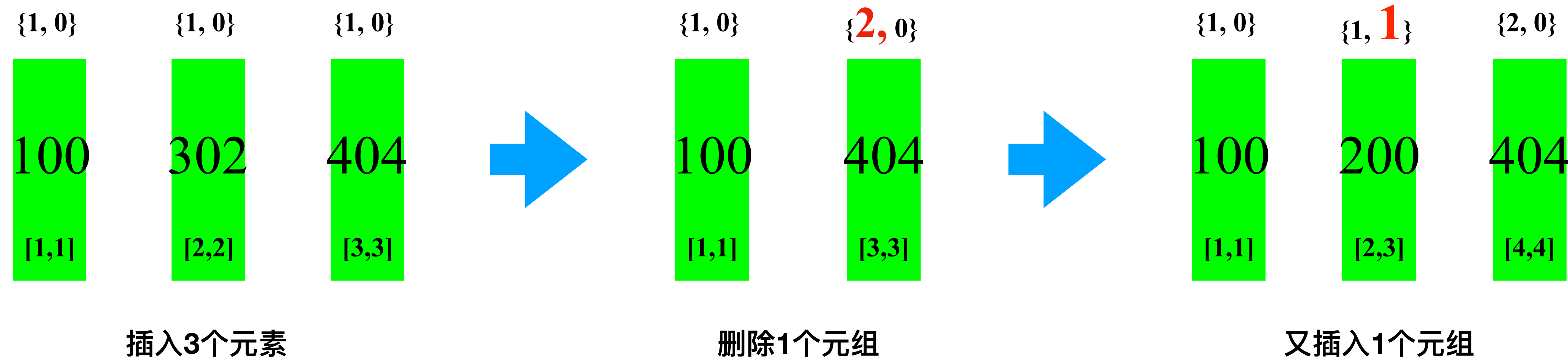
- 数据像这样一个一个来. 我们来一个就插一个 $g=1$ $\Delta=0$ 的新元组
- 那 g 和 Δ 什么时候长大呢?
- 聪明的你回顾刚才插入和删除的两个公式



GK算法 初始插入操作

200 302 404 100

- 数据像这样一个一个来. 我们来一个就插一个 $g=1$ $\Delta=0$ 的新元组
- 那 g 和 Δ 什么时候长大呢?
- 聪明的你回顾刚才插入和删除的两个公式



GK算法

执行时机: 每 $1/2\varepsilon$ 从右到左执行一波删除, 确保 $g+\Delta$ 小于等于 $2\varepsilon N$

优化目标: g 尽量大 Δ 尽量小

优化:

