

浅析顺丰快递收费标准与探究

徐润麒

[引入]

2018 年快递业务突破年累计共 500 亿件,标志着当今快递业务在现代生活中所占据极为重要的部分.快递行业的收费标准如何?现以顺丰快递为例进行探究.

专有名词解释:

首重:类似于计程车的“起步费”寄件必须支付的费用

续重:寄件重量超过 1kg 的部分每 kg 支付的价格

$[x]$ 表示 x 的整数部分(不大于 x 的最大整数)

例: $[3.8]=[3.4]=3$

$\{x\}$ 表示 x 的小数部分($x-[x]$)

例: $\{3.8\}=0.8$ $\{3.4\}=0.4$

[过程]

选择“上海→山东”,规定重量 $x(\text{kg})$,费用 $y(\text{元})$

1. 顺丰次日

100kg 以内:

不超过 1kg

$$y=22$$

超过 1kg 且小数部分 ≤ 0.5 $y=22+([x]+0.5-1)*10=17+10[x]$

超过 1kg 且小数部分 > 0.5 $y=22+([x]+1-1)*10=22+10[x]$

100kg 以上

小数部分 < 0.5

$$y=22+([x]-1)*10=12+10[x]$$

小数部分 ≥ 0.5

$$y=22+([x]+1-1)*10=22+10[x]$$

综合:

$$y = \begin{cases} 22 & x < 1 \\ 17 + 10[x] & 1 < x < 100, \{x\} \leq 0.5 \\ 22 + 10[x] & 1 < x < 100, \{x\} > 0.5 \\ 12 + 10[x] & x \geq 100, \{x\} < 0.5 \\ 22 + 10[x] & x \geq 100, \{x\} \geq 0.5 \end{cases}$$

2. 顺丰隔日

-> 区别: 四舍五入的标准改为了 $\geq 30\text{kg}$

同样, 规定 $x(\text{kg})$ 的物品, 费用 $y(\text{元})$

$$y = \begin{cases} 18 & x < 1 \\ 15.5 + 5[x] & 1 < x < 30, \{x\} \leq 0.5 \\ 18 + 5[x] & 1 < x < 30, \{x\} > 0.5 \\ 5[x] & x \geq 30, \{x\} < 0.5 \\ 5[x] + 5 & x \geq 30, \{x\} \geq 0.5 \end{cases}$$

3. 通过总结, 可以得到关于重量->价格的通式

-> 设首重 a 元, 续重 b 元/kg

则有:

$$y = \begin{cases} a & x < 1 \\ b - 0.5a + b[x] & 1 < x < 100, \{x\} \leq 0.5 \\ a + b[x] & 1 < x < 100, \{x\} > 0.5 \\ b - a + b[x] & x \geq 100, \{x\} < 0.5 \\ a + b[x] & x \geq 100, \{x\} \geq 0.5 \end{cases}$$

程序(cpp)

```
SFKD.cpp
1 //made by MX-R, copywrite by MX-R.
2 //NO COPY WITHOUT PERMISSION!
3 //only for studying and learning.
4 //coding is a kind of art.
5 //99 little bugs in the code.
6 //99 bugs in the code.
7 //fix one bug, compile it again.
8 //101 little bugs in the code.
9 //101 little bugs in the code...
10 //(Repeat until BUGS = 0)
11 /* Porgram about SF's parcel */
12 #include <bits/stdc++.h>
13 using namespace std;
14 double x,y;
15 long long a,b;
16 double strange_solve(double o){
17     if(o<100) if(o-floor(o)<=0.5) return floor(o)+0.5; //设置
18     else return ceil(o); //ceil:上取整
19     return round(o); //round:四舍五入
20 }
21 int main(){
22     cout<<"输入首重(kg/元)·",cin>>a,cout<<"输入续重(kg/元)·",cin>>b,cout<<"输入物体重量(kg)·",cin>>x;
23     x=strange_solve(x); //按照顺丰的方式计费
24     cout<<"a+(x-1)*b<<"元"; //按照顺丰的方式计费
25     return 0;
26 }
```

仔细观察可得: $a+b[x]$ 有可能小于等于 $b-a+b[x]$

->当 $x=99.6$ 到 100.4 时,

$y=1012$ (在 0.8kg 的差值内相等)

