黄金点猜数游戏

游戏规则

1) 假设有 M 个玩家，P1,P2,…Pm
2) 在 (0-100) 开区间内，所有玩家自由选择两个正有理数数字提交（可以相同或者不同）给服务器，假设 N11,N12,N21,N22,Nm1,Nm2
3) 等 M*2 个数字都提交后，服务器做如下计算：
   (N11+N12+N21+N22+…+Nm1+Nm2)/(M*2)*0.618 = GN
   由此得到黄金点数字 GN
4) 查看所有玩家提交的数字与 GN 的算术差的绝对值，值最小者得分，值最大者扣分。其它玩家不得分
5) 此轮结束，进行第二轮，多轮（100 或更多）后，累计得分高者获胜。如最后得分相同者由组委会裁决最终名次（比如是否采用了微软工具/技术，程序运行效率等等）。

计分规则

1）M 个玩家比赛时，每轮离 GN 最近的玩家得 M 分，最远的扣 2 分，其它玩家不得分
2）如果一个玩家在一轮内提交两个相同的数字并得分时，只计一次分
3）多个玩家在一轮内同时离 GN 最近时，每个玩家都得 M 分

程序规则

1）玩家编写程序，每次运行只提交一轮数字，然后退出。以 EXE 的形式拷贝到服务器上。有病毒者取消比赛资格，不得恶意攻击服务器。
2）强烈建议用微软 Visual Studio 2017 完成程序，用 C++/C#等语言可以直接编译成可执行的 EXE。
3）如果运行该 EXE 需要额外的库支持，需要把支持库打包也一并提交。不建议使用 Java 等需要下载额外大量支持库的编程语言。如果是 Python 等脚本语言，需要直接打包成 EXE。服务器未配置 GPU，如有调用计算库，请使用 CPU 版本。
4）可以使用第三方算法或者模型，但要注明出处，注意 open source 等版权问题。
5）猜一轮数字的程序代码必须在 5 秒之内运行完成，然后自动关闭。
6）组委会提供一台服务器，所有程序拷贝并运行在该服务器上。玩家不得与该服务器交互。
7）服务器上的调度程序并行运行玩家的程序，输入为所有前 Y 轮的历史数据（包括其他玩家的数字和 G 值），输出为本轮猜测的两个最新数字。使用标准输入输出与玩家程序进行交互。

a. 输入格式：

格式举例：

2 5

18.07 30 30 17 40

24.87 18.08 18.08 99.9 25

格式说明：

Bot 程序需从标准输入读取数据

第一行两个数表示下面有两行数据，每行数据有 5 个值

后面每行代表一轮比赛的数据，最后一行是最新一轮的数据

行中第一个数据是该轮的黄金点值，后面第 2N 个数和第 2N+1 个数是第 N 个 bot 输出的预测值，数据之间以制表符分隔

如果某个值为 0，表示该 bot 在该轮超时未提交数据或输出数据非法，不在(0,100)之间

b. 输出格式：

输出（0, 100）中的两个数，以制表符\t 分割。格式非法或 5 秒内没有输出，即认为此玩家放弃本轮比赛，只在剩余玩家中计算 G 值

格式举例：

12.34 56.78

8）Bot 如果需要知道自己上一轮提交的值，或者需要多轮之间共享数据，可以通过本地文件存储中转数据，数据文件只可以放在 Bot 同级目录下，且大小不得超过 100M。

9）服务器实时计算本轮运行结果并在大屏幕上显示最新结果（G 值，得分，排名）。

10） 组委会事先提供一些历史模拟数据，供玩家参考，决定如何编写代码/模型。

11） 先进行上半场游戏，然后玩家可以调整程序策略，再次提交后，进行下半场游戏。按 30%+70%的得分权重比来计算最终胜者。

“潜”规则

1）建议使用微软工具/技术/编程语言

2） 可以采用的算法/技术（仅供参考，对这个场景还要具体问题具体分析）

a. 随机

- 轮盘赌/锦标赛/精英保留策略等等

b. 状态机

- 相当于博弈论的具体实现，但是随着参赛玩家数量的增减而复杂度增加

c. 决策树
    - 根据前 N 轮的历史预测下一轮其他玩家的行为，从而制定自己的策略

d. 遗传算法
    - 根据自己的前 N 轮的成绩和 G 值的变化函数，筛选出下一代数字

e. 神经网络
    - 用传统神经网络搭建一个模型，用历史数据训练权重，预测输出的数字

f. AI 模型
    - 用现代神经网络搭建模型，把一维数字空间转换为二维矩阵空间来试图解决问题