- 根据元组的delta不同分组, delta小的尽量删
- 利用树形加快操作效率

空间复杂度: $(11/2\varepsilon)\log(2\varepsilon n)$

问题: 合并后错误率提高

Doubles Sketch

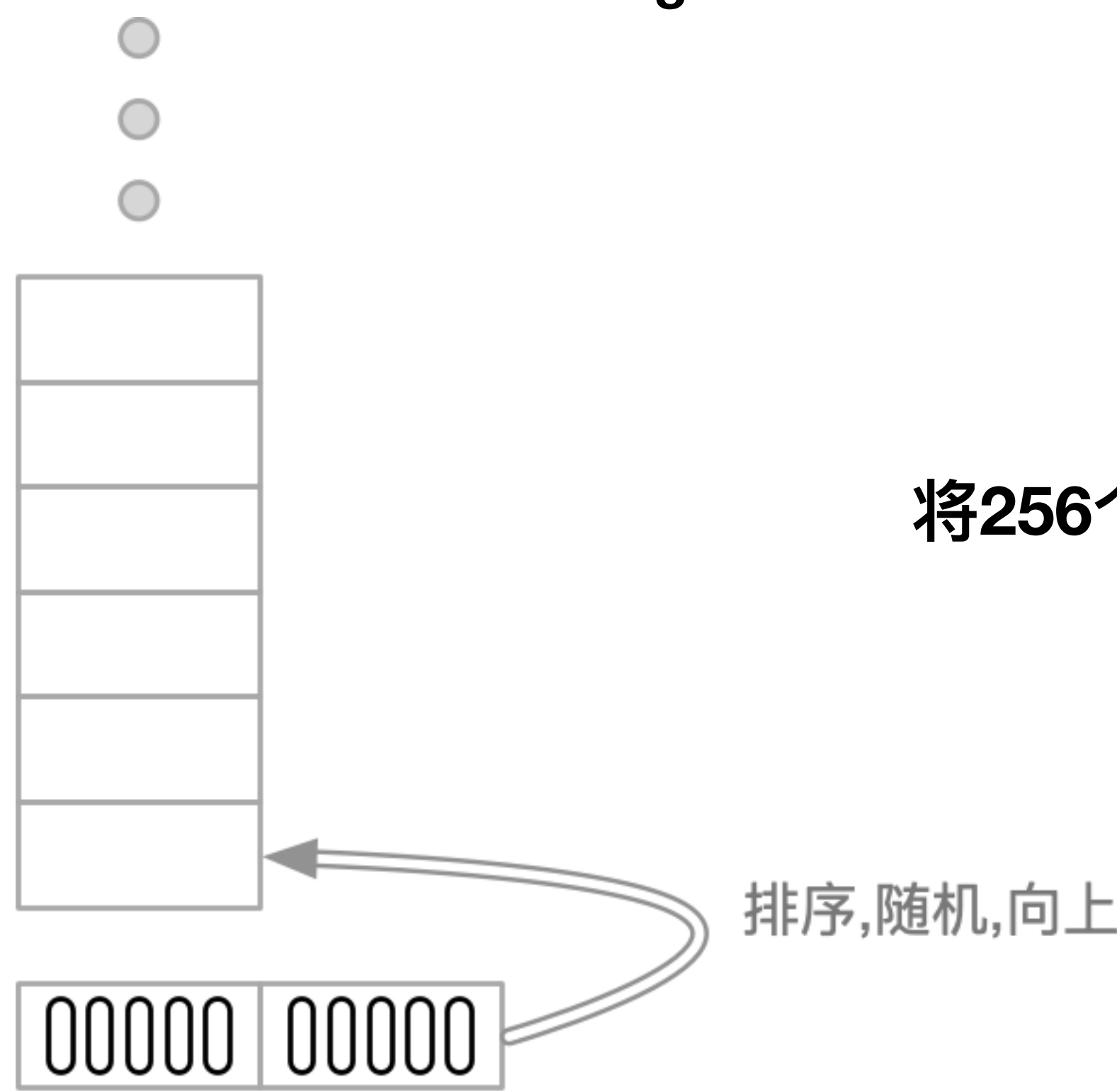
