

LICENCIATURA EN ESTADÍSTICA

“Análisis de distintas estrategias en el problema Multi-armed Bandit con estadística bayesiana”

Trabajo Práctico 1

Autores: Tomas Anderson - Alejo Vaschetti - Manuel Moresi

Docentes: Ignacio Evangelista - Tomás Capretto

17/04/2024

Tabla de contenidos

Introduccion	1
Desarrollo	1
Situación ideal	2
Estrategia 1: Completamente al azar	3
Estrategia 2: Greedy con tasa observada	5
Estrategia 3: Greedy con probabilidad a posteriori	7
Estrategia 4: ϵ -greedy (con tasa observada)	9
Estrategia 5: Softmax	12
Estrategia 6: Upper-bound	15
Estrategia 7: Thompson sampling	17
Conclusion	19
Anexo	20

Introduccion

El humano tiene que enfrentarse a una gran cantidad de decisiones en su vida diaria. Algunas pueden ser irrelevantes, como la elección de para qué lado dormirse o cuál es el primer pie al que se le pone la media al vestirse. En cambio, otros dilemas pueden ser de una importancia superior en la que se espera un mayor uso de racionalidad. Por ejemplo, al terminar la escuela, ¿empezar a trabajar o estudiar?, ¿qué carrera seguir?, ¿salir de fiesta con tanta lluvia el fin de semana?. Estos problemas de decisión que conllevan un poco más de esfuerzo pueden ser analizados de forma dicotómica, basar la elección en base a la seguridad de lo familiar y el conocimiento previo o decantarse por lo desconocido y arriesgado pero con posibles recompensas mayores. Usando los ejemplos anteriores, ¿seguir una carrera que se adecúe a las técnicas propias o elegir una carrera de conocimientos inexplorados?, ¿quedarse en casa viendo una serie o salir de fiesta pudiendo ser una noche inolvidable o una desastrosa?. Este dilema se lo conoce como “explorar vs. explotar” en el cual el primero hace referencia a la promesa de lo desconocido y lo segundo a la seguridad de lo familiar.

En este informe se analizará una situación hipotética que plantea la disyuntiva de “explorar vs. explotar”. El problema del *Multi-armed Bandit* se trata de imaginarse una situación en donde se tienen varias máquinas tragamonedas, es decir, máquinas en las que se hace girar unos rodillos con figuras en cada cara y en el caso de coincidir la figura se gana dinero, en caso contrario, no se gana nada. Para simplificar, se analizará el problema con el caso de 3 máquinas con distintas probabilidad de ganar, 0.3, 0.45 y 0.55 respectivamente, y donde jugar en las máquinas no cuesta dinero. Cada día se puede jugar en una única máquina una sola vez y el objetivo será ganar la mayor cantidad de unidades monetarias en un año bisiesto (366 días) idealmente descubriendo la máquina con mayor probabilidad de éxito. Este informe busca comparar distintas estrategias que tratarán de encontrar un equilibrio en el dilema “explorar vs. explotar”.

Desarrollo

Para la resolución de las incógnitas planteadas en la introducción se usan herramientas correspondientes al análisis bayesiano, donde la creencia a priori para las probabilidades de éxito de cada máquina a principio de año toma una distribución Beta(2, 2). Al considerar la variable ganar o no al jugar una máquina, esta toma una distribución Binomial(1, p) donde p es la probabilidad de éxito de cada máquina, por lo que la distribución a posteriori de p luego de un día tiene forma de Beta(2 + 1, 2) si la observación fue un éxito y Beta(2, 2 + 1) en el caso de perder. Por la secuencialidad de las técnicas bayesianas, esta distribución a posteriori pasa a ser la probabilidad a priori en el siguiente día, de esta manera se puede ir actualizando día a día la creencia sobre las probabilidades de éxito de cada máquina.

Se comparan siete estrategias utilizando simulaciones de, al principio un año para cada una y luego de 1000 años para ver las posibles variaciones que pueda llegar a tener cada método debido a la naturaleza aleatoria de las máquinas tragamonedas. Se programa una función capaz de elegir una máquina siguiendo la estrategia seleccionada, devolver un resultado y actualizar las creencias sobre las probabilidades de éxito. Esta es la que se usa para realizar las simulaciones.

Para tener un punto de referencia con las que se pueda comparar cada método a continuación se plantea un caso irreal e idóneo en el se conocen las probabilidades de éxito de cada máquina y se explota que más gana.

Situación ideal

La situación ideal sería conocer de antemano cual máquina es la que tiene la mayor probabilidad de ganar y jugar siempre con ella. Esta sería la mejor forma de ganar dinero, por lo que una buena estrategia tendría que asemejarse a dicha situación.

Para ver la cantidad de dinero que gana en promedio este método se simulan a 1000 años siguiéndolo.

Estos son los resultados:

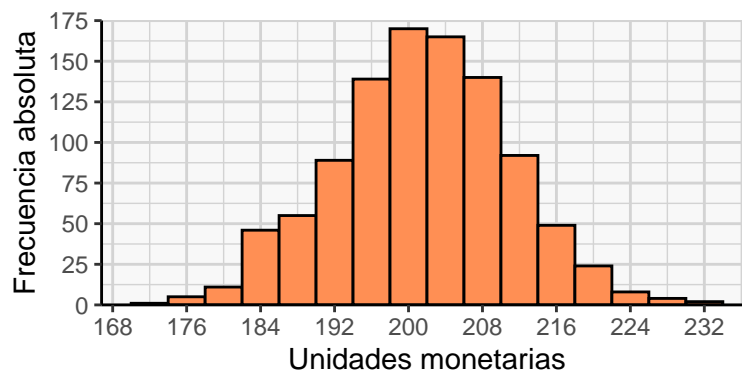


Figura 1: Ganancias en los distintos años

En este modus operandi una persona que juegue un año se espera que va a ganar alrededor de 201.986 unidades monetarias.

En el 95% de los años simulados las ganancias totales se encuentran entre 184 y 219 unidades monetarias. Dicho intervalo, y los presentados en las próximas estrategias, se encuentran entre el 2,5% y el 97,5% de las ganancias totales ordenadas de menor a mayor. La amplitud del intervalo es de 35.

Para ver que tan beneficiosa es cada estrategia, se comparan la cantidad de veces que se gana en promedio, esperando que se acerque lo máximo posible a esta forma de jugar. Además, se contrastan las amplitudes de los intervalos propuestos.

Estrategia 1: Completamente al azar

Esta estrategia consiste en elegir cada día al azar qué máquina jugar, teniendo las tres la misma probabilidad de ser selectas. Esta es la estrategia más básica, ya que no requiere ningún tipo de información previa ni evoluciona en el tiempo. A pesar de lo planteado en la introducción, este no es un método bayesiano, ya que no utiliza de ninguna manera la regla de Bayes para la elección de máquinas.

Se juega un año con esta estrategia y se obtienen los siguientes resultados:

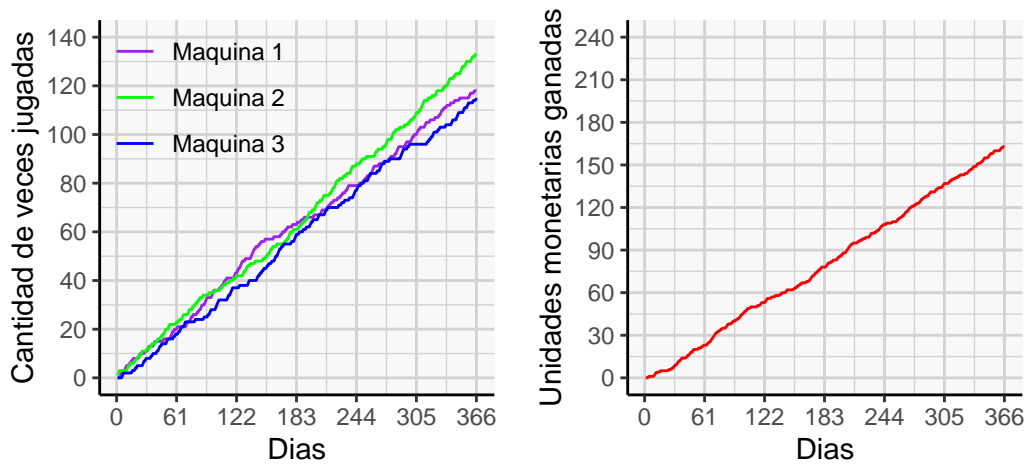


Figura 2: Evolución diaria del uso de las máquinas y la ganancia generada

La cantidad de veces que se juegan las 3 máquinas son similares, y la diferencia entre ellas se debe puramente al azar. Este hecho repercute en la evolución de la creencia sobre la probabilidad de éxito de cada máquina de la siguiente manera.

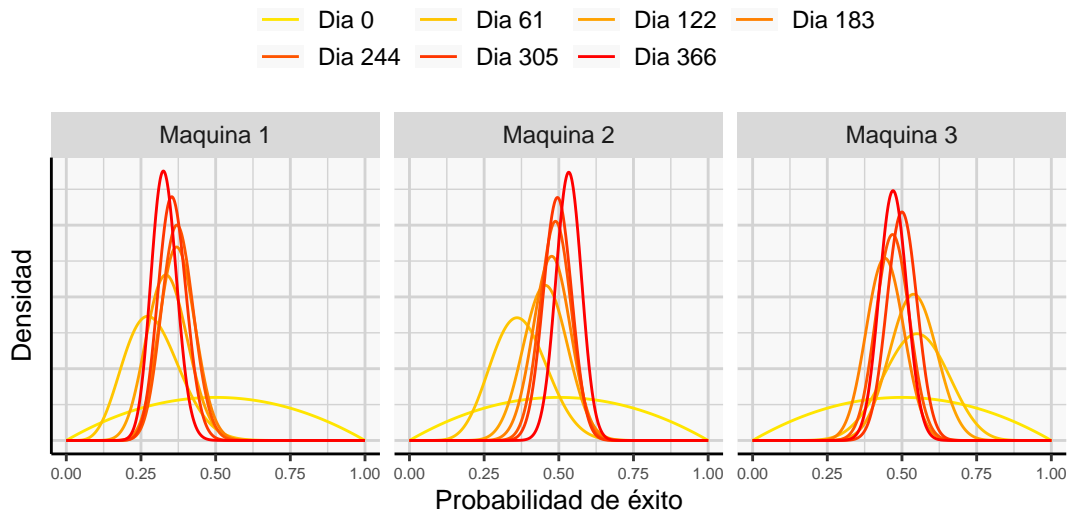


Figura 3: Distribución a posteriori de la probabilidad de éxito de las máquinas en ciertos días

Al jugarse una cantidad similar de veces en cada máquina la creencia se concentra en igual medida alrededor de los valores de las probabilidades de éxito de cada una.

Las simulaciones de los 1000 años arrojaron los siguientes resultados:

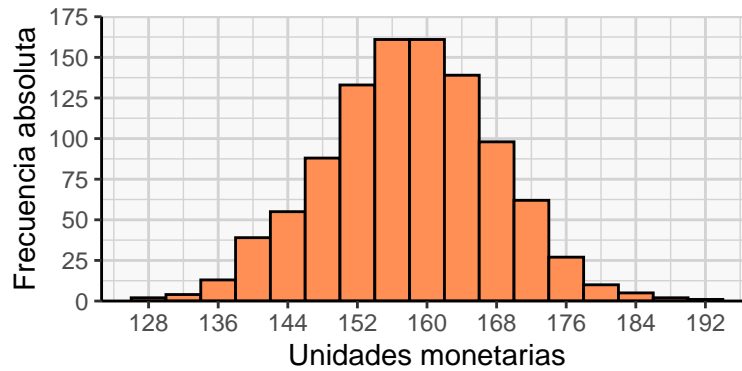


Figura 4: Ganancias en los distintos años

En promedio, siguiendo esta estrategia, se espera ganar 158.5 unidades monetarias en un año. Esto es un 21.53% menos que el caso ideal.