\therefore 其他的表达式间的差都小于 $|(a+b[x])-(b-a+b[x])|$

\therefore 易证可得: **不会出现重的物品所要支付的寄件费用比轻的物品少的情况**

->就这么结束了? 😐 😐

假设有人寄了 100Kg 棉花..... 😐

->引入轻抛系数

定义函数(封装,把计重和计价分开模块化处理)

$$\text{strange_solve}(x) = \begin{cases} 1 & 0 < x < 1 \\ [x] + 0.5 & 1 < x < 100, \{x\} \leq 0.5 \\ [x] + 1 & 1 < x < 100, \{x\} > 0.5 \\ [x] & x \geq 100, \{x\} < 0.5 \\ [x] + 1 & 1 < x < 100, \{x\} \geq 0.5 \end{cases}$$

$$v(x) = (l[\text{长}] * w[\text{宽}] * h[\text{高}]) \div pq[\text{轻抛系数}]$$

随后在实际重量 `strange_solve(x)` 和体积重量 $v(x)$ 中
取最大值照旧计算即可

```

SFKD.cpp
7 //fix one bug, compile it again.
8 //101 little bugs in the code.
9 //101 little bugs in the code...
10 //(Repeat until BUGS = 0)
11 /* Program about SF's parcel */
12 #include <bits/stdc++.h>
13 using namespace std;
14 double x, y, qp=1, v; char c; //qp=1: 27 斤不能除以 0
15 long long a, b, l, w, h;
16 double strange_solve(double o){ //计算
17     if(a<=1) return 1;
18     if(o<100) if(o-floor(o)<=0.5) return floor(o)+0.5; //floor: 下取整
19     else return ceil(o); //ceil: 上取整
20     return round(o); //round: 四舍五入
21 }
22 int main(){
23     cout<<"输入首重(kg/元):", cin >> a, cout<<"输入续重(kg/元):", cin >> b, cout<<"输入物体重量(kg):", cin >> x;
24     cout<<"计算体积重量吗输入(Y/N)", cin >> c; if(c=='Y'){
25         cout<<"输入长(m)", cin >> l, cout<<"输入宽(m)", cin >> w, cout<<"输入高(m)", cin >> h;
26         cout<<"输入轻抛系数", cin >> qp;
27         x=strange_solve(x); //按照顺丰的方式计算
28         v=(l*w*h)/qp; //计算体积重量
29         x=max(v, x); //取体积重量, 实际重量最大值
30         cout << a+(x-1)*b << "元"; //按照顺丰的方式计价
31         getchar();
32         return 0;
33     }

```

就这么结束了? ->还没有

[设想]

按照接近于以重量位自变量的正比例单调函数计价的方式真的合适吗??

当然不合适.原因:各种交通工具的硬限制(必须满足)是

体积,而不是重量<-(软限制,小超载没关系)

而且重量,体积与几个的关系不能只使用一个一次正比例函数来进行描述:送一个 1kg,0.1m*0.1m*0.1m 的包裹

与 送一个 2kg,0.2m*0.2m*0.1m 包裹

的实际价格差距小于上述计价方式所产生的价格差值.

所以:提出一种"较为理想"的新快递计价方式:

规定 x 为实际重量(不经过顺丰的计重处理)

与体积重量(经过顺丰的轻抛系数处理)

的最大值

规定 $x(\text{kg})$ 为重量 $y(\text{元})$ 为价格

L 为始发地到终点的路程(Km),

T 为寄件后到达时间(h)

运用 **ML** 算法优化参数,不断逼近用重量根据“价目表”计算得出的价格(具体过程省略,详见 **GitHub** 后期代码,目前只针对山东地区进行了计算)

$y=L*1.2/(T^2/9.7)*[(x/30.0)]$ $[]$:取整数部分

为什么要给 T 平方?-运用缩放法,扩大时间维度在函数中所占影响比例的大小(送达时间与价格不可用一次函数描述)

为什么要给 x 除以 30?-区分梯度(以 30kg 为梯度,计量重量,令计价方式更为合理)

以上的计算方法还未来得及考虑体积,欢迎提出 **Issues** 或给出意见.

关于本文档的所有资料(程序,文件...etc)都可在[这里](#)查看并下载
单击“Clone or Download”->“Download ZIP”即可下载全部文件

徐润麒

2019.6.6