公式推导以及改进意见

编译日期: February 17, 2020

彭淏

1700010601@pku.edu.cn

1 公式推导 **1**

2 平均投点数 **3**

3 改进意见 **3**

Part 1 公式推导

First of all, according to $<$ **教学手册** $>$, the electing system obeys the weighted lottery model. Now we make a crude assumption that all other person put the same point on a particular course.

Assume a course containing n_1 students is preselected by n_2 students. You placed X points and the others placed m points, then the probability you dropped is

$$\frac{(n_2-1)m}{(n_2-1)m+X}\cdot\frac{(n-2-2)m}{(n_2-2)m+X}\cdots\frac{(n_2-n_1)m}{(n_2-n_1)m+X}.$$

Set $x=\frac{X}{m}$, the expression equals:

$$\frac{(n_2-1)!(n_2-n_1+x)!}{(n_2-n_1)!(n_2+x-1)!}.$$

Using Stirling's approximation:

$$n!\sim\sqrt{2\pi n}\cdot\left(\frac{n}{e}\right)^n,$$

it is:

$$\frac{n_2^{n_2+0.5}\cdot(n_2-n_1+x)^{n_2-n_1+x+0.5}}{(n_2-n_1)^{n_2-n_1+0.5}\cdot(n_2+x)^{n_2+x+0.5}}$$
$$=\frac{(1+\frac{x}{n_2-n_1})^{n_2-n_1+0.45x}}{(1+\frac{x}{n_2})^{n_2+0.45x}}\cdot\left\{(1-\frac{n_1}{n_2+x})^{0.55x+0.5}(\frac{n_2}{n_2-n_1})^{0.5-0.45x}\right\}$$

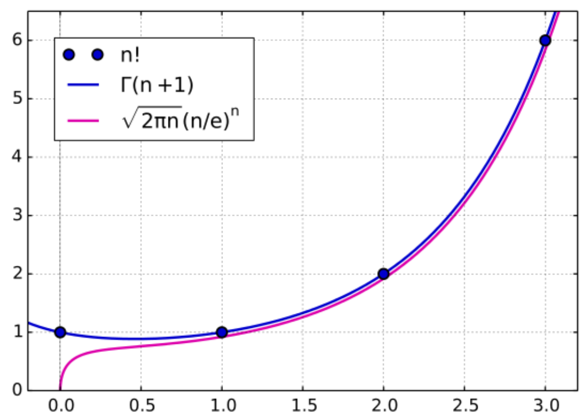

Figure 1: $\Gamma(n)$

Now under assumptions that $n_2 - n_1 \geq 5, 1.02 \leq \frac{n_2}{n_1} \leq 9$ and $0 \leq x \leq 4.5$:

$$0.985 \leq \frac{(1 + \frac{x}{n_2 - n_1})^{n_2 - n_1 + 0.45x}}{(1 + \frac{x}{n_2})^{n_2 + 0.45x}} \leq 1.015.$$

