

# 20180403 定价算法更新

## 综述

因为 Artval 的定价功能进行了升级，加入了多倍投注等功能，旧的定价算法不再适用于新需求，本文是针对旧的定价与奖励算法的一些问题的更新。

## 引入多倍投注及其相关函数

旧的定价算法只考虑到单个用户只能投注单倍的情况，由于之后加入了单个用户多倍投注的情况，旧算法需要补足多倍投注部分。并且考虑到多倍投注对定价的影响，多倍投注需要调整定价权重。

设某用户投注倍数为  $B$ ，其定价权重为  $P$ ，奖励权重为  $R$ 。

### 投注倍数

投注倍数  $B$  即为用户定价时，购买倍数的份额。投注倍数  $B$  是定价权重  $P$  与奖励权重  $R$  的计算基础。

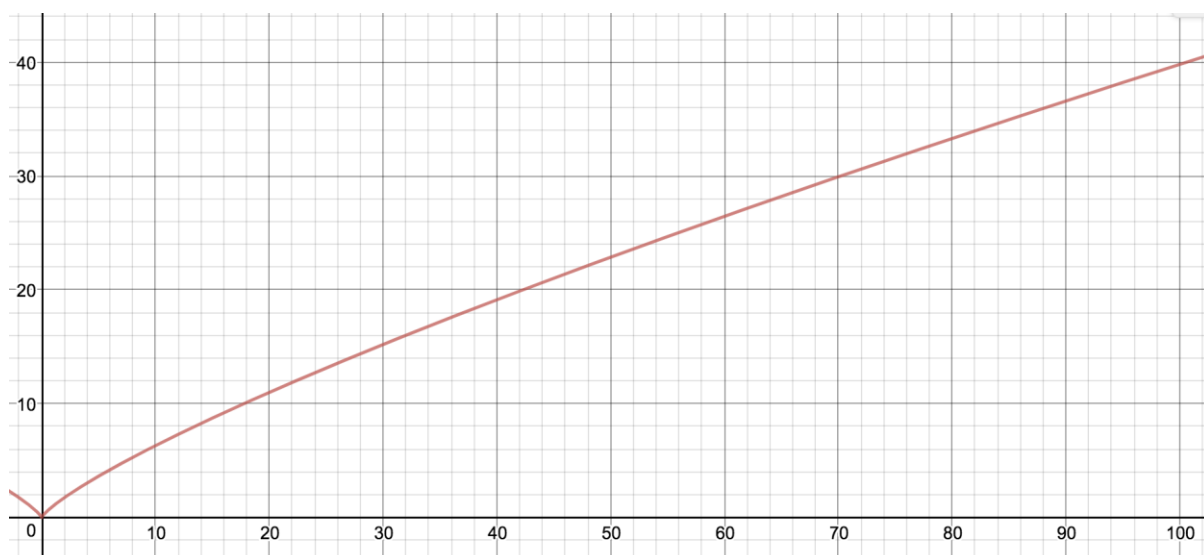
### 奖励权重

奖励权重  $R$  只与用户最终的奖励情况相关，不会对定价系统造成影响，同时参考实际生活中投注数量与获奖倍数相等的情况，奖励权重  $R$  应与投注倍数  $B$  相等，即  $R = B$ 。

### 定价权重

而定价权重  $P$  将直接影响定价系统，为了避免大量多倍投注对定价造成不正常的波动，需要通过算法降低大量多倍投注的影响权重，使得  $P$  只与投注倍数  $B$  正相关而不相等。

经过研究， $P = B^{0.8}$  被选为调整奖励权重的函数，该函数的图形：



该函数的若干特征值：

B	1	2	3	5	10	20	30	50	80	100
P	1	1.7411	2.4082	3.6239	6.3096	10.985	15.194	22.865	33.302	39.810

该函数不再是简单的  $P = B$  式的线性增长，而是保证了  $P$  的增量在  $B$  的增长中逐渐降低，使得单个大量投注用户的资金有效率降至大约 0.4 ~ 0.5 的区间，同时保证了小额投注的用户仍然有稳定的资金有效率。

因此，针对新增的多倍投注，将使用用户投注倍数  $B$ 、定价权重  $P$  以及奖励权重为  $R$  三个变量，同时有  $P = B^{0.8}$ ， $R = B$ 。