Fig. 1



如图 每个格子是128个空间
每新来一个元素就插入下面的两个格子里

Doubles Sketch

Fig. 2



当俩工作区满了(来了256个元素)
将256个工作区元素排序, 排序后每两个(奇偶)随机抽一个
抽出的128个放到上一级格子

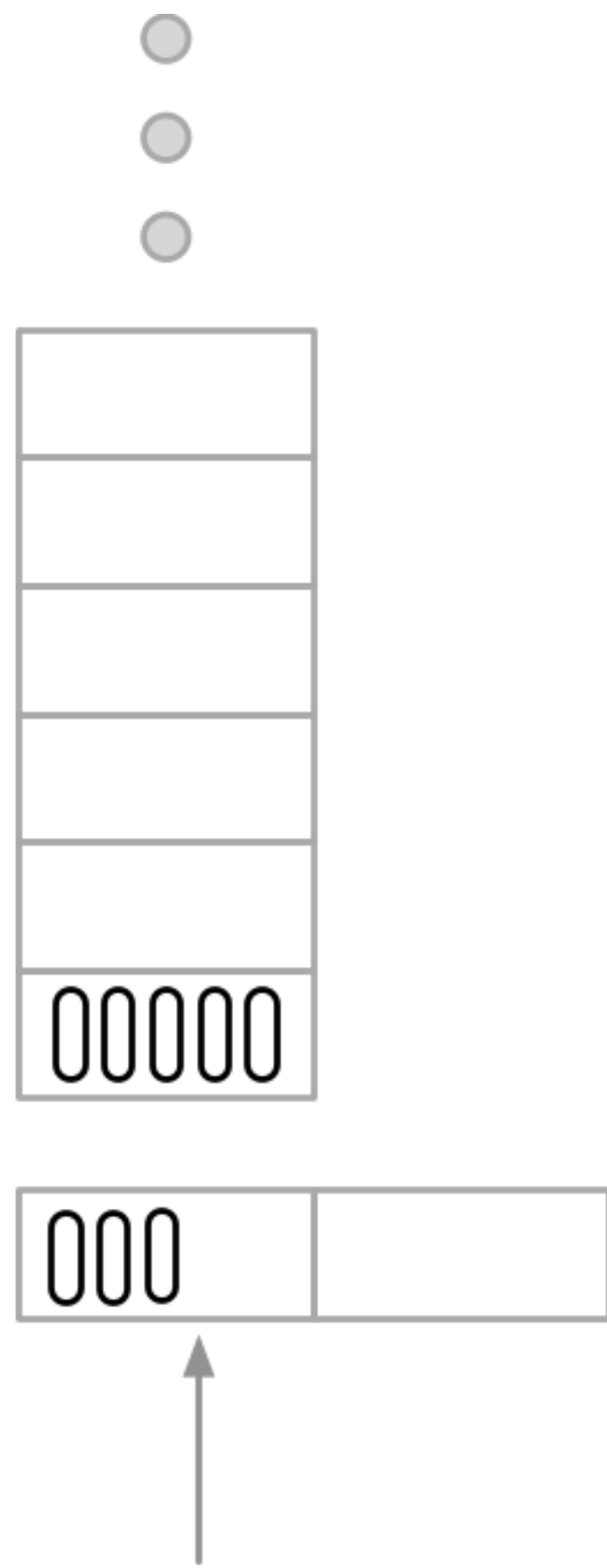
