Exploración matemática: Mastermind, una breve exploración al juego.

André Robles Bueckmann
A01706832
Grupo 76

Mastermind es un juego de mesa de dos jugadores diseñado por Mordecai Meirowitz en 1970. El juego consiste en que el *jugador*$_1$ crea un código a partir de pines de 6 colores diferentes que coloca en 4 espacios (el juego no está limitado a estas condiciones, sino las que se utilizarán en esta exploración por su familiaridad). De esta manera, el *jugador*$_2$ trata de descifrar el código recibiendo retroalimentación en cada turno que el *jugador*$_1$ le proporciona en forma de pines blancos y negros (para uso de esta exploración). Los pines negros representarán entonces que el jugador$_2$ acertó un color del código en la posición correspondiente (sin saber cuál de los 4 espacios es); por el otro lado, la retroalimentación en forma de un pin blanco significa que el jugador únicamente acertó el color del código. Las anteriores condiciones vuelven a Mastermind un juego muy interesante de jugar y que personalmente captó mi atención, sobre todo por el razonamiento que puede llegar a existir detrás de un juego aparentemente sencillo.

El objetivo inicial de la exploración consistía en descubrir la mejor estrategia de jugadas o turnos necesarios para garantizar una victoria en la variante de 6 colores y 4 espacios. Para conseguir esto, se necesita iniciar por averiguar el tamaño de la muestra con la que se requiere lidiar, lo cual se logra multiplicando *L* veces la cantidad *C* de colores.

$$C \times C \times C \ldots \times C = C^L$$

Si tomamos en consideración una longitud de código de 4 espacios $(L = 4)$ con 6 posibles colores $(C = 6)$ obtenemos que el total de permutaciones (con repetición) posibles sería:

$$6 \times 6 \times 6 \times 6 = 6^4 = 1296.$$

Esta cantidad de combinaciones ya representa un reto si se pretende explorar la cantidad de turnos mínimos que se requiere para resolver cualquier código. Ahora bien, si el objetivo es saber en qué turno la probabilidad de acertar es del 100%, debemos encontrar el número total de caminos que tiene el jugador2 de seguir un camino para adivinar el código. El problema con esto es que crece en un orden factorial, puesto que, si en el primer turno la posibilidad de un código es de 1/1296, entonces se podría suponer que la probabilidad de seguir un camino cualquiera de dos turnos es de $\frac{1}{1296} \times \frac{1}{1295}$, y así sucesivamente. De hacer $1296!$, una calculadora ni siquiera es capaz de arrojar el orden de magnitud del número de caminos que podría tomar el juego. Pero esto es solo una condición independiente que cambia por completo cuando añades limitantes como el número máximo de turnos (10), y las formas de retroalimentación del juego (*FR*). Para calcular el número de *FR* se debe entender que existen 4 posiciones en las que el jugador1 le da retroalimentación al jugador2, y existen 3 formas diferentes de retroalimentación: no acertar nada (no hay pin vinculado), acertar el color (el pin blanco), acertar el color y la posición (el pin negro). Para saber la cantidad de *FR* debemos calcular el número de combinaciones (con repetición) de las *n* posiciones en *m* fases, dada por la siguiente fórmula:

$$CR_m^n = \frac{(m+n-1)!}{n! \times (m-1)!}$$

$$CR_3^4 = \frac{(3+4-1)!}{4! \times (3-1)!} = \frac{6!}{4! \times 2!} = \frac{6 \times 5 \times 4 \times 3 \times 2 \times 1}{4 \times 3 \times 2 \times 1 \times 2 \times 1} = \frac{6 \times 5}{2} = \frac{30}{2} = 15$$

Es decir, tenemos 15 *FR* para cada permutación de código.

Con lo anterior establecido, es necesario crear una conexión en donde el código del turno 1 genere un FR que reduzca la cantidad de posibles combinaciones para el turno 2, y que este a su vez reciba retroalimentación que elimine posibles cantidades de códigos para el turno 3. Esto sin embargo requiere de un algoritmo complejo que precisa evaluar todas las posibilidades de código por separado, para posteriormente unirlos y establecer las condiciones en donde el *FR* de los turnos 1, 2 y 3 se cumpla. Por ejemplo:

> Tenemos un código desconocido (x_1, x_2, x_3, x_4), y sugerimos como código en el turno1 $(1,2,3,3)$, (se atribuye un número del 1 al 6 a cada color). Supongamos que para esto hay un FR vacío. Esto quiere decir que no hemos acertado ninguno de los colores, y por ende solo quedan 3 colores restantes (el color 4,5 y 6). Así, en el siguiente turno se propone la combinación $(4,4,5,5)$ y el *FR* correspondiente es un pin negro y uno blanco. Quiere decir que tenemos un color que adivinamos la posición, y uno segundo que solamente seleccionamos el color. Pero en este momento nos es todavía irrelevante el haber obtenido un pin negro, puesto que no tenemos la información suficiente para saber de cual color estamos hablando. Esto implica que en algún turno X posterior sea indispensable combinar las condiciones de cada *FR.* Es así que, al menos para la persona promedio, le es imposible evaluar todas las posibilidades por cada turno, y crear vínculos que minimicen el número de turnos requeridos para ganar.