## 价格区间的确定

在一件艺术品的定价期完成后，平台将把期间收集的所有定价信息进行整合进算，提取出其中有效的价格区间以算出更为准确的最终定价。

### 价格区间算法一

算法一即现有的价格区间算法，它是一个使用平均值的系统：

- 最终价格确定

a) 将鉴宝人所投价序列 $Q$ 按从小到大排序:

$$\vec{Q} = \text{Sort}(V_1, V_2, V_3, \dots, V_n)$$

$\vec{Q}$ 为排序后的序列,  $V_{x, x \in \{1, 2, 3, \dots, n\}}$ 为鉴宝人所投价格

b) 从 $\vec{Q}$ 中选出:

最高值 $MAX(\vec{Q})$

最低值 $MIN(\vec{Q})$

c) 计算最高值和最低值价格区间, 并划分成为 7 个组:

区间最高值 (向上取整):  $Max = \lceil MAX(\vec{Q}) - MAX(\vec{Q}) \bmod 10 \rceil$

区间最低值 (向下取整):  $Min = \lfloor MIN(\vec{Q}) - MIN(\vec{Q}) \bmod 10 \rfloor$

小组区间跨度 (向下取整):  $S = \lfloor (Max - Min) \div 7 \rfloor$

小组区间:  $R_i = (Min + S \times (i - 1) + 1, (Min + S \times i)], i \in \{1, 2, 3, 4, 5, 6, 7\}$

d) 将价格序列 $Q$ 中落在 $R_1$ 和 $R_7$ 中的价格值舍弃, 获得最终有效价格序列 $Q'$ :

该算法从排序后的价格取出最低值以及最高值, 将它们相加, 再将结果除以 7 以计算出基于定价数值的 7 个价格区间, 从 7 个区间中除去代表最低值以及最高值的两个区间, 使用居中的 5 个作为最终有效价格区间。

该算法存在一个被极大值扰乱的可能。例如在一个人均定价接近 200 的数组中, 如果有一个恶意用户输入 100,000 的定价, 这套算法将使用 100,000 作为区间计算的最高值, 将它加上最低值然后除以 7, 所得的结果约为 14286, 并以此生成的 7 个以 14286 为大小的价格区间, 其中的最低值区间将为 [最低值 ~ 14286], 这将包括绝大多数估价结果。因为最低值区间将被算法舍弃, 所以绝大多数合理的估价结果将被视为无效, 这将使最终结果无意义。

## 价格区间算法二

将算法一进行改进, 在完成 a) 步骤数组重新排列后, 直接使用数组长度, 除以 7, 进行基于数组内元素数量, 基本等量的价格区间划分 (在数组长度无法被 7 整除的情况下, 调整最后一个区间的元素数量)。再将代

表最低值与最高值的两个区间舍弃，取出居中的 5 个区间作为最后的有效价格区间。

算法二能有效避免少数恶意用户利用极大值进行骚扰的可能，但是相对算法一，算法二对于大量恶意用户使用小数量定价进行骚扰的抵抗较弱，大量的小数额定价将使价格区间向低值移动，将降低计算出来的最终定价。

### 混合算法

在两种算法各有优势的情况下，将他们进行混合使用将是最好的做法。

首先使用算法二剔除极大值的影响，将重新排序后的数组按照长度分成 7 个区间，去除代表最高值的区间，将剩下的 6 个区间形成新的数组进行下一步运算。

对新数组使用算法一，使用平均值计算出 6 个区间，剔除最低值区间，最终留下的 5 个区间即为有效的价格区间。

### 最终定价的确定

因为多倍投注的加入，每个用户可能有不同的定价权重，他们对最终定价的影响不一样，因此已有的算法需要改进。

- 最终价格确定

a) 将鉴宝人所投价序列 $Q$ 格按从小到大排序:

$$\vec{Q} = Sort(V_1, V_2, V_3, \dots, V_n)$$

$\vec{Q}$ 为排序后的序列,  $V_{x, x \in \{1, 2, 3, \dots, n\}}$ 为鉴宝人所投价格

b) 从 $\vec{Q}$ 中选出:

最高值 $MAX(\vec{Q})$

最低值 $MIN(\vec{Q})$

c) 计算最高值和最低值价格区间, 并划分成为 7 个组:

区间最高值 (向上取整):  $Max = \lceil MAX(\vec{Q}) - MAX(\vec{Q}) \bmod 10 \rceil$

区间最低值 (向下取整):  $Min = \lfloor MIN(\vec{Q}) - MIN(\vec{Q}) \bmod 10 \rfloor$

小组区间跨度 (向下取整):  $S = \lfloor (Max - Min) \div 7 \rfloor$

小组区间:  $R_i = (Min + S \times (i - 1) + 1, (Min + S \times i)], i \in \{1, 2, 3, 4, 5, 6, 7\}$

d) 将价格序列 $Q$ 中落在 $R_1$ 和 $R_7$ 中的价格值舍弃, 获得最终有效价格序列 $Q'$ ;

e) 计算 $Q'$ 中价格的平均值:

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

其中 $n$ 为 $Q'$ 中价格值的个数,  $X_i$ 为 $Q'$ 中的第 $i$ 个值;

f) 取 $\bar{X}$ 为最终价格。

现有算法的做法是在获取有效区间后, 直接进行平均值计算。考虑加入权重, 可以将算式更新为:

$$\bar{X} = \frac{1}{p} \sum_{i=1}^n X_i P_i$$

$\bar{X}$  为最终定价,  $n$  为有效区间  $Q'$  中定价的个数,  $p$  为  $Q'$  中所有定价对应的所有估价权重的总和,  $X_i$  为  $Q'$  中的第  $i$  个值,  $P_i$  为  $X_i$  对应的权重。

如此即可将定价权重加入到最终定价的计算中, 使得最终定价被不同的定价权重值影响。

## 奖励分配算法

基于现有算法：

- a) 获取奖金的人数固定占有一定百分比（也就是获取奖金的人数确定）：

$$P = N * 15\%$$

其中 $P$ 是最终获得奖金的人数， $N$ 为鉴宝人总数，15%为获取奖金的比率；

- b) 鉴宝成功后，计算每个鉴宝人所投价格的权重：

$$W_i = \left(1 - \frac{|X_i - \bar{X}|}{\bar{X}}\right) \times 100$$

其中 $W_i$ 为第 $i$ 个鉴宝人所投价格的权重， $X_i$ 为第 $i$ 个鉴宝人所投价格；

- c) 将鉴宝人所投价格和所取价格 $\bar{X}$ ，按照从小到大排列，并把每个人所投价格权重标注在下方；

$$X_1, X_2, X_3, X_4, \dots, \bar{X}, \dots, X_{n-3}, X_{n-2}, X_{n-1}, X_n$$

---

$$W_1, W_2, W_3, W_4, \dots, 100, \dots, W_{n-3}, W_{n-2}, W_{n-1}, W_n,$$

（权重 100 是 $\bar{X}$ 本身所计算的值）

(1) 从中间值即 $\bar{X}$ 向两端进行探测，选择权重接近 100 的鉴宝人，直至选够人数 $P$ ；选满 $P$ 个人后，会继续向两端进行探测，检查是否有权重相同的人，如果权重相同则会继续选入候选列表；

(2) 对候选列表中权重相同的人按照其投票的时间戳进行排序，投票早的最终被选入获奖人列表；

- d) 最终在获奖人列表中，按照权重比例发放奖金：

$$M_i = \frac{W_i}{\sum_{i=1}^n W_i} \times B$$

其中 $B$ 为总奖金数， $M_i$ 为第 $i$ 个人所得奖金数。

由于需要添加用户的奖励倍数  $R$ ，在完成 c) 步骤选取获奖人列表后，新建算式，加入  $X_i$  对应的奖励权重  $R_i$  进行运算，此时需要将最终 15% 人数获奖区间内的  $X_i$  的  $R_i$  与  $W_i$  进行分别计算与统计：

$$T_i = \frac{R_i}{\sum_{i=1}^n R_i} * \frac{W_i}{\sum_{i=1}^n W_i}$$

其中  $T_i$  为  $X_i$  对应的奖励系数，通过将奖励权重  $R_i$  与所投定价权重  $W_i$  相乘获得。

$$M_i = \frac{T_i}{\sum_{i=1}^n T_i} * B$$

其中  $M_i$  为第  $i$  个用户所得奖金数目， $B$  为总奖金。此时可以获得  $X_i$  对应的奖励。

## 结论

通过更新算法，定价系统将兼容多倍投注系统，并且避免了单一用户大量投注对定价造成很大的影响。同时通过更新一些细节，使算法能够更稳定地筛选有效价格区间，并使得最终定价结果更为准确。