Doubles Sketch

Fig. 3



Doubles Sketch

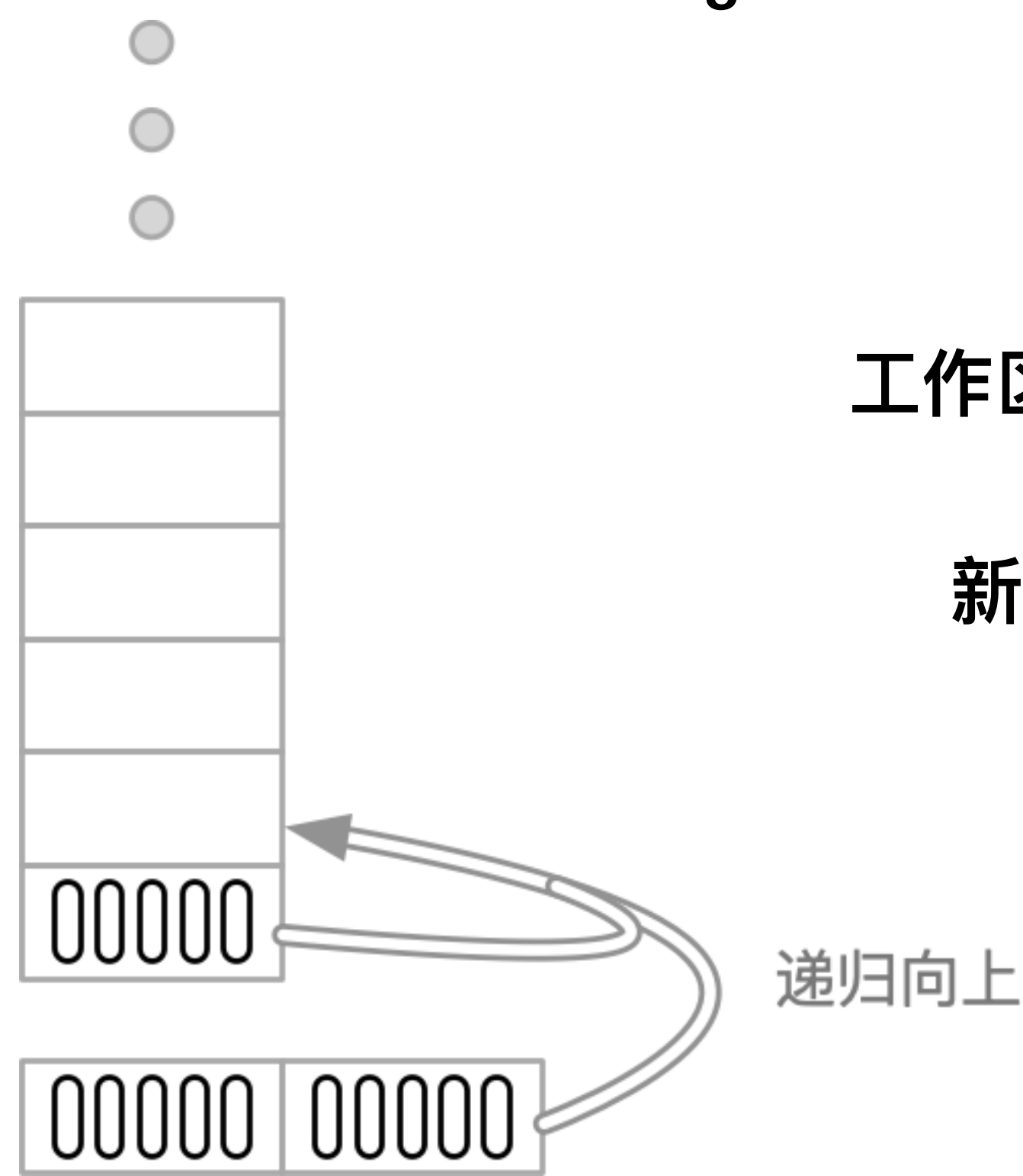
Fig. 4



新元素继续向工作区放

Doubles Sketch