Por la anterior razón, el cálculo de los turnos necesarios para ganar es cambiante, y dependiente del camino que se escoja: imposible de realizar de manera manual. El algoritmo del que se hizo referencia anteriormente es el creado por Donald Knuth, un profesor de estadística de la Universidad de Stanford que en 1977 consiguió garantizar éxito para descubrir cualquier código del juego en 5 turnos o menos. Como mencionamos, esto no es posible manualmente, pero si nos puede dar un punto de referencia de la mejor estrategia inicial que si es posible comprobar. El profesor Knuth establece que en el juego de Mastermind la mejor primera jugada es elegir un conjunto de forma $(1,1,2,2)$, lo que se traduciría a comenzar con una combinación bicolor con dos pines de cada uno.

(Comprobación matemática de la mejor primera jugada a partir del potencial que tiene con cada uno de los FR de eliminar posibilidades de código).

Existen dos formas de premiar que tan cercano está una combinación del cógido final (la primera propia la segunda externa).

Forma 1:

o = no le atinaste a nada.
x = le atinaste al color y posición.
% = le atinaste al color pero no la posición.

(Este es el cálculo del porcentaje de acierto **independiente**):

- hay 4 hoyos, cada uno representa el 25% del código).
- tener un “%” representa un porcentaje de acierto de “4.16% periódico”.

combinación	porcentaje
X x x x	100%
O x x x	75%
o o x x	50%
o o o x	25%
o o o o	0%
% % % %	16.64%
% % % o	12.48%
% % o o	8.32%
% o o o	4.16%
X x x %	79.16%
X x % %	58.32%
% % % x	37.48%
% o o x	29.16%
% o x x	54.16%
% % o x	33.32%

Forma 2:

Esta es la función fitness de Kallister, que a diferencia de los porcentajes anteriores, premia a la combinación por la cantidad de aciertos a partir de:

$$f(\mathbf{x}_i) = (2B + W) + \sum_{k=1}^{N-1} k$$

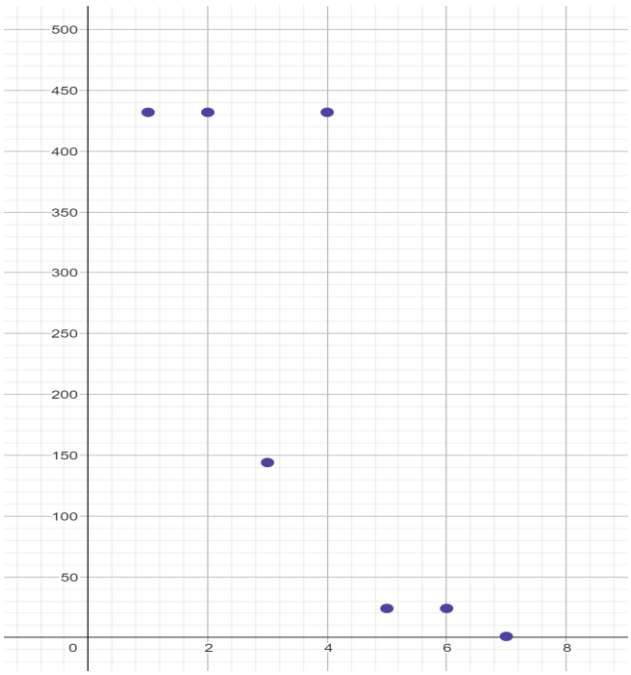
Ahora bien, la estrategia e Knuth continúa por eliminar posibles códigos haciendo uso de su algoritmo, entonces a partir de aquí la comprobación de la efectividad de las estrategias es meramente empírica y frecuentística.

Estrategia 1: Se debe permanecer con la conducta de suposiciones bicolor hasta haber completado todos los colores. (Esto con el objetivo de eliminar la variabilidad de los colores lo antes posible y continuar por su posición).

Estrategia 2: Se debe permanecer con la conducta de suposiciones bicolor hasta el turno 2, pues en el turno 3 se jugará una combinación monocolor para descartar los últimos dos colores, y proseguir por averiguar la posición de cada uno.

Hipótesis final: Al jugar, si no se comete un error (a lo que definiremos como volver a preguntar una condición, con más profundidad más adelante), la tendencia de la cantidad de posibilidades que tiene de acertar un código en cualquiera de las dos estrategias: disminuye, sube, y baja hasta ganar.

Por ejemplo (1 ejemplo no es la muestra completa), la gráfica tiene el número de turnos en el eje x y el número de posibilidades independiente en el eje y:



Referencias:

Nelson, T (2000). *A Brief History of the Master Mind Board Game*. WayBack machine.

https://web.archive.org/web/20150906044819/http://www.tnelson.demon.co.uk/mastermind/history.html

Mastermind Solver on dCode.fr [online website], retrieved on 2023-01-21,

https://www.dcode.fr/mastermind-solver

Cabana, L., & Túpac, Y. (2014). *Cómo resolver el juego de Mastermind a través de una computación evolutiva.* [Artículo divulgativo]. Universidad Católica de San Pablo Arequipa.

Knuth, D. (1975). *The computer as mastermind* [Artículo divulgativo]. Universidad de Stanford.

Notas extra, no es parte de la actividad:

Ahora bien, si quisiéramos analizar todo…tendríamos que recurrir a una cantidad impresionante de posibles caminos..que solo se logran con un algoritmo…este experimento lo hizo Knuth…

Yo voy a estudiar solo las primeras dos turnos…análisis frecuentístico vs clásico.

Preguntas a responder:
¿Cómo se relaciona el feedback con las jugadas?
¿Cómo se desempeñan las estrategias de mastermind que elegí analizar?
Iteración= repetición de un proceso.