

Juegos de azar

TAdP - 1C 2021 - Trabajo Práctico Grupal Objeto/Funcional



Introducción

IMPORTANTE: Este trabajo práctico debe implementarse de manera que se apliquen los principios del paradigma híbrido objeto-funcional enseñados en clase. No alcanza con hacer que el código funcione en objetos, hay que aprovechar las herramientas funcionales, poder justificar las decisiones de diseño y elegir el modo y lugar para usar conceptos de un paradigma u otro.

Se tendrán en cuenta para la corrección los siguientes aspectos:

- Uso de Inmutabilidad vs. Mutabilidad.
- Uso de Polimorfismo paramétrico (Pattern Matching) vs. Polimorfismo Ad-Hoc.
- Aprovechamiento del polimorfismo entre objetos y funciones y Orden Superior.
- Uso adecuado del sistema de tipos.
- Manejo de herramientas funcionales.
- Cualidades de Software.
- Diseño de interfaces y elección de tipos.

Jugadas y Apuestas

Los jugadores que vamos a simular en nuestro sistema van a apostar plata en juegos de azar y, dependiendo del resultado de estos, van a obtener una ganancia o van a perder todo lo que hayan apostado.

En una **jugada** se apuesta un monto de dinero y según cuál sea el resultado de un juego va a devolver el monto modificado. Por ejemplo, si nuestro juego fuese algo tan simple como tirar una moneda a cara o cruz, una jugada podría ser jugar a que si sale cara se duplica la apuesta y si no se pierde; en ese caso si utilizo la jugada “jugar a duplicar si sale cara” poniendo \$20 y el resultado del juego es cara, obtengo \$40, mientras que si uso la misma jugada con el mismo monto pero el resultado del juego es seca, obtengo \$0, es decir, pierdo lo que aposté.

Una **apuesta** es una jugada que ya tiene la información de cuál es el monto a apostar, o sea que teniendo una apuesta sólo falta el resultado del juego como para conocer la ganancia (o no) del jugador.

En principio, queremos modelar dos juegos: cara o cruz y la ruleta:

En cara y cruz hay solo dos resultados de juego, Cara y Cruz, y queremos modelar las siguientes dos jugadas:

- Jugar a cara: que si sale cara duplica el monto apostado, si no se pierde.
- Jugar a cruz: que si sale cruz duplica y si no se pierde todo.

En la ruleta, que es un juego más complejo, los resultados de juego posibles se representan por la combinación de un número entre 1 y 36 y un color que puede ser rojo o negro, y por el 0, que no tiene color. Se puede usar como referencia para los 37 resultados posibles el siguiente cuadro:

Conociendo esto, queremos modelar las siguientes **jugadas** para la ruleta:

0		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	2a1	2a1	2a1
Primeros 12				Segundos 12												Terceros 12																								
1 a 18		PARES						IMPARES		19 a 36																														

- Jugar al rojo: duplica el monto apostado si sale rojo, si no se pierde el mismo.
- Jugar al negro: duplica el monto apostado si sale negro, si no se pierde el mismo.
- Jugar a un número: si sale el número elegido, el monto apostado se multiplica por 36, si no se pierde.

- Jugar a par: duplica el monto si sale par, se pierde si sale impar. (el 0 no cuenta como par ni impar)
- Jugar a impar: duplica el monto si sale impar, se pierde si sale par.
- Jugar a una docena: si sale algún número dentro de la docena elegida, el monto se triplica, si no se pierde. Las docenas son la 1era (1 a 12), 2da (13 a 24) y 3era (25 a 36)

Bien!, con esto ya podemos modelar jugadas y apuestas sobre ambos juegos. Sin embargo, cuando se apuesta en juegos como la ruleta no se suele hacer sólo una apuesta por cada vez que giran la ruleta, si no que se combinan varias apuestas, como por ej: jugarle \$25 al rojo, \$10 a la segunda docena y \$30 al número 23. ¿Cómo se resolvería eso? Digamos que sale el 3, que es rojo, en ese caso por la primera apuesta ganamos \$50, por la segunda nada y por la tercera nada, entonces quedamos con una ganancia de \$50 (habíamos apostado \$65 así que salimos perdiendo :(). En cambio, si sale el 14, que es rojo y está en la segunda docena, ganamos \$50 por la primera apuesta, \$30 por la segunda y nada por la tercera, dando un total de \$80. Finalmente, si sale el 23, que es rojo y está en la segunda docena, ganamos $\$50 + \$30 + \$1080 = \1160 !