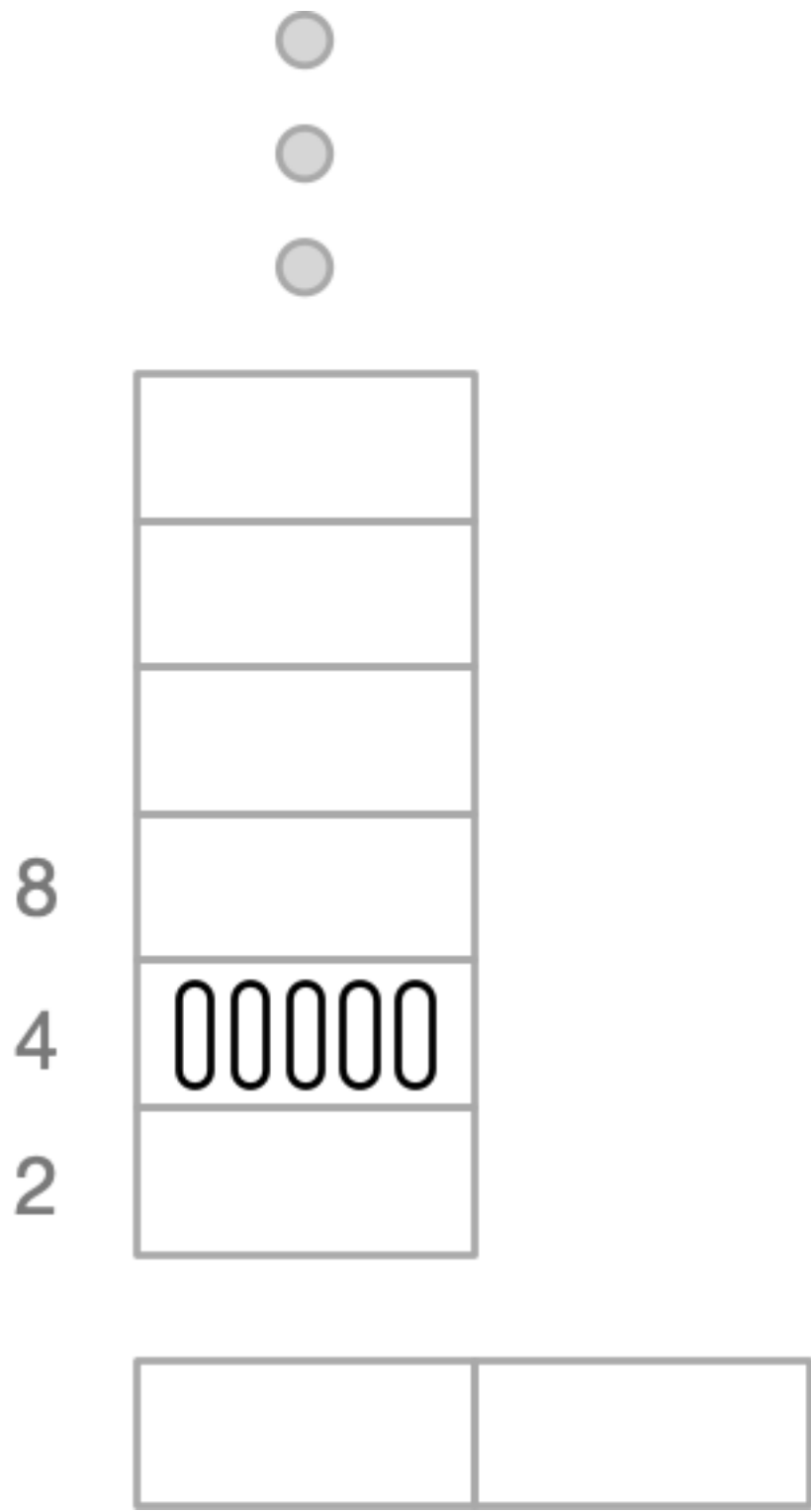
Fig. 5



工作区又放满了, 执行Fig.2里那种“排序抽样向上”的动作,
然而上一级格子已经满了.
新抽的128和上一级格子的128继续”排序抽样向上”

Doubles Sketch

Fig. 6



这样经过两次抽样, 就上升到第二级了

可以看出:

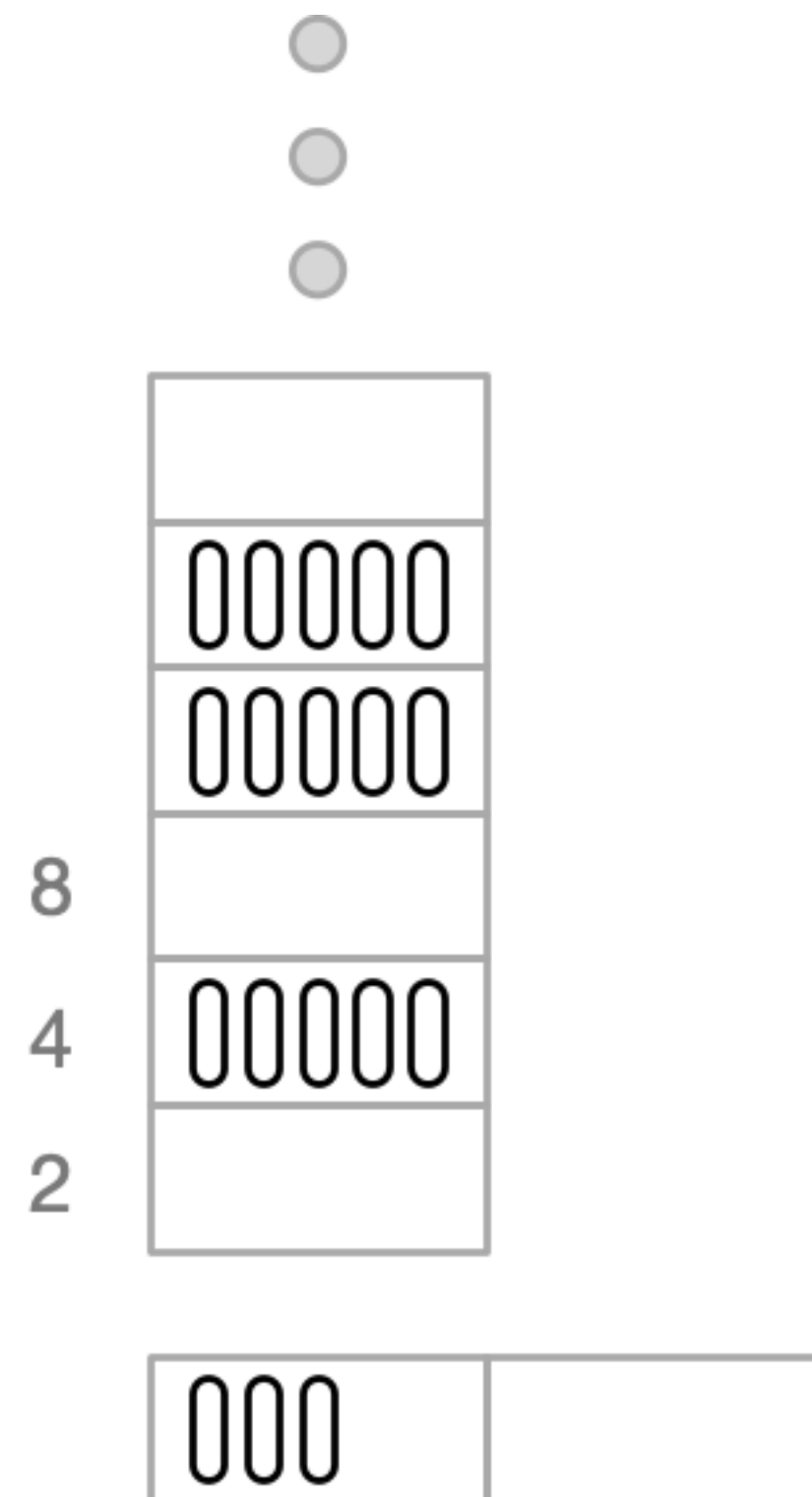
工作区填了2个128会上到第一级
工作区填了4个128才会上到第二级

就和知名游戏1024一样, 2 2得4,
要想得8, 还再再来个4 (再来2个2)



Doubles Sketch

Fig. 7



按照以上逻辑维护这样的数据结构

每个格子可能有或者没有 有效数据

格子级别不同, 权重不同

如何查询 Φ -quantile?



Doubles Sketch

	341				值	权重	rank
	496.24				217.83	64	0
	507.41				330.86	32	64
	470.68				332.19	8	96
	482.42				341	1	104
	468.1				341.12	16	105
	409.23	前16个是工作区			361.01	4	121
	515.64	里面有7个有效值			392.89	16	125
	545.43				394.29	8	141
	546.29	权重1			400.97	64	149
	570.25				407.33	32	213
	571.61				409.23	1	245
	575.82				430.29	16	246
	593.33				434.28	4	262
	642.41				439.31	64	266
	832.52				443.79	32	330
	434.28				459.79	8	362
	456.97				468.1	1	370
	501.64				470.68	1	371
	515.64	2			471.86	64	372
	545.43				475.15	16	436
	571.61				482.42	1	452
	593.33				485.51	4	453
	832.52				490.76	8	457
	361.01				496.24	1	465
	434.28				500	32	466
	485.51				503.49	64	498
	515.64	4			507.41	1	562
	546.29				515.24	8	563
	571.61				515.64	4	571
	593.33				523.45	32	575
	832.52				524.37	64	607
	332.19				525.04	16	671
	394.29				542.75	16	687
	459.79				546.29	4	703
	490.76	8			564.64	32	707
	515.24				569.43	64	739
	573.32				571.61	4	803

这个表格截图中：

- 左侧是999个学生分数灌入刚才那个结构之后的实际情况
- 为了方便观察, 这里的每个格子不是128, 改成了8
- 和刚才的图上下是反的, 这里上面16个是工作区
- 这整个左侧列表每行都有值, 然而有的是无效的, 只不过程序为了速度
- 右侧第一列是我将所有有效值拿出来, 从小到大排序, 一共47个
- 这些值原来所在的级别不同, 每个都有自己的权重
- 第三列则是这个值的可能的rank
- 一个值权重, 就是他在rank里站的坑数
- (或者说一个权重64的值, 是代表原来64个人站在这里的),
- 比如217的权重64, 排名0-63都是他.
- 下一个值330, rank只能从64开始, 他占32个坑.
- 由此把整个rank加了出来

排完rank, 想知道任意分位点的值, 很轻易找到

Doubles Sketch

- 空间复杂度还是 $\log n$ 级别的, 可以接受
- 可以merge, 两个这种结构要合并, 同级进行”排序抽样向上”操作即可
- 随机算法理解起来简单

其他算法

- **Q-Digest**
- **T-Digest**
- **KLL**
- **Moment Sketch**
- **Count-Min**

参考

- Emory.edu Courses584-StreamDB
- GK介绍博客
- GREENWALD, M. AND KHANNA, S. 2001. Space-efficient online computation of quantile summaries.
- Pankaj K. Agarwal, Graham Cormode, Zengfeng Huang, Jeff Phillips, Zhewei Wei, and Ke Yi. Mergeable summaries.
- DataSketches及源码



Q & A

Thanks