El 95% de las ganancias al final del año estuvieron entre 140 y 177. Su amplitud es de 37 unidades monetarias, siendo 1.06 veces el caso ideal, resultando muy parecidos entre sí.

El promedio de ganancias al final del año es sustancialmente más chico que el de la situación ideal y los intervalos no se superponen. Se puede pensar que esta estrategia no es recomendable al diferir tanto de un resultado ideal.

Estrategia 2: Greedy con tasa observada

La máquina que esta estrategia selecciona en un cierto día es aquella que tenga la mayor probabilidad de éxito observada hasta el momento. Como al empezar el año no se tiene ninguna información sobre la probabilidad de éxito de cada máquina, se decide jugar cada una por separado hasta que todas tengan su primer éxito. Si coinciden en algunas máquinas se elige una de ellas aleatoriamente. Al no hacer uso de las probabilidades a priori o posteriori, este método no se considera bayesiano.

Se juega un año con esta estrategia y se obtienen los siguientes resultados:

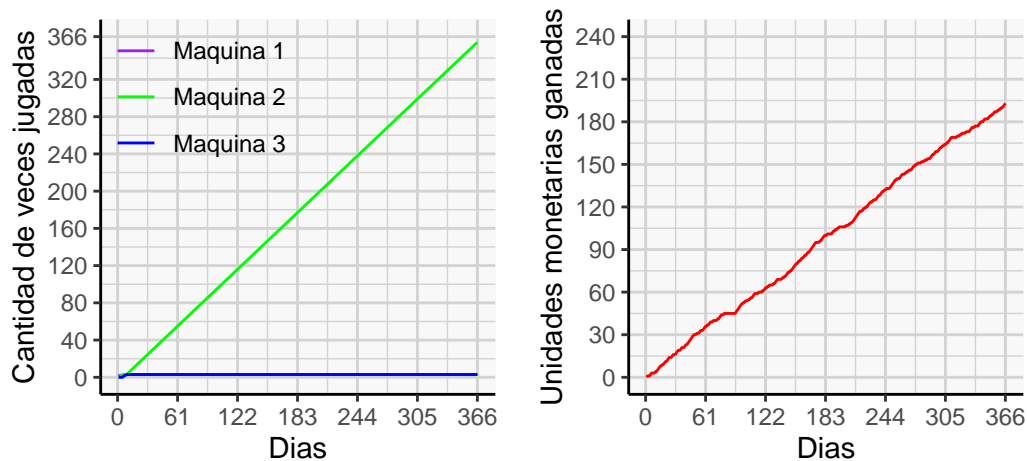


Figura 5: Evolución diaria del uso de las máquinas y la ganancia generada

En este año se usa casi exclusivamente la maquina 2 y apenas se utilizan las otras dos. Esto pasa porque las probabilidades de éxito iniciales observadas son muy variables, por lo que la primera maquina en tener varios fracasos va a tener una probabilidad muy chica y se

dejara de usar. Esto afecta a la creencia sobre la probabilidad de éxito de cada máquina de la siguiente manera.

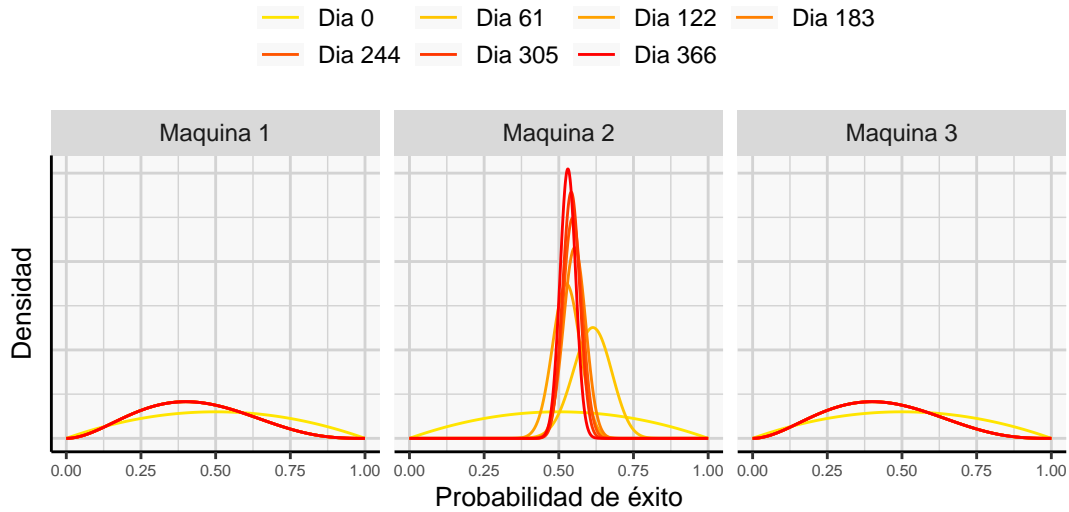


Figura 6: Distribución a posteriori de la probabilidad de éxito de las máquinas en ciertos días

La creencia sobre la probabilidad de éxito de la máquina 2 va a ser muy precisa, al tener muchas muestras de ella, y las demas seran muy parecidas a la suposición inicial.

Las simulaciones de los 1000 años arrojaron los siguientes resultados:

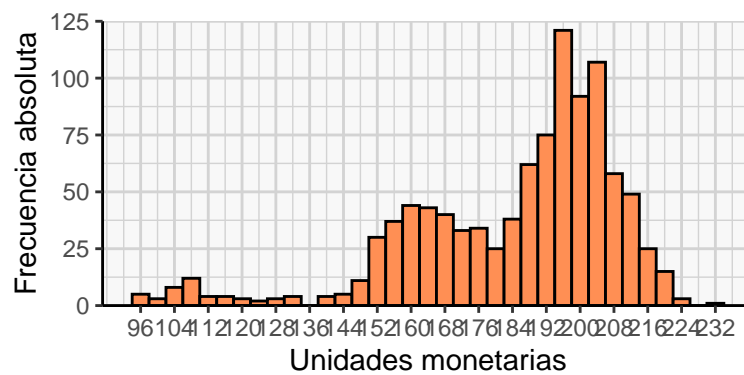


Figura 7: Ganancias en los distintos años

En promedio, siguiendo esta estrategia, se espera ganar 185.1 unidades monetarias en un año. Esto es un 8.36% menos que el caso ideal.

El 95% de las ganancias al final del año estuvieron entre 109 y 218. Su amplitud es de 109 unidades monetarias, siendo 3.11 veces más grande que el caso ideal.

Si bien es un método que en promedio gana una cantidad similar de dinero comparado a la situación ideal, existe el riesgo de ganar mucho menos que el promedio, ya que en general la estrategia suele usar una misma máquina en casi todo el año y hay una probabilidad no despreciable de elegir la peor de ellas. Esta es la razón de porque la amplitud del intervalo es tan grande.

Estrategia 3: Greedy con probabilidad a posteriori

En esta estrategia, la máquina a elegir en cada día es aquella que, hasta el momento, tenga la mayor esperanza entre las distribuciones a posteriori de las probabilidades de éxito. Si las máquinas tienen esperanzas iguales entonces se elige una de ellas aleatoriamente. Este sí es un método bayesiano, ya que hace uso de las distribuciones a priori y actualiza secuencialmente las creencias sobre las distribuciones de las probabilidades de éxito de cada máquina.

Se juega un año con esta estrategia y se obtienen los siguientes resultados:

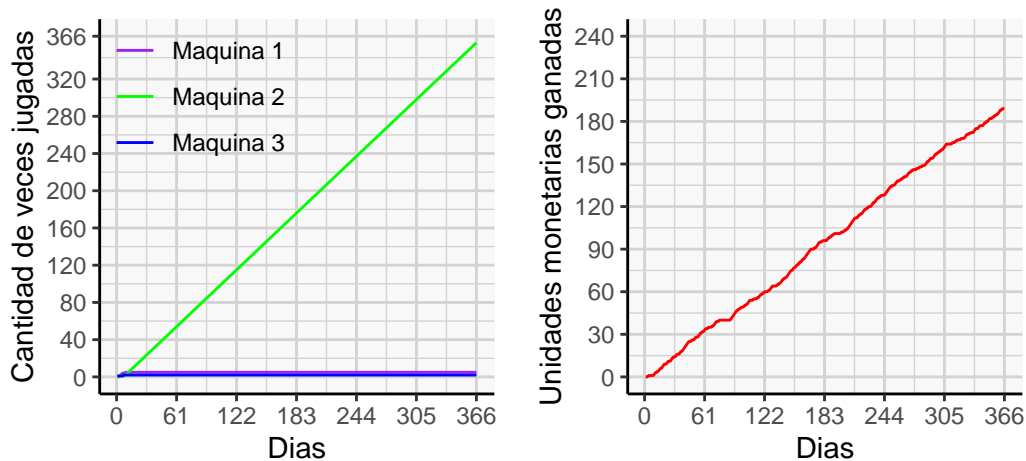


Figura 8: Evolución diaria del uso de las máquinas y la ganancia generada

La seleccion de las máquinas en el año simulado es notablemente similar a la estrategia anterior. La unica diferencia es que las máquinas 1 y 3 se usan un poco mas al principio del año, aunque sea difícil de ver.

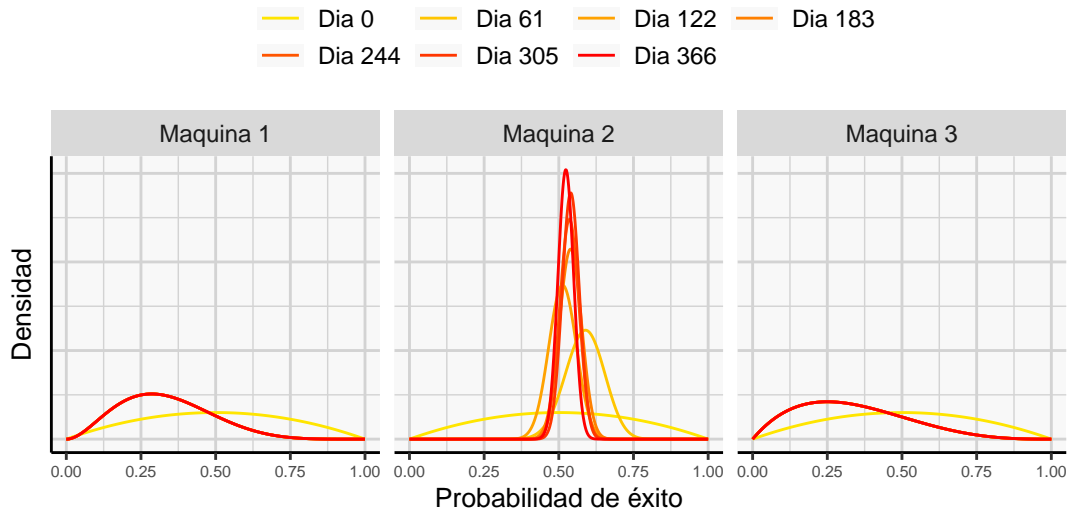


Figura 9: Distribución a posteriori de la probabilidad de éxito de las máquinas en ciertos días

Este método genera muchas observaciones de una máquina e ignora las demás, debido a su parecido con la estrategia anterior, por ende, la precision sobre la creencia de las probabilidades de éxito sera grande para una máquina y chica en las otras.

Las simulaciones de los 1000 años arrojaron los siguientes resultados:

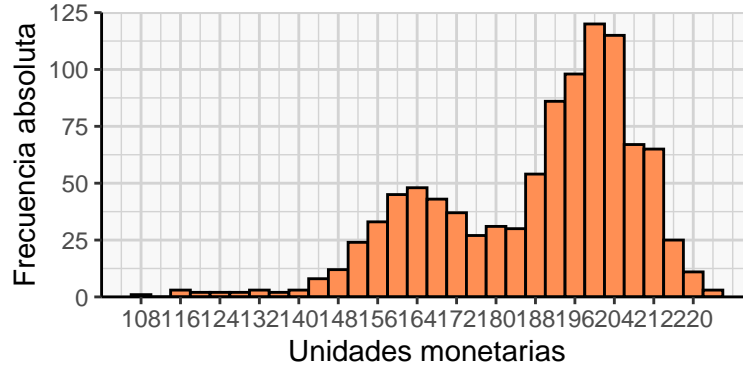


Figura 10: Ganancias en los distintos años

En promedio, siguiendo esta estrategia, se espera ganar 188.53 unidades monetarias en un año. Esto es un 6.66% menos que el caso ideal.