Queremos permitir este tipo de apuestas combinadas, donde lo que se obtiene como resultado al conocer el resultado del juego es la suma de lo que hubiesen dado las apuestas individuales ante ese resultado de juego.

Resultados de los Juegos

Hasta ahora hablamos de cómo actúan las apuestas ante resultados de juego dados, pero aún nunca dijimos cómo obtenemos dichos resultados.

Un juego de azar, por definición, no tiene un resultado determinístico. Cada vez que jugamos puede salir diferente. Además, en nuestro sistema no nos va a interesar simular un caso particular si no que queremos conocer todos los posibles desenlaces del juego.

Para esto vamos a introducir algunos conceptos de probabilidad:

Suceso

Un suceso es cada valor posible que puede tomar un evento aleatorio. En el caso de los juegos, cada resultado posible que puede tomar. Por ejemplo, si hablamos de tirar una moneda, los sucesos posibles son Cara y Cruz.

Los juegos que modelemos siempre van a tener uno o más sucesos como resultado.

Distribución de probabilidad

Cuando tiramos una moneda, existe la misma chance de obtener el suceso Cara como de obtener el suceso Cruz, entonces, podemos decir que la probabilidad de cada suceso es 50%. Si nuestra moneda estuviese cargada que siempre da Cara, la probabilidad del evento Cara es 100%, y la de Cruz es 0%, que es lo mismo que que no sea un suceso posible de tirar la moneda cargada.

La distribución de probabilidad es la manera de representar el conjunto de sucesos con sus probabilidades asociadas. Entonces, todos los resultados del juego se pueden representar con una distribución de probabilidad que contenga todos los sucesos posibles con sus probabilidades.

La suma de probabilidades de los sucesos de una distribución siempre tiene que ser 100%.

De una distribución de probabilidad queremos poder saber:

- sus sucesos posibles, que sería el conjunto de sucesos de la distribución sin su probabilidad (si un suceso tiene probabilidad 0 no debería estar acá).
- la probabilidad de un suceso en particular.

Queremos tener algunas facilidades para construir distribuciones de probabilidad de diferentes maneras:

- Evento Seguro: dado un suceso, es una distribución que sólo contiene ese suceso con probabilidad de 100%.
- Distribución equiprobable: a partir de una lista de sucesos genera una distribución donde cada suceso tiene la misma probabilidad de ocurrir.
- Distribución a partir de valores ponderados: dada una lista de sucesos y el “peso” de cada suceso, se obtiene una distribución. Ejemplo: si tenemos un test horrible y no determinístico que falla 2 de cada 3 veces, deberíamos poder construir la distribución a partir de esos valores en vez de tener que calcular la distribución a mano. Entonces, a partir de TestPasa con peso 1 y TestFalla con peso 2, obtendríamos una distribución con el suceso TestPasa con probabilidad 66.66% y TestFalla con probabilidad 33.34%.

Jugando un juego

Cada juego tiene asociada una distribución con sus resultados posibles. Por ejemplo, si fuera tirar la moneda a cara o cruz, sabemos que tenemos un 50% de que salga Cara y 50% de que salga Cruz. Cuando jugamos un juego ponemos una apuesta sobre el mismo y esto nos permite calcular otra distribución diferente, esta vez de las posibles ganancias. Entonces, si apostara \$30 a que sale cara, debería obtener una distribución con 50% de ganar \$60 y 50% de perder el monto apostado. En el caso de la ruleta, cuando se apuesta \$10 a un número, debería calcularse esta distribución sobre ganancias como un 1/37 de probabilidad de ganar \$360 y 36/37 de probabilidad de perder la apuesta, en vez de ver la misma como una distribución de 37 sucesos posibles donde 36 de ellos son perder la apuesta.

Juegos sucesivos

Queremos permitir simular que nuestro jugador juegue varios juegos uno atrás de otro partiendo de un monto inicial a apostar, los juegos que va a jugar y la apuesta que va a hacer en cada juego (dicha apuesta puede ser una apuesta combinada).

La forma en la que se resuelve esta sucesión de juegos es:

- El jugador va a jugar el primer juego apostando el monto y esto va a dar una distribución de posibles resultados.

- Para cada uno de esos resultados, si el jugador tiene suficiente plata para apostar en el siguiente juego, lo juega, si no, se queda en el molde y el resultado sería un evento seguro: no jugar, lo que va a hacer que siga con la misma plata.

Esto puede resultar en que un jugador “saltee” un juego porque no le da para apostar en ese, pero juegue a un juego que venía después porque para ese sí tenía suficiente plata.

El resultado final debería ser una distribución probabilística de todos los posibles desenlaces de jugar todos los juegos.

Ej.: se quiere jugar a tirar una moneda apostando \$10 a cara y luego a la ruleta apostando \$15 al 0, esto partiendo de un monto inicial de \$15.

En la primer jugada, se puede terminar con \$25 ($\$15 - \$10 + \20) con 50% o con \$5 ($\$15 - \$10 + 0$) con 50%.

En la segunda jugada bifurcamos por ambos casos:

- Tenemos \$25:
 - Tenemos 2,77% de probabilidad de conseguir \$550 ($\$25 - \$15 + \540).
 - Hay 97,23% de chances de quedar con \$10 ($\$25 - \$15 + \0).
- Tenemos \$5
 - No tenemos suficiente para apostar así que nos quedamos con \$5 con 100% de chances.

En definitiva, lo que nos interesa en cada rama es que ocurran ambos sucesos (el que generó la rama y el nuevo) para poder calcular las probabilidades de los distintos resultados finales, y cómo estos sucesos son independientes, es decir, no afectan la probabilidad del siguiente, se puede obtener la probabilidad de que ambos sucedan multiplicando sus probabilidades. En otras palabras, si obtenemos una distribución a partir de un suceso A, la probabilidad de los sucesos de la nueva distribución van a estar multiplicados por la probabilidad del suceso A.

En este caso, habría un 1,38% ($0,5 * 0,277$) de conseguir \$550 por jugar y ganar tanto con la moneda como con la ruleta, 48,61% ($0,5 * 0,97$) de terminar con \$10 por ganar a la moneda y perder a la ruleta y 50% ($0,5 * 1$) terminar con \$5 por perder la moneda y en consecuencia no jugar a la ruleta.

Eligiendo un plan de juego

Ahora que podemos conocer todos los resultados posibles dada una lista de juegos a jugar y un monto inicial, queremos elegir cuál de varias combinaciones de juegos es la que elegiría un jugador con un cierto monto inicial. Esto va a depender del tipo de jugador:

- Si es un jugador racional, va a ponerle un puntaje a la distribución final de las posibles ganancias haciendo la suma de cada posible ganancia * su probabilidad de ocurrencia, y va a elegir la lista de juegos que puntúe mejor según ese criterio.
- Un jugador arriesgado, en cambio, va a elegir la lista de juegos para la cual el suceso que le deje mayor ganancia sea mejor, sin importar la probabilidad del mismo.
- Un jugador cauto elegiría la sucesión de juegos en la cuál la probabilidad de no perder plata sea mayor. O sea, si en uno hay 20% de duplicar, 15% de quedar igual y 70% de quedar en 0, y en el otro hay 30% de triplicar, 50% de perder la mitad y 20% de quedar en 0, elige los primeros porque tiene 35% de chance de no perder plata vs 30% de chance con los segundos juegos.
- También queremos dar la posibilidad al usuario de crear un jugador al que le pasamos un criterio para comparar distribuciones y elige según ese criterio.

Requerimientos

1. Modelar los resultados de juegos posibles para la ruleta y para 'cara o cruz' y las jugadas mencionadas que se pueden hacer sobre ellos.
2. Permitir crear apuestas compuestas para los juegos cuyos resultados se modelaron en el punto anterior.
3. Crear las distribuciones de probabilidad de:
 - a. El juego de 'cara o cruz', donde hay 50% de chances de que salga Cara y 50% de que salga Cruz.
 - b. La ruleta, que tiene las mismas chances de que salga cualquiera de los 37 números.
 - c. 'Cara o cruz' pero con una moneda cargada, en este caso sale Cara 4 de cada 7 veces y Cruz las restantes.
4. Permitir que un jugador juegue sucesivamente varios juegos (hacer tests mínimamente con el caso del ejemplo).
5. Modelar los 3 diferentes tipos de jugadores e inventar uno más a partir de lo que se haya modelado para crear jugadores nuevos a partir de criterios.