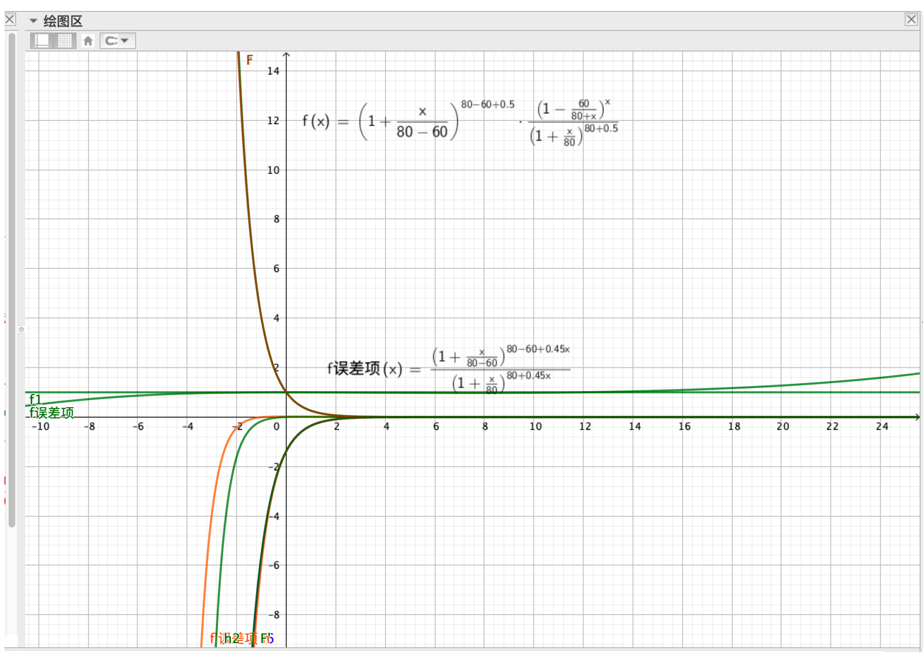

Figure 2: 误差项

So we will now only consider the last component.

So we use adjustment method, because the function is convex, the optimal solution is guaranteed.

Part 2 平均投点数

This is the hard part. Because I don't have any data, I can only guess how others put points. In the first verison, I assumed that people placed more points on a course with more fierce competition, but yesterday I opened a anonymous topic on < **北大树洞** > and collected about 180 data, then I run the program with the data and match the average points placed any run again and again, then noticed that the average points place does'n vary much, only on the interval of about $[20, 35]$.

I noticed that students who only have one class to put points won't came and use my program, so I made another guess that a class has 8% students who placed 99 points. so the average ranges from 26 to 40.

I assumed that the average points only have to do with the ratio, and I find the rato too aggregated at 1. So I made a transformation $x \mapsto (x-1)^{\frac{1}{8}} + 1$ and used polyfit in Matlab to get a quadratic curve:

$$f(x) = -101x^2 + 392.6x - 347.8$$

.

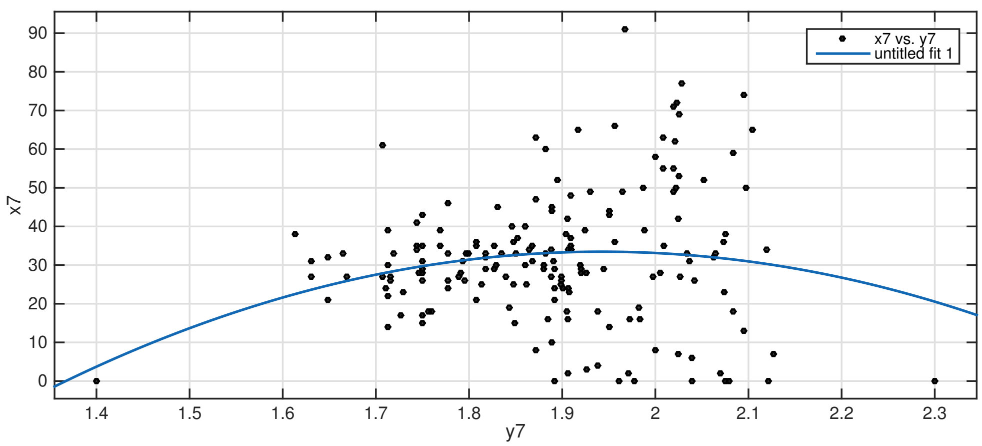

Figure 3: polyfit with degree 2 of abscissa $(x-1)^{\frac{1}{8}} + 1$

Part 3 改进意见

Like I said, I made many assumptions, any of them might be violated.

The first is the assumption that all others placed the same points, which is absurd, and I do not know how this affects the result. But I know neither the distribution nor how to calculate the probability at that case, so I hope someone, if find this code helpful and interesting, can delve in deeper and use maybe tools from statistics to improve this. (I major in pure math, I'm poor in programming, and I do not know about statistics).

The second is to gain more data. I dreamed of someday dean will show alongside each course a number of how many total points are put. It will be perfect. But for now, I hope there is someway to gain some data legally, and help know the distribution of points put.

My purpose to write this is that I will soon graduate so I'm planning to let go of this project. In fact I'm surprised by the fact that there's no one before that have an idea to calculate the optimal point-putting strategy. I hope someday maybe many one will use this code (or advanced version of it), just like we use the course browsing tool.