El 95% de las ganancias al final del año estuvieron entre 145 y 216. Su amplitud es de 71 unidades monetarias, siendo 2.03 veces más grande que el caso ideal.

Esta estrategia es un poco mejor que la anterior ya que la cola izquierda del histograma es menos pesada. Esto se debe a que al principio de cada año se tiene una mejor aproximación de las probabilidades de éxito al haber utilizado la información de sus distribuciones a priori. Aun así, comparada con la situación ideal, sus ganancias son muy variables, por lo que no es muy recomendable.

Estrategia 4: ϵ -greedy (con tasa observada)

Esta estrategia es una combinación de otras dos anteriores, la greedy con tasa observada y la completamente al azar. Se elige utilizar el primer método mencionado con una probabilidad de $1 - \epsilon$ y el segundo con una probabilidad de ϵ . Mientras más cerca esté ϵ de 1, más se parece al completamente al azar y mientras más cerca esté del 0, al greedy con tasa observada. Es decir, a valores chicos de ϵ se prioriza el concepto de explotación, y a valores más grandes le da una mayor importancia a la exploración. Debido a que uno decide qué número asignarle a ϵ , se realiza un análisis previo para obtener su mejor valor. No es una estrategia que use técnicas bayesianas, ya que es una combinación de dos métodos que no las utilizan.

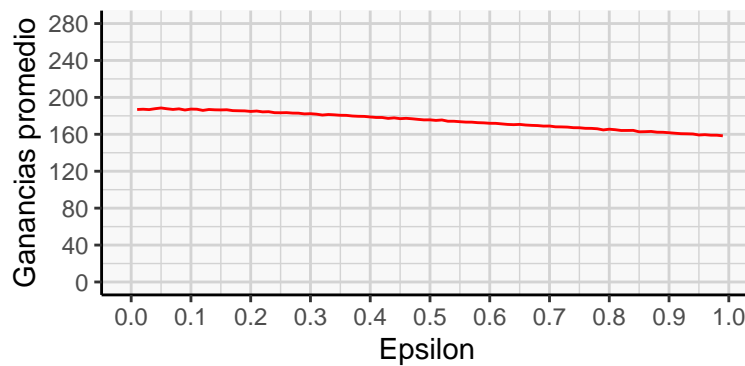


Figura 11: Ganancia promedio para distintos valores de

Se quiere ganar en promedio lo mayor posible. El valor de ϵ que maximiza las ganancias es 0.05.

Se juega un año con esta estrategia y se obtienen los siguientes resultados:

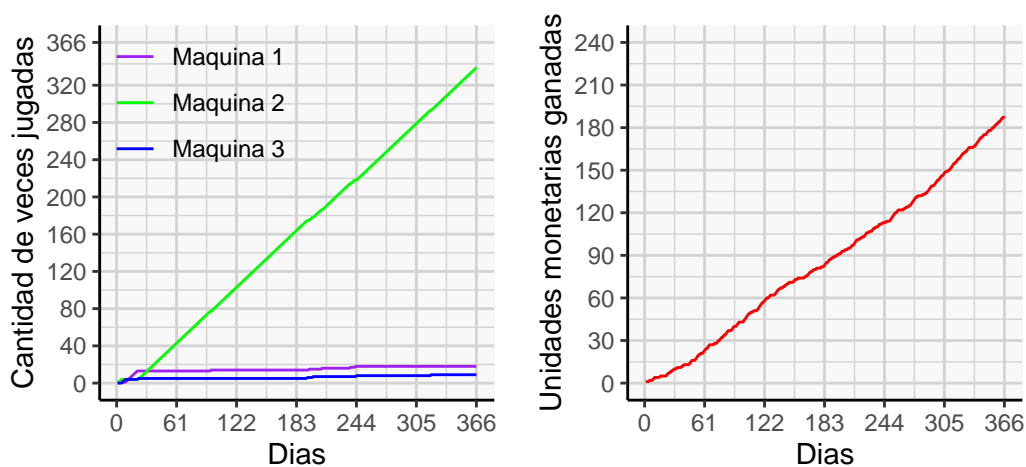


Figura 12: Evolución diaria del uso de las máquinas y la ganancia generada

En el primer mes de juego, se decide cual es la máquina con mayor probabilidad de éxito y luego se utiliza mayoritariamente una sola hasta el final del año. La máquina 1 y 3 se utilizan en el resto del año solo cuando se emplea la estrategia completamente al azar.

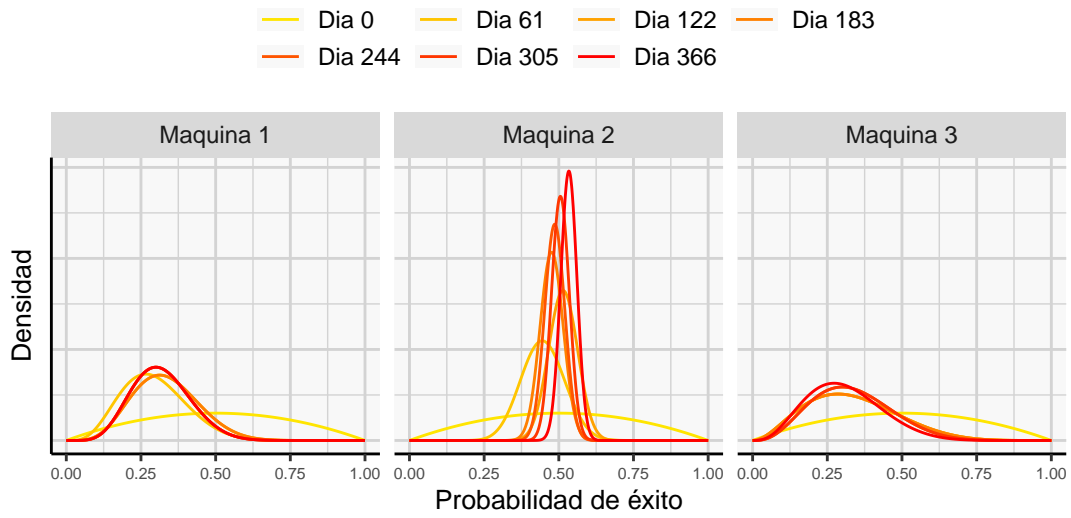


Figura 13: Distribución a posteriori de la probabilidad de éxito de las máquinas en ciertos días

La precisión de la creencia de la máquina 2 es alta, ya que es la que se utiliza con mayor frecuencia. Las demás, luego de los primeros meses, no cambia mucho.

Las simulaciones de los 1000 años arrojaron los siguientes resultados:

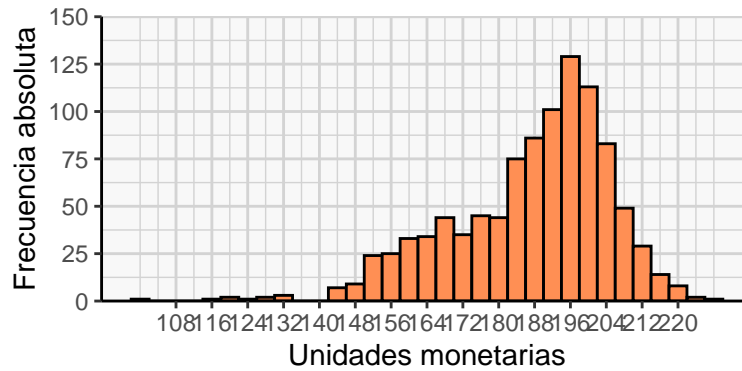


Figura 14: Ganancias en los distintos años

En promedio, siguiendo esta estrategia, se espera ganar 187.67 unidades monetarias en un año. Esto es un 7.09% menos que el caso ideal.

El 95% de las ganancias al final del año estuvieron entre 149 y 214. Su amplitud es de 65 unidades monetarias, siendo 1.86 veces el caso ideal.

El problema de la greedy con tasa observada es, a veces, no utilizar mayoritariamente la máquina con mayor probabilidad de éxito. El usar la estrategia completamente al azar esporádicamente, permite explorar las probabilidades de éxito de las máquinas para que al utilizar la greedy con tasa observada, esta seleccione la mejor de ellas con mayor regularidad. Comparada con la greedy con probabilidad a posteriori, su ganancia promedio y variabilidad son levemente menores. Aun así, su intervalo es demasiado amplio en comparación a la situación ideal.

Estrategia 5: Softmax

En esta estrategia, para elegir la máquina a jugar primero se obtiene la probabilidad de éxito observada π_i de cada máquina y se les aplica la siguiente fórmula:

$$\Pr(i) = \frac{e^{\pi_i/\tau}}{\sum_{j=1}^3 e^{\pi_j/\tau}}$$

Siendo τ un parámetro de “temperatura” que controla el grado de exploración. Luego se elige la máquina i con probabilidad $\Pr(i)$. Esto significa que no es un método bayesiano porque no utiliza distribuciones de probabilidad a priori ni a posteriori.

Se observa en la siguiente tabla las razones de las probabilidades de éxito reales de las máquinas aplicando la fórmula para ver qué les ocurre cuando τ varía.

	Maquina 2 / Maquina 1	Maquina 2 / Maquina 3
Originales	1.83	1.22
Temperatura = 1/25	557.77	13.12
Temperatura = 1	1.28	1.11
Temperatura = 25	1.01	1.00

Comparación de probabilidades usando Softmax

Cuando la temperatura crece, la diferencias entre las probabilidades de éxito de cada maquina disminuyen, mientras que las diferencias aumentan cuando la temperatura es menor. Cuando la temperatura es alta, se exploran todas las máquinas, mientras que cuando es baja, se explota la que hasta el momento tenga la mayor probabilidad de éxito.

Para ver la temperatura a seleccionar, se grafican las ganancias promedio en 1000 años para distintos valores.

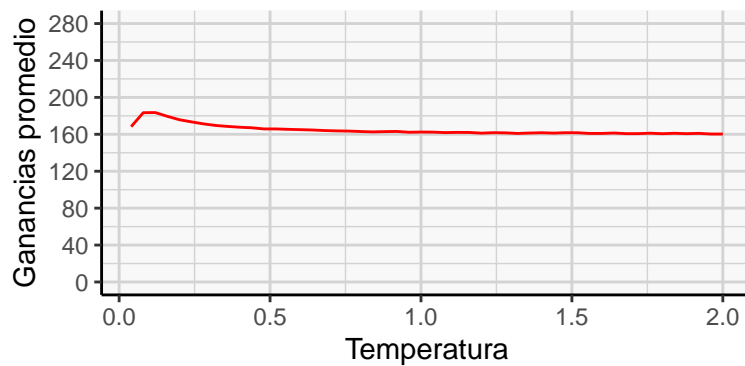


Figura 15: Ganancia promedio para distintos valores de

Se quiere ganar en promedio lo mayor posible. El valor de τ que maximiza las ganancias es 0.12.

Se juega un año con esta estrategia y se obtienen los siguientes resultados:

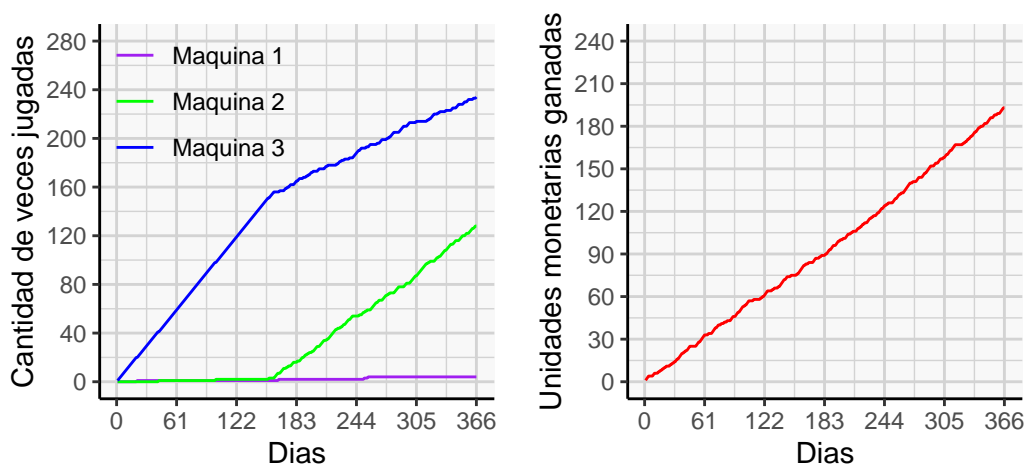


Figura 16: Evolución diaria del uso de las máquinas y la ganancia generada

En los primeros cinco meses del año, se usa mayoritariamente la máquina 3 pero llega un punto donde también comienza a utilizar la máquina 2. Esto se debe a que el método

permite la exploración de las máquinas que no está explotando. Al usar la máquina 2, aumenta su probabilidad de éxito observada, que a su vez incrementa la probabilidad de ser elegida. En el siguiente gráfico se puede observar dicho efecto.

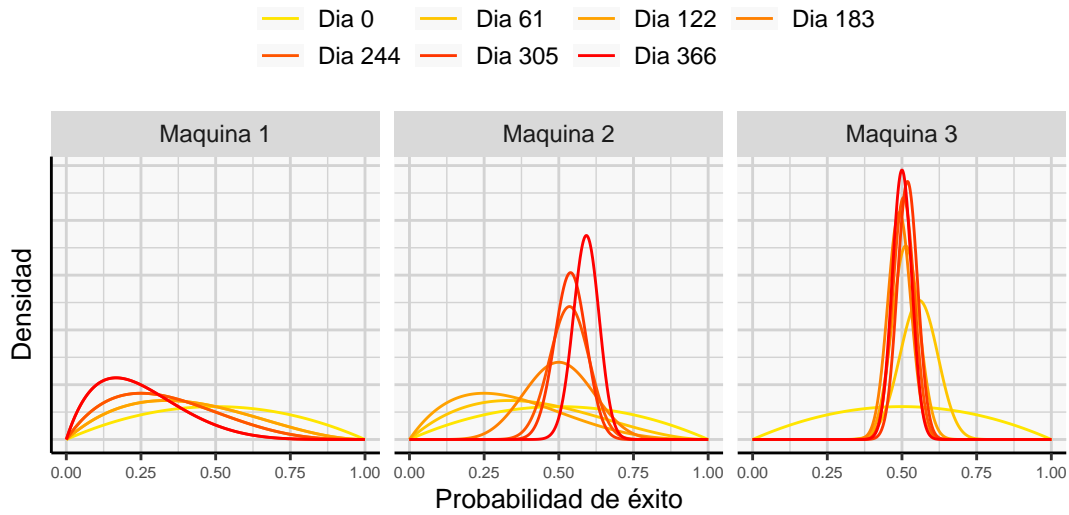


Figura 17: Distribución a posteriori de la probabilidad de éxito de las máquinas en ciertos días

Para observar como varían las ganancias usando este método, se simularon 1000 años, los cuales arrojaron los siguientes resultados:

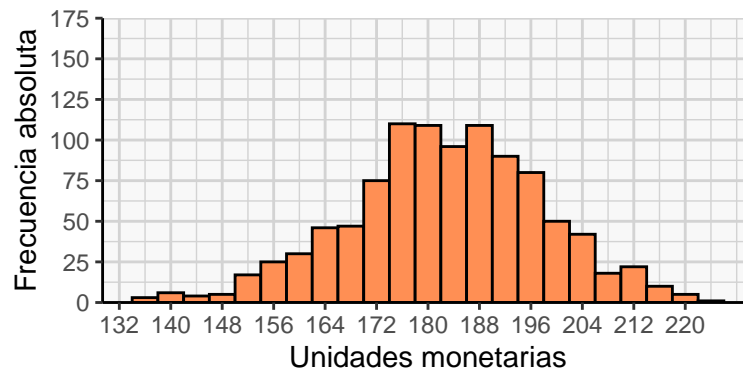


Figura 18: Ganancias en los distintos años

En promedio, siguiendo esta estrategia, se espera ganar 183.22 unidades monetarias en un año. Esto es un 9.29% menos que el caso ideal.

El 95% de las ganancias al final del año estuvieron entre 152 y 216. Su amplitud es de 60 unidades monetarias, siendo 1.71 veces el caso ideal.

Su ganancia promedio es menor que todas las estrategias greedy, pero la amplitud de su intervalo es mas baja. Es decir, si una persona quisiera tener más seguridad sobre la cantidad de unidades monetarias a ganar jugaría esta a las mencionadas.

Estrategia 6: Upper-bound

Se selecciona la máquina que tenga el mayor extremo derecho de un intervalo de credibilidad de la distribución a posteriori de la probabilidad de éxito. En este caso, el extremo derecho del intervalo usado es aquel que acumula el 97.5% de la densidad. Si los extremos derechos de los intervalos son iguales para algunas máquinas, se elige una de ellas aleatoriamente. Este es un método bayesiano por usar intervalos de credibilidad de las distribuciones a posteriori de las probabilidades de éxito de cada máquina.

Se juega un año con esta estrategia y se obtienen los siguientes resultados:

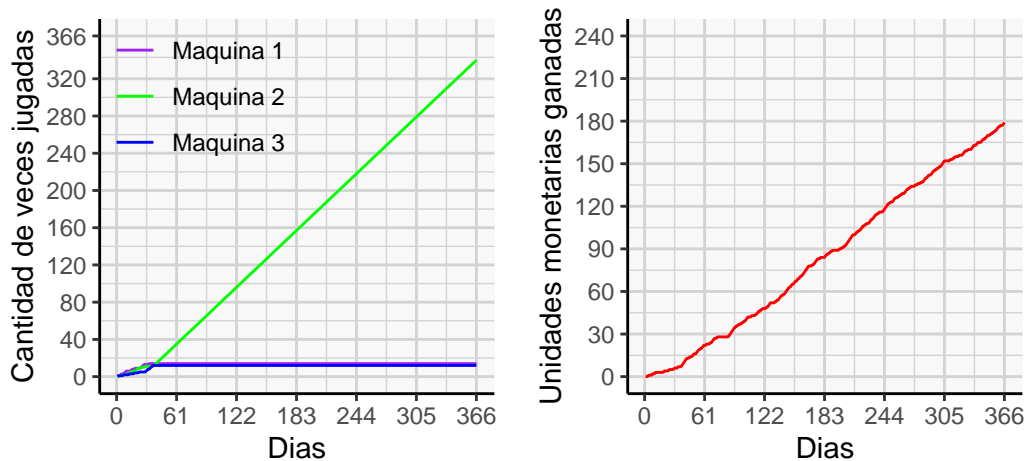


Figura 19: Evolución diaria del uso de las máquinas y la ganancia generada

Al comienzo del año se juegan todas las máquinas en una proporción parecida, pero al cabo de un mes y un par de días el método elige utilizar exclusivamente la máquina 2.

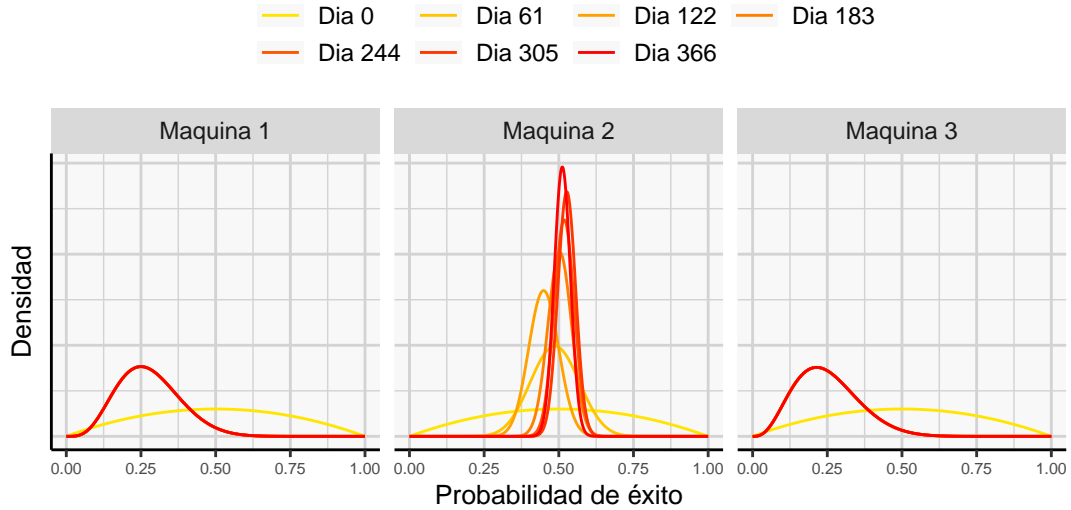


Figura 20: Distribución a posteriori de la probabilidad de éxito de las máquinas en ciertos días

Como varias de las estrategias anteriores, luego de un cierto número de días procede a jugar con una sola máquina. Pero para poder observar su diferencia con las demás técnicas se realizaron simulaciones de 1000 años, las cuales arrojaron los siguientes resultados:

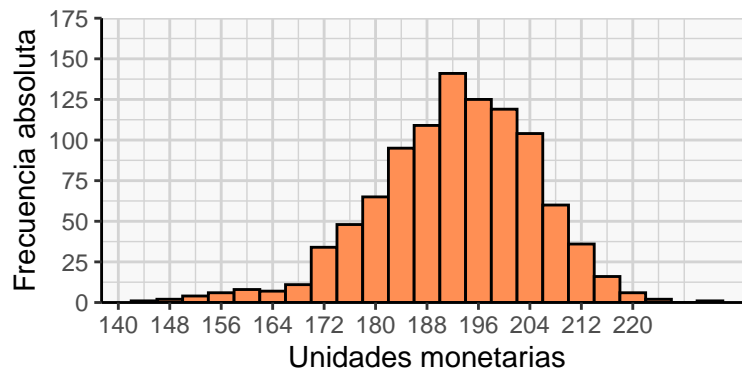


Figura 21: Ganancias en los distintos años

En promedio, siguiendo esta estrategia, se espera ganar 192.9 unidades monetarias en un año. Esto es un 4.5% menos que el caso ideal.

El 95% de las ganancias al final del año estuvieron entre 164 y 214. Su amplitud es de 50 unidades monetarias, siendo 1.43 veces el caso ideal.

Se puede observar que el histograma de esta estrategia es mucho mas simétrico que el de las técnicas anteriores. Esto se debe a que cada año la máquina con la que más veces se suele quedar el método es la 2, es decir, la que mayor probabilidad de éxito tiene.

Estrategia 7: Thompson sampling

Para elegir que máquina se usa en el día, se saca una muestra de la distribución a posteriori de las probabilidades de éxito de cada una de ellas y se selecciona aquella con el mayor valor muestral. Este es un método bayesiano por usar las distribuciones a posteriori de las probabilidades de éxito de cada máquina.

Se juega un año con esta estrategia y se obtienen los siguientes resultados:

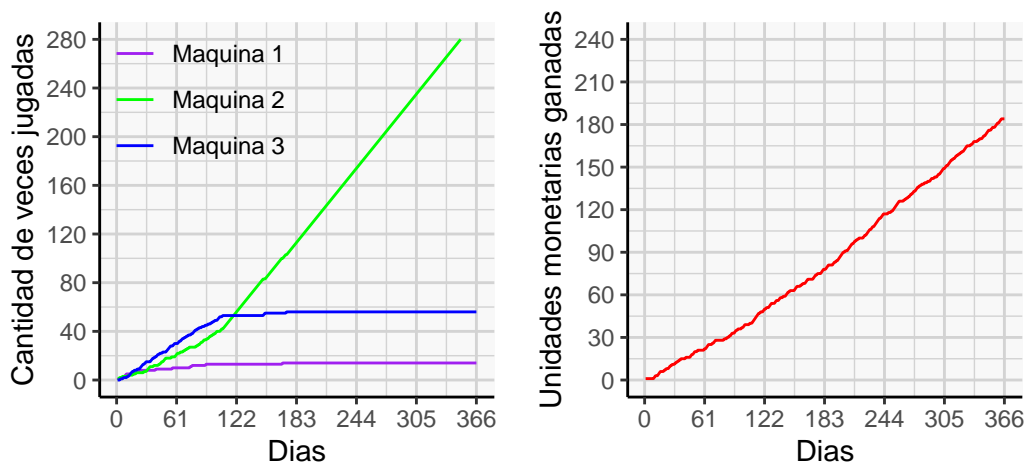


Figura 22: Evolución diaria del uso de las máquinas y la ganancia generada

En los primero 3 o 4 meses, la estrategia usa todas las máquinas. Luego de este período, utiliza mayormente la segunda dejando de utilizar las demás aproximadamente a mitad de año.

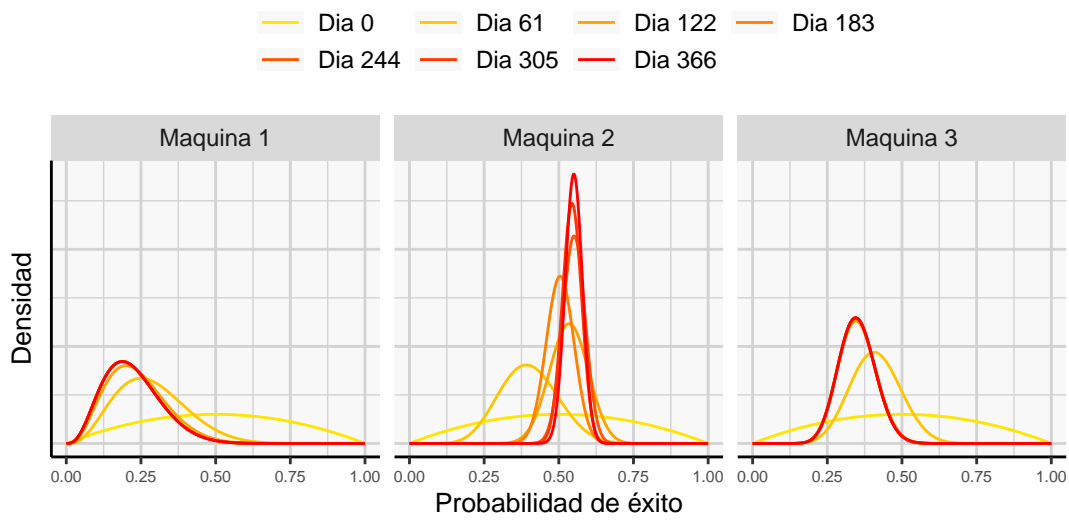


Figura 23: Distribución a posteriori de la probabilidad de éxito de las máquinas en ciertos días

Generalmente esta estrategia, como las demás, luego de un cierto número de días procede a jugar con una sola máquina.

Para observar como varían las ganancias usando este método se simularon 1000 años, los cuales arrojaron los siguientes resultados:

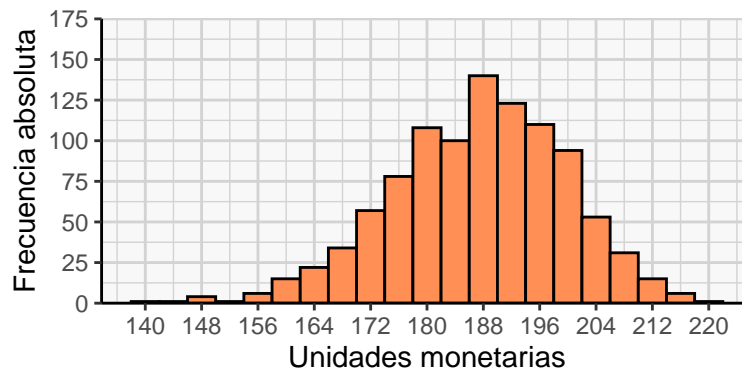


Figura 24: Ganancias en los distintos años

En promedio, siguiendo esta estrategia, se espera ganar 187.89 unidades monetarias en un

año. Esto es un 6.98% menos que el caso ideal.

El 95% de las ganancias al final del año estuvieron entre 161 y 210. Su amplitud es de 49 unidades monetarias, siendo 1.4 veces el caso ideal.

Esta estrategia, como la anterior, tiene un histograma bastante simétrico. Es mejor que los otros métodos y aunque la amplitud de su intervalo sea ligeramente más pequeño, es peor que el upper bound, ya que tiene un promedio de ganancias más chico.

Conclusion

En todas las estrategias usadas se vio entrelazado el concepto de explorar o explotar, dándole distintas importancias a cada una. Los mejores métodos son lo que pueden encontrar un equilibrio entre ambos enfoques, es decir, aquel que explore lo justo y necesario para asegurarse de elegir la mejor opción al momento de explotar. Viendo desde este enfoque los resultados, la mejor estrategia entre las utilizadas en el informe es la upper bound, siendo esta la que mejor supo balancear ambos conceptos dando como consecuencia el mayor promedio de unidades monetarias ganadas y con casi la menor variabilidad entre todos los métodos. Puede existir una estrategia más eficiente que capture mejor el concepto de “explorar vs explotar”, por lo que se recomienda profundizar en el tema.

Anexo

```
#Paquetes
library(tidyverse)
library(ggplot2)
library(gridExtra)
library(ggpubr)
library(knitr)
library(kableExtra)

#Probabilidades de exito de las maquinas
prob=c(.3, .55, .45)

#Tema de los graficos
tema = theme(panel.grid = element_line(color = "lightgrey"),
             panel.background = element_rect(fill = "#f8f8f8"),
             axis.line = element_line(color = "black"))

#Colores de los graficos de las distribuciones a posteriori
colores=c("#ffe500", "#ffa200", "#ff8000", "#ff5500", "#ff3600", "#ff0000", "#ffc500")
```

Funciones

```
# ----- Cual Maximo -----
cual_maximo = function(max_vec){
  maximo = which(max(max_vec) == max_vec)
  if (length(maximo) == 1) {
    return(maximo)
  } else {return(sample(maximo, 1))}
}

#Elige la posición del maximo de un vector. Si hay empate, elige uno aleatoriamente

#Función para generar n muestras de una persona que juegue k días especificando la estrategia

actualizar = function(dias = 366, muestras = 1, metodo = "random", temp = 1, e = .5){
  lista = list(random = random, greedy_tasa = greedy_tasa, greedy_posterior = greedy_posterior,
              e_greedy = e_greedy, softmax = soft, upper = upper, thompson = thompson)
  metodo = lista[[metodo]]
  matriz = matrix(0, dias*muestras, 10)
```

```

for (j in 1:muestras) {
  y = c(0, 0, 0, 0, 0, 0, j)
  for (i in 1:dias) {
    x=metodo(y, temp, e)
    y[x[2]] = y[x[2]] + x[1]
    y[x[2]+3] = y[x[2] + 3] + 1
    matriz[i+(dias*(j-1)),] = c(x, y, i)
  }
}
datos = as.data.frame(matriz)
colnames(datos) = c("Exito", "Maquina", "Maquina1", "Maquina2", "Maquina3", "Maquina1Total",
                    "Maquina2Total", "Maquina3Total", "Muestra", "Dia")
return(datos)
}

```

Situación ideal

```

#Generación de 1000 años
set.seed(69)
datos = matrix(rbinom(366000, 1, prob[2]), nrow = 1000, ncol = 366, byrow = T)
ideal = as.data.frame(rowSums(datos))
colnames(ideal) = "x"

#Histograma de las ganancias para los años
ggplot(ideal, aes(x))+geom_histogram(fill = "#FF8F54", color = "black",binwidth = 4) +
  scale_y_continuous(expand = expand_scale(add = c(0, 0)), limits = c(0,175), breaks = seq(0
  scale_x_continuous(breaks = seq(168,232,8), name = "Unidades monetarias")+ tema

#Calculo de la ganancia promedio y los intervalos
Media_ideal = mean(rowSums(datos))
Q_ideal = sort(rowSums(datos))
Q_ideal_int = c(mean(Q_ideal[25], Q_ideal[26]), mean(Q_ideal[975], Q_ideal[976]))

```

Estrategia 1: Completamente al azar

```

#Generación de un dia para la estrategia

random = function(y, temp, e){
  cual = floor(runif(1,1,4))
  vector = c(rbinom(1, 1, prob[cual]), cual)
  return(vector)}

```

```

#Generación de 1 año
set.seed(69)
datos_random_dia = actualizar(366, 1)

#Gráfico de la elección diaria de máquina

plot1_random_dia = ggplot(datos_random_dia)+
  geom_line(aes(x = Dia, y = Maquina1Total, colour = "Maquina 1"))+
  geom_line(aes(x = Dia, y = Maquina2Total, colour = "Maquina 2"))+
  geom_line(aes(x = Dia, y = Maquina3Total, colour = "Maquina 3"))+
  tema+
  scale_y_continuous(name = "Cantidad de veces jugadas", breaks = c(seq(0, 140, 20), 366), limits = c(0, 366))+
  scale_x_continuous(name="Dias",breaks = seq(0, 366, 61)) +
  scale_color_discrete(breaks=c("Maquina 1", "Maquina 2", "Maquina 3"),type = c("purple", "green", "blue"))+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))+
  theme(legend.position = c(.25,.85), legend.background = element_blank(), legend.key = element_blank())

#Gráfico de la ganancia diaria

plot2_random_dia = ggplot(datos_random_dia)+
  geom_line(aes(x = Dia, y = Maquina1 + Maquina2 + Maquina3), color = "red")+
  scale_y_continuous(name="Unidades monetarias ganadas", breaks = c(seq(0, 240, 30)), limits = c(0, 240))+
  scale_x_continuous(name="Dias", breaks = seq(0, 366, 61))+tema+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))

#Unión de los dos gráficos
grid.arrange(plot1_random_dia, plot2_random_dia, ncol = 2)

#Generación de las curvas de las binomiales para los días seleccionados
set.seed(69)
datos_random_post = matrix(seq(0, 1, .001), nrow = 3003)
datos_random_post = cbind(datos_random_post, dbeta(seq(0, 1, .001), 2, 2))
Maquina1_random = matrix(nrow = 1001, ncol = 6)
Maquina2_random = matrix(nrow = 1001, ncol = 6)
Maquina3_random = matrix(nrow = 1001, ncol = 6)

for (i in 1:6) {
  Maquina1_random[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_random_dia$Maquina1[61*i], 2 + datos_random_dia$Maquina1[61*i]), nrow = 1001)
  Maquina2_random[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_random_dia$Maquina2[61*i], 2 + datos_random_dia$Maquina2[61*i]), nrow = 1001)
  Maquina3_random[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_random_dia$Maquina3[61*i], 2 + datos_random_dia$Maquina3[61*i]), nrow = 1001)
}

```



```

Maquina3_random[,i] = matrix(dbeta(seq(0,1,.001), 2+datos_random_dia$Maquina3[61*i], 2-datos_random_dia$Maquina3[61*i]), 2-datos_random_dia$Maquina3[61*i])
}
Maquinas_random = as.data.frame(rbind(Maquina1_random, Maquina2_random, Maquina3_random))
Maquinas_random = cbind(Maquinas_random, matrix(rep(c("Maquina 1", "Maquina 2", "Maquina 3"), nrow(Maquinas_random)), nrow(Maquinas_random)))
datos_random_post = cbind(datos_random_post, Maquinas_random)
colnames(datos_random_post) = c("seq", "Dia0", "Dia61", "Dia122", "Dia183", "Dia244", "Dia305", "Dia366")

#Gráfico de las distribuciones a posteriori para cada día
plot3_random_dia = ggplot(datos_random_post, aes(col = factor(Maquina))) +
  geom_line(aes(x = seq, y = Dia0 ,color = "Dia 0"))+
  geom_line(aes(x = seq, y = Dia61,color = "Dia 61"))+
  geom_line(aes(x = seq, y = Dia122,color = "Dia 122"))+
  geom_line(aes(x = seq, y = Dia183,color = "Dia 183"))+
  geom_line(aes(x = seq, y = Dia244,color = "Dia 244"))+
  geom_line(aes(x = seq, y = Dia305,color = "Dia 305"))+
  geom_line(aes(x = seq, y = Dia366, color = "Dia 366"))+
  facet_grid(~ Maquina)+
  tema+
  scale_y_continuous(name = "Densidad")+
  theme(axis.ticks.y = element_blank(), axis.text.y = element_blank())+
  scale_x_continuous(name = "Probabilidad de éxito", breaks = seq(0,1,.25)) +
  scale_color_discrete(breaks=c("Dia 0", "Dia 244", "Dia 61", "Dia 305", "Dia 122", "Dia 366"))+
  theme(legend.background = element_blank(), legend.key.height = unit(.1, "cm"), plot.margin = margin(10, 10, 10, 10))
  theme(axis.title.x = element_text(size = 10), axis.title.y = element_text(size = 10), axis.text.x = element_text(size = 10), axis.text.y = element_text(size = 10))

plot3_random_dia

#Generación de 1000 años
set.seed(69)
datos_random_año = actualizar(366, 1000)

#Obtención de la ganancia total al final de cada año
datos_random_filter = filter(datos_random_año, Dia == 366)
Exitos_random_totales = as.data.frame(rowSums(datos_random_filter[, 3:5]))
colnames(Exitos_random_totales) = "x"

#Histograma de las ganancias promedio para los años

```

```

ggplot(Exitos_random_totales, aes(x))+geom_histogram(fill = "#FF8F54", color = "black", binwidth = 1,
  scale_y_continuous(expand = expand_scale(add = c(0, 0)), breaks = seq(0,175,25), limits = c(0, 175)),
  scale_x_continuous(breaks = seq(128,192,8), name = "Unidades monetarias")+ tema+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))

#Cálculo de la ganancia promedio y los intervalos
Media_random = round(mean(Exitos_random_totales$x),2)

Q_random = sort(Exitos_random_totales$x)

Q_random_int = c(mean(Q_random[25], Q_random[26]), mean(Q_random[975], Q_random[976]))

```

Estrategia 2: Greedy con tasa observada

```

#Generación de un día para la estrategia
greedy_tasa = function(y, temp, e){
  if (y[1] == 0) {
    vector = c(rbinom(1, 1, prob[1]), 1)
    return(vector)
  }
  if (y[2] == 0) {
    vector = c(rbinom(1, 1, prob[2]), 2)
    return(vector)
  }
  if (y[3] == 0) {
    vector = c(rbinom(1, 1, prob[3]), 3)
    return(vector)
  }
  cual = cual_maximo(c(y[1:3]) / c(y[4:6]))
  vector = c(rbinom(1, 1, prob[cual]), cual)
  return(vector)
}

#Generación de 1 año

set.seed(69)

datos_greedy_tasa_dia = actualizar(366, 1,"greedy_tasa")

#Gráfico de la elección diaria de máquina

```

```

plot1_greedy_tasa_dia = ggplot(datos_greedy_tasa_dia)+
  geom_line(aes(x = Dia, y = Maquina1Total, colour = "Maquina 1"))+
  geom_line(aes(x = Dia, y = Maquina2Total, colour = "Maquina 2"))+
  geom_line(aes(x = Dia, y = Maquina3Total, colour = "Maquina 3"))+
  tema+
  scale_y_continuous(name = "Cantidad de veces jugadas", breaks = c(seq(0, 320, 40), 366), 1)
  scale_x_continuous(name="Dias",breaks = seq(0, 366, 61)) +
  scale_color_discrete(breaks=c("Maquina 1", "Maquina 2", "Maquina 3"),type = c("purple", "green", "red"))+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))+
  theme(legend.position = c(.25,.85), legend.background = element_blank(), legend.key = element_blank())

#Gráfico de la ganancia diaria

plot2_greedy_tasa_dia = ggplot(datos_greedy_tasa_dia)+
  geom_line(aes(x = Dia, y = Maquina1 + Maquina2 + Maquina3), color = "red")+
  scale_y_continuous(name="Unidades monetarias ganadas", breaks = c(seq(0, 240, 30)), limits = c(0, 366))
  scale_x_continuous(name="Dias", breaks = seq(0, 366, 61))+tema+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))

#Unión de los dos gráficos

grid.arrange(plot1_greedy_tasa_dia, plot2_greedy_tasa_dia, ncol = 2)

#Generación de las curvas de las binomiales para los días seleccionados

set.seed(69)
datos_greedy_tasa_post = matrix(seq(0,1,.001), nrow = 3003)
datos_greedy_tasa_post = cbind(datos_greedy_tasa_post, dbeta(seq(0, 1, .001), 2, 2))
Maquina1_greedy_tasa = matrix(nrow = 1001, ncol = 6)
Maquina2_greedy_tasa = matrix(nrow = 1001, ncol = 6)
Maquina3_greedy_tasa = matrix(nrow = 1001, ncol = 6)

for (i in 1:6) {
  Maquina1_greedy_tasa[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_greedy_tasa_dia$Maquina1Total, 2 + datos_greedy_tasa_dia$Maquina2Total), nrow = 1001, ncol = 1)
  Maquina2_greedy_tasa[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_greedy_tasa_dia$Maquina2Total, 2 + datos_greedy_tasa_dia$Maquina3Total), nrow = 1001, ncol = 1)
  Maquina3_greedy_tasa[,i] = matrix(dbeta(seq(0,1,.001), 2+datos_greedy_tasa_dia$Maquina3Total, 2+datos_greedy_tasa_dia$Maquina1Total), nrow = 1001, ncol = 1)
}

Maquinas_greedy_tasa = as.data.frame(rbind(Maquina1_greedy_tasa, Maquina2_greedy_tasa, Maquina3_greedy_tasa))
Maquinas_greedy_tasa = cbind(Maquinas_greedy_tasa, matrix(rep(c("Maquina 1", "Maquina 2", "Maquina 3"), 1001), ncol = 3))

```

```

datos_greedy_tasa_post = cbind(datos_greedy_tasa_post, Maquinas_greedy_tasa)
colnames(datos_greedy_tasa_post) = c("seq", "Dia0", "Dia61", "Dia122", "Dia183", "Dia244", "Dia305", "Dia366")

#Gráfico de las distribuciones a posteriori para cada día

plot3_greedy_tasa_dia = ggplot(datos_greedy_tasa_post, aes(col = factor(Maquina))) +
  geom_line(aes(x = seq, y = Dia0 ,color = "Dia 0"))+
  geom_line(aes(x = seq, y = Dia61,color = "Dia 61"))+
  geom_line(aes(x = seq, y = Dia122,color = "Dia 122"))+
  geom_line(aes(x = seq, y = Dia183,color = "Dia 183"))+
  geom_line(aes(x = seq, y = Dia244,color = "Dia 244"))+
  geom_line(aes(x = seq, y = Dia305,color = "Dia 305"))+
  geom_line(aes(x = seq, y = Dia366, color = "Dia 366"))+
  facet_grid(~ Maquina)+
  tema+
  scale_y_continuous(name = "Densidad")+
  theme(axis.ticks.y = element_blank(), axis.text.y = element_blank())+
  scale_x_continuous(name = "Probabilidad de éxito", breaks = seq(0,1,.25)) +
  scale_color_discrete(breaks=c("Dia 0", "Dia 244", "Dia 61", "Dia 305", "Dia 122", "Dia 366"))+
  theme(legend.background = element_blank(), legend.key.height = unit(.1, "cm"), plot.margin = margin(10, 10, 10, 10))
  theme(axis.title.x = element_text(size = 10), axis.title.y = element_text(size = 10), axis.text.x = element_text(size = 10), axis.text.y = element_text(size = 10))

plot3_greedy_tasa_dia

#Generación de 1000 años

set.seed(69)

datos_greedy_tasa_año = actualizar(366, 1000, "greedy_tasa")

#Obtención de la ganancia total al final de cada año

datos_greedy_tasa_filter = filter(datos_greedy_tasa_año, Dia == 366)

Exitos_greedy_tasa_totales = as.data.frame(rowSums(datos_greedy_tasa_filter[, 3:5]))
colnames(Exitos_greedy_tasa_totales) = "x"

#Histograma de las ganancias promedio para los años

```

```

ggplot(Exitos_greedy_tasa_totales, aes(x))+geom_histogram(fill = "#FF8F54", color = "black",
  scale_y_continuous(expand = expand_scale(add = c(0, 0)), limits = c(0,125), breaks = seq(0
  scale_x_continuous(breaks = seq(96,232,8), name = "Unidades monetarias")+ tema

#Cálculo de la ganancia promedio y los intervalos

Media_greedy_tasa = round(mean(Exitos_greedy_tasa_totales$x),2)

Q_greedy_tasa = sort(Exitos_greedy_tasa_totales$x)

Q_greedy_tasa_int = c(mean(Q_greedy_tasa[25], Q_greedy_tasa[26]), mean(Q_greedy_tasa[975], Q

```

Estrategia 3: Greedy con probabilidad a posteriori

```

#Generación de un dia para la estrategia

greedy_posterior = function(y, temp, e){
  maquina1 = (2 + y[1]) / (4 + y[4])
  maquina2 = (2 + y[2]) / (4 + y[5])
  maquina3 = (2 + y[3]) / (4 + y[6])
  cual = cual_maximo(c(maquina1, maquina2, maquina3))
  vector = c(rbinom(1, 1, prob[cual]), cual)
  return(vector)
}

#Generación de 1 año

set.seed(69)
datos_greedy_posterior_dia = actualizar(366, 1,"greedy_posterior")

#Gráfico de la elección diaria de máquina

plot1_greedy_posterior_dia = ggplot(datos_greedy_posterior_dia)+
  geom_line(aes(x = Dia, y = Maquina1Total, colour = "Maquina 1"))+
  geom_line(aes(x = Dia, y = Maquina2Total, colour = "Maquina 2"))+
  geom_line(aes(x = Dia, y = Maquina3Total, colour = "Maquina 3"))+
  tema+
  scale_y_continuous(name = "Cantidad de veces jugadas", breaks = c(seq(0, 320, 40), 366), 1.
  scale_x_continuous(name="Dias",breaks = seq(0, 366, 61)) +
  scale_color_discrete(breaks=c("Maquina 1", "Maquina 2", "Maquina 3"),type = c("purple", "g

```

```

theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))+
theme(legend.position = c(.25,.85), legend.background = element_blank(), legend.key = elem

#Gráfico de la ganancia diaria

plot2_greedy_posterior_dia = ggplot(datos_greedy_posterior_dia)+
  geom_line(aes(x = Dia, y = Maquina1 + Maquina2 + Maquina3), color = "red")+
  scale_y_continuous(name="Unidades monetarias ganadas", breaks = c(seq(0, 240, 30)), limits
  scale_x_continuous(name="Dias", breaks = seq(0, 366, 61))+tema+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))

#Unión de los dos gráficos

grid.arrange(plot1_greedy_posterior_dia, plot2_greedy_posterior_dia, ncol = 2)

#Generación de las curvas de las binomiales para los días seleccionados

set.seed(69)
datos_greedy_posterior_post = matrix(seq(0,1,.001), nrow = 3003)
datos_greedy_posterior_post = cbind(datos_greedy_posterior_post, dbeta(seq(0, 1, .001), 2, 2)
Maquina1_greedy_posterior = matrix(nrow = 1001, ncol = 6)
Maquina2_greedy_posterior = matrix(nrow = 1001, ncol = 6)
Maquina3_greedy_posterior = matrix(nrow = 1001, ncol = 6)

for (i in 1:6) {
  Maquina1_greedy_posterior[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_greedy_posterior_d
  Maquina2_greedy_posterior[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_greedy_posterior_d
  Maquina3_greedy_posterior[,i] = matrix(dbeta(seq(0,1,.001), 2+datos_greedy_posterior_dia$M

}
Maquinas_greedy_posterior = as.data.frame(rbind(Maquina1_greedy_posterior, Maquina2_greedy_p
Maquinas_greedy_posterior = cbind(Maquinas_greedy_posterior, matrix(rep(c("Maquina 1", "Maqu
datos_greedy_posterior_post = cbind(datos_greedy_posterior_post, Maquinas_greedy_posterior)
colnames(datos_greedy_posterior_post) = c("seq","Dia0", "Dia61", "Dia122", "Dia183", "Dia244

#Gráfico de las distribuciones a posteriori para cada día

plot3_greedy_posterior_dia = ggplot(datos_greedy_posterior_post, aes(col = factor(Maquina)))
  geom_line(aes(x = seq, y = Dia0 ,color = "Dia 0"))+

```

```

geom_line(aes(x = seq, y = Dia61,color = "Dia 61"))+
geom_line(aes(x = seq, y = Dia122,color = "Dia 122"))+
geom_line(aes(x = seq, y = Dia183,color = "Dia 183"))+
geom_line(aes(x = seq, y = Dia244,color = "Dia 244"))+
geom_line(aes(x = seq, y = Dia305,color = "Dia 305"))+
geom_line(aes(x = seq, y = Dia366, color ="Dia 366"))+
facet_grid(~ Maquina)+
tema+
scale_y_continuous(name = "Densidad")+
theme(axis.ticks.y = element_blank(), axis.text.y = element_blank())+
scale_x_continuous(name = "Probabilidad de éxito", breaks = seq(0,1,.25)) +
scale_color_discrete(breaks=c("Dia 0", "Dia 244", "Dia 61", "Dia 305", "Dia 122", "Dia 366"))+
theme(legend.background = element_blank(), legend.key.height = unit(.1, "cm"), plot.margin = margin(10, 10, 10, 10))
theme(axis.title.x = element_text(size = 10), axis.title.y = element_text(size = 10), axis.ticks.x = element_text(size = 10), axis.ticks.y = element_text(size = 10))

plot3_greedy_posterior_dia

#Generación de 1000 años

set.seed(69)

datos_greedy_posterior_año = actualizar(366, 1000, "greedy_posterior")

#Obtención de la ganancia total al final de cada año

datos_greedy_posterior_filter = filter(datos_greedy_posterior_año, Dia == 366)

Exitos_greedy_posterior_totales = as.data.frame(rowSums(datos_greedy_posterior_filter[, 3:5]))
colnames(Exitos_greedy_posterior_totales) = "x"

#Histograma de las ganancias promedio para los años

ggplot(Exitos_greedy_posterior_totales, aes(x))+geom_histogram(fill = "#FF8F54", color = "black")+
  scale_y_continuous(expand = expand_scale(add = c(0, 0)), name = "Frecuencia absoluta", limits = c(0, 100)) +
  scale_x_continuous(breaks = seq(108,220,8), name = "Unidades monetarias")+ tema

#Cálculo de la ganancia promedio y los intervalos

Media_greedy_posterior = round(mean(Exitos_greedy_posterior_totales$x),2)

Q_greedy_posterior = sort(Exitos_greedy_posterior_totales$x)

```

```
Q_greedy_posterior_int = c(mean(Q_greedy_posterior[25], Q_greedy_posterior[26]), mean(Q_greedy_posterior[27], Q_greedy_posterior[28]))
```

Estrategia 4: ϵ -greedy (con tasa observada)

```
#Generación de un dia para la estrategia

e_greedy = function(y, temp, e){
  which = rbinom(1, 1, e)
  if (which == 0) {
    return(greedy_tasa(y, temp ,e))
  } else {
    return(random(y, temp, e))
  }
}

# Generación de los 1000 años para cada valor de epsilon
vector = data.frame(matrix(nrow = 101, ncol = 1))

for (i in 0:100) {
  matriz = actualizar(366, 1000, "e_greedy", e = i/100)
  matriz = filter(matriz, Dia == 366)
  vector[i,] = sum(matriz[, 3:5])
}

xxx = as.matrix(vector/1000)
xxx = as.data.frame(cbind(xxx, seq(0, .99, .01)))
colnames(xxx) = c("x", "seq")

#Gráfico de las ganancias promedio para cada valor de epsilon

ggplot(xxx)+geom_line(aes(x = seq, y = x), color = "red")+
tema+
scale_y_continuous(name="Ganancias promedio",breaks = seq(0,280,40),limits = c(0,280))+
  scale_x_continuous(name="Epsilon",breaks = seq(0,1,.1))+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))+
  theme(legend.position = c(.85,.85), legend.background = element_blank(), legend.key = element_blank())

#Valor de epsilon
```



```

e = xxx$seq[which.max(xxx$x)]

#Generación de 1 año

set.seed(69)

epsilon = e

datos_e_greedy_dia = actualizar(366, 1,"e_greedy", e = epsilon)

#Gráfico de la elección diaria de máquina
plot1_e_greedy_dia = ggplot(datos_e_greedy_dia)+
  geom_line(aes(x = Dia, y = Maquina1Total, colour = "Maquina 1"))+
  geom_line(aes(x = Dia, y = Maquina2Total, colour = "Maquina 2"))+
  geom_line(aes(x = Dia, y = Maquina3Total, colour = "Maquina 3"))+
  tema+
  scale_y_continuous(name = "Cantidad de veces jugadas", breaks = c(seq(0, 320, 40), 366), limits = c(0, 366))+
  scale_x_continuous(name="Dias",breaks = seq(0, 366, 61)) +
  scale_color_discrete(breaks=c("Maquina 1", "Maquina 2", "Maquina 3"),type = c("purple", "green", "blue"))+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))+
  theme(legend.position = c(.25,.85), legend.background = element_blank(), legend.key = element_blank())

#Gráfico de la ganancia diaria

plot2_e_greedy_dia = ggplot(datos_e_greedy_dia)+
  geom_line(aes(x = Dia, y = Maquina1 + Maquina2 + Maquina3), color = "red")+
  scale_y_continuous(name="Unidades monetarias ganadas", breaks = c(seq(0, 240, 30)), limits = c(0, 300))+
  scale_x_continuous(name="Dias", breaks = seq(0, 366, 61))+tema+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))

#Unión de los dos gráficos

grid.arrange(plot1_e_greedy_dia, plot2_e_greedy_dia, ncol = 2)

#Generación de las curvas de las binomiales para los días seleccionados

set.seed(69)
datos_e_greedy_post = matrix(seq(0,1,.001), nrow = 3003)

```

```

datos_e_greedy_post = cbind(datos_e_greedy_post, dbeta(seq(0, 1, .001), 2, 2))
Maquina1_e_greedy = matrix(nrow = 1001, ncol = 6)
Maquina2_e_greedy = matrix(nrow = 1001, ncol = 6)
Maquina3_e_greedy = matrix(nrow = 1001, ncol = 6)

for (i in 1:6) {
  Maquina1_e_greedy[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_e_greedy_dia$Maquina1[61*i], 2), nrow = 1001, ncol = 1)
  Maquina2_e_greedy[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_e_greedy_dia$Maquina2[61*i], 2), nrow = 1001, ncol = 1)
  Maquina3_e_greedy[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_e_greedy_dia$Maquina3[61*i], 2), nrow = 1001, ncol = 1)
}

Maquinas_e_greedy = as.data.frame(rbind(Maquina1_e_greedy, Maquina2_e_greedy, Maquina3_e_greedy))
Maquinas_e_greedy = cbind(Maquinas_e_greedy, matrix(rep(c("Maquina 1", "Maquina 2", "Maquina 3"), nrow(Maquinas_e_greedy)), ncol = 3))
datos_e_greedy_post = cbind(datos_e_greedy_post, Maquinas_e_greedy)
colnames(datos_e_greedy_post) = c("seq", "Dia0", "Dia61", "Dia122", "Dia183", "Dia244", "Dia305", "Dia366")

#Gráfico de las distribuciones a posteriori para cada día

plot3_e_greedy_dia = ggplot(datos_e_greedy_post, aes(col = factor(Maquina))) +
  geom_line(aes(x = seq, y = Dia0 ,color = "Dia 0"))+
  geom_line(aes(x = seq, y = Dia61,color = "Dia 61"))+
  geom_line(aes(x = seq, y = Dia122,color = "Dia 122"))+
  geom_line(aes(x = seq, y = Dia183,color = "Dia 183"))+
  geom_line(aes(x = seq, y = Dia244,color = "Dia 244"))+
  geom_line(aes(x = seq, y = Dia305,color = "Dia 305"))+
  geom_line(aes(x = seq, y = Dia366, color = "Dia 366"))+
  facet_grid(~ Maquina)+
  tema+
  scale_y_continuous(name = "Densidad")+
  theme(axis.ticks.y = element_blank(), axis.text.y = element_blank())+
  scale_x_continuous(name = "Probabilidad de éxito", breaks = seq(0,1,.25)) +
  scale_color_discrete(breaks=c("Dia 0", "Dia 244", "Dia 61", "Dia 305", "Dia 122", "Dia 366"))+
  theme(legend.background = element_blank(), legend.key.height = unit(.1, "cm"), plot.margin = margin(10, 10, 10, 10))
  theme(axis.title.x = element_text(size = 10), axis.title.y = element_text(size = 10), axis.ticks.x = element_text(size = 10))

plot3_e_greedy_dia

#Generación de 1000 años

set.seed(69)

```

```

datos_e_greedy_año = actualizar(366, 1000, "e_greedy", e = epsilon)

#Obtención de la ganancia total al final de cada año

datos_e_greedy_filter = filter(datos_e_greedy_año, Dia == 366)

Exitos_e_greedy_totales = as.data.frame(rowSums(datos_e_greedy_filter[, 3:5]))
colnames(Exitos_e_greedy_totales) = "x"

#Histograma de las ganancias promedio para los años

ggplot(Exitos_e_greedy_totales, aes(x))+geom_histogram(fill = "#FF8F54", color = "black", bin
  scale_y_continuous(expand = expand_scale(add = c(0, 0)), name = "Frecuencia absoluta", lim
  scale_x_continuous(breaks = seq(108,220,8), name = "Unidades monetarias")+ tema

#Cálculo de la ganancia promedio y los intervalos

Media_e_greedy = round(mean(Exitos_e_greedy_totales$x),2)

Q_e_greedy = sort(Exitos_e_greedy_totales$x)

Q_e_greedy_int = c(mean(Q_e_greedy[25], Q_e_greedy[26]), mean(Q_e_greedy[975], Q_e_greedy[97

```

Estrategia 5: Softmax

```

#Generación de un día para la estrategia

softmax = function(tasas, temp, e){
  p = exp(tasas / temp) / sum(exp(tasas / temp))
  for (k in 1:length(p)) {
    if (is.na(p[k])) {
      p[k] = 0
    }
  }
  return(p)}

soft = function(y, temp, e){
  tasa = y[1:3] / y[4:6]
  for (k in 1:length(tasa)) {

```

```

    if (is.na(tasa[k])) {
      tasa[k] = 0
    }
  }
  p = softmax(tasa, temp)
  if (sum(p) == 0) {
    p = rep(1/3,3)
  }
  cual = sample(c(1, 2, 3), size = 1, prob = p)
  vector = c(rbinom(1, 1, prob[cual]), cual)
  return(vector)
}

#Generación de la tabla

s1 = round(c(prob[2]/prob[1],prob[2]/prob[3]),2)

s3 = round(c(softmax(prob,c(1),1)[2]/softmax(prob,1,1)[1], softmax(prob,1,1)[2]/softmax(prob,1,1)[1]),2)

s2 = round(c(softmax(prob,.02,1)[2]/softmax(prob,.04,1)[1], softmax(prob,.02,1)[2]/softmax(prob,.04,1)[1]),2)

s4 = round(c(softmax(prob,50,1)[2]/softmax(prob,25,1)[1], softmax(prob,50,1)[2]/softmax(prob,25,1)[1]),2)

s = as.data.frame(rbind(s1, s2, s3, s4))
rownames(s) = c("Originales", "Temperatura = 1/25", "Temperatura = 1", "Temperatura = 25" )

kable(s, col.names = c("Maquina 2 / Maquina 1", "Maquina 2 / Maquina 3")) %>%
  kable_material(c("striped", "hover"))

# Generación de los 1000 años para cada valor de la temperatura
vector2 = data.frame(matrix(nrow = 50, ncol = 1))

temperatura = c(seq(1/25,2,1/25))

for (i in 1:50) {
  matriz2 = actualizar(366, 1000, "softmax", temp = temperatura[i] )
  matriz2 = filter(matriz2, Dia == 366)
  vector2[i,] = sum(matriz2[, 3:5])
}

xxx2 = as.matrix(vector2/1000)

```

```

xxx2 = as.data.frame(cbind(xxx2, temperatura))
colnames(xxx2) = c("x", "seq")

#Gráfico de las ganancias promedio para cada valor de temperatura

ggplot(xxx2)+geom_line(aes(x = seq, y = x), color = "red")+
tema+
scale_y_continuous(name="Ganancias promedio",breaks = seq(0,280,40),limits = c(0,280))+
  scale_x_continuous(name="Temperatura")+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))

#Valor de la temperatura
temp = xxx2$seq[which.max(xxx2$x)]

#Generación de 1 año

set.seed(69)

temperatura = temp

datos_softmax_dia = actualizar(366, 1,"softmax", temp = temperatura)

#Gráfico de la elección diaria de máquina

plot1_softmax_dia = ggplot(datos_softmax_dia)+
  geom_line(aes(x = Dia, y = Maquina1Total, colour = "Maquina 1"))+
  geom_line(aes(x = Dia, y = Maquina2Total, colour = "Maquina 2"))+
  geom_line(aes(x = Dia, y = Maquina3Total, colour = "Maquina 3"))+
  tema+
  scale_y_continuous(name = "Cantidad de veces jugadas", breaks = seq(0, 280, 40), limits = c(0, 280))+
  scale_x_continuous(name="Dias",breaks = seq(0, 366, 61)) +
  scale_color_discrete(breaks=c("Maquina 1", "Maquina 2", "Maquina 3"),type = c("purple", "green", "red"))+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))+
  theme(legend.position = c(.25,.85), legend.background = element_blank(), legend.key = element_blank())

#Gráfico de la ganancia diaria

plot2_softmax_dia = ggplot(datos_softmax_dia)+

```

```

geom_line(aes(x = Dia, y = Maquina1 + Maquina2 + Maquina3), color = "red")+
scale_y_continuous(name="Unidades monetarias ganadas", breaks = c(seq(0, 240, 30)), limits = c(0, 240))+
scale_x_continuous(name="Dias", breaks = seq(0, 366, 61))+tema+
theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))

#Unión de los dos gráficos

grid.arrange(plot1_softmax_dia, plot2_softmax_dia, ncol = 2)

#Generación de las curvas de las binomiales para los días seleccionados

set.seed(69)
datos_softmax_post = matrix(seq(0,1,.001), nrow = 3003)
datos_softmax_post = cbind(datos_softmax_post, dbeta(seq(0, 1, .001), 2, 2))
Maquina1_softmax = matrix(nrow = 1001, ncol = 6)
Maquina2_softmax = matrix(nrow = 1001, ncol = 6)
Maquina3_softmax = matrix(nrow = 1001, ncol = 6)

for (i in 1:6) {
  Maquina1_softmax[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_softmax_dia$Maquina1[61*i],
  Maquina2_softmax[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_softmax_dia$Maquina2[61*i],
  Maquina3_softmax[,i] = matrix(dbeta(seq(0,1,.001), 2+datos_softmax_dia$Maquina3[61*i], 2-datos_softmax_dia$Maquina3[61*i], 2))

}
Maquinas_softmax = as.data.frame(rbind(Maquina1_softmax, Maquina2_softmax, Maquina3_softmax))
Maquinas_softmax = cbind(Maquinas_softmax, matrix(rep(c("Maquina 1", "Maquina 2", "Maquina 3"), nrow(Maquinas_softmax)), ncol = 3))
datos_softmax_post = cbind(datos_softmax_post, Maquinas_softmax)
colnames(datos_softmax_post) = c("seq","Dia0", "Dia61", "Dia122", "Dia183", "Dia244", "Dia305", "Dia366")

#Gráfico de las distribuciones a posteriori para cada día

plot3_softmax_dia = ggplot(datos_softmax_post, aes(col = factor(Maquina))) +
  geom_line(aes(x = seq, y = Dia0 ,color = "Dia 0"))+
  geom_line(aes(x = seq, y = Dia61,color = "Dia 61"))+
  geom_line(aes(x = seq, y = Dia122,color = "Dia 122"))+
  geom_line(aes(x = seq, y = Dia183,color = "Dia 183"))+
  geom_line(aes(x = seq, y = Dia244,color = "Dia 244"))+
  geom_line(aes(x = seq, y = Dia305,color = "Dia 305"))+
  geom_line(aes(x = seq, y = Dia366, color = "Dia 366"))+
  facet_grid(~ Maquina)+

```

```

tema+
scale_y_continuous(name = "Densidad")+
theme(axis.ticks.y = element_blank(), axis.text.y = element_blank())+
scale_x_continuous(name = "Probabilidad de éxito", breaks = seq(0,1,.25)) +
scale_color_discrete(breaks=c("Dia 0", "Dia 244", "Dia 61", "Dia 305", "Dia 122", "Dia 366"))
theme(legend.background = element_blank(), legend.key.height = unit(.1, "cm"), plot.margin = margin(10, 10, 10, 10))
theme(axis.title.x = element_text(size = 10), axis.title.y = element_text(size = 10), axis.text.x = element_text(size = 10), axis.text.y = element_text(size = 10))

plot3_softmax_dia

#Generación de 1000 años

set.seed(69)

temperatura = temp

datos_softmax_año = actualizar(366, 1000, "softmax", temp = temperatura)

#Obtención de la ganancia total al final de cada año

datos_softmax_filter = filter(datos_softmax_año, Dia == 366)

Exitos_softmax_totales = as.data.frame(rowSums(datos_softmax_filter[, 3:5]))
colnames(EXITOS_softmax_totales) = "x"

#Histograma de las ganancias promedio para los años

ggplot(EXITOS_softmax_totales, aes(x))+geom_histogram(fill = "#FF8F54", color = "black", binwidth = 10,
  scale_y_continuous(expand = expand_scale(add = c(0, 0)), name = "Frecuencia absoluta", limits = c(0, 100)),
  scale_x_continuous(breaks = seq(108,220,8), name = "Unidades monetarias")+ tema

#Cálculo de la ganancia promedio y los intervalos

Media_softmax = round(mean(EXITOS_softmax_totales$x),2)

Q_softmax = sort(EXITOS_softmax_totales$x)

Q_softmax_int = c(mean(Q_softmax[25], Q_softmax[26]), mean(Q_softmax[975], Q_softmax[976]))

```

Estrategia 6: Upper-bound

```

#Generación de un día para la estrategia

upper = function(y, temp, e){
  maquina1 = qbeta(.975, 2 + y[1], 2 + y[4] - y[1])
  maquina2 = qbeta(.975, 2 + y[2], 2 + y[5] - y[2])
  maquina3 = qbeta(.975, 2 + y[3], 2 + y[6] - y[3])
  cual = cual_maximo(c(maquina1, maquina2, maquina3))
  vector = c(rbinom(1, 1, prob[cual]), cual)
  return(vector)
}

#Generación de 1 año

set.seed(69)

datos_upper_dia = actualizar(366, 1,"upper")

#Gráfico de la elección diaria de máquina

plot1_upper_dia = ggplot(datos_upper_dia)+
  geom_line(aes(x = Dia, y = Maquina1Total, colour = "Maquina 1"))+
  geom_line(aes(x = Dia, y = Maquina2Total, colour = "Maquina 2"))+
  geom_line(aes(x = Dia, y = Maquina3Total, colour = "Maquina 3"))+
  tema+
  scale_y_continuous(name = "Cantidad de veces jugadas", breaks = c(seq(0, 320, 40), 366), limits = c(0, 366))+
  scale_x_continuous(name="Dias",breaks = seq(0, 366, 61)) +
  scale_color_discrete(breaks=c("Maquina 1", "Maquina 2", "Maquina 3"),type = c("purple", "green", "red"))+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))+
  theme(legend.position = c(.25,.85), legend.background = element_blank(), legend.key = element_blank())

#Gráfico de la ganancia diaria

plot2_upper_dia = ggplot(datos_upper_dia)+
  geom_line(aes(x = Dia, y = Maquina1 + Maquina2 + Maquina3), color = "red")+
  scale_y_continuous(name="Unidades monetarias ganadas", breaks = c(seq(0, 240, 30)), limits = c(0, 300))+
  scale_x_continuous(name="Dias", breaks = seq(0, 366, 61))+tema+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))

#Unión de los dos gráficos

grid.arrange(plot1_upper_dia, plot2_upper_dia, ncol = 2)

```



```

#Generación de las curvas de las binomiales para los días seleccionados

set.seed(69)
datos_upper_post = matrix(seq(0,1,.001), nrow = 3003)
datos_upper_post = cbind(datos_upper_post, dbeta(seq(0, 1, .001), 2, 2))
Maquina1_upper = matrix(nrow = 1001, ncol = 6)
Maquina2_upper = matrix(nrow = 1001, ncol = 6)
Maquina3_upper = matrix(nrow = 1001, ncol = 6)

for (i in 1:6) {
  Maquina1_upper[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_upper_dia$Maquina1[61*i], 2-datos_upper_dia$Maquina1[61*i]), nrow = 1001)
  Maquina2_upper[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_upper_dia$Maquina2[61*i], 2-datos_upper_dia$Maquina2[61*i]), nrow = 1001)
  Maquina3_upper[,i] = matrix(dbeta(seq(0,1,.001), 2+datos_upper_dia$Maquina3[61*i], 2-datos_upper_dia$Maquina3[61*i]), nrow = 1001)
}
Maquinas_upper = as.data.frame(rbind(Maquina1_upper, Maquina2_upper, Maquina3_upper))
Maquinas_upper = cbind(Maquinas_upper, matrix(rep(c("Maquina 1", "Maquina 2", "Maquina 3"), 1001), 1, 3))
datos_upper_post = cbind(datos_upper_post, Maquinas_upper)
colnames(datos_upper_post) = c("seq","Dia0", "Dia61", "Dia122", "Dia183", "Dia244", "Dia305", "Dia366")

#Gráfico de las distribuciones a posteriori para cada día

plot3_upper_dia = ggplot(datos_upper_post, aes(col = factor(Maquina))) +
  geom_line(aes(x = seq, y = Dia0 ,color = "Dia 0"))+
  geom_line(aes(x = seq, y = Dia61,color = "Dia 61"))+
  geom_line(aes(x = seq, y = Dia122,color = "Dia 122"))+
  geom_line(aes(x = seq, y = Dia183,color = "Dia 183"))+
  geom_line(aes(x = seq, y = Dia244,color = "Dia 244"))+
  geom_line(aes(x = seq, y = Dia305,color = "Dia 305"))+
  geom_line(aes(x = seq, y = Dia366, color = "Dia 366"))+
  facet_grid(~ Maquina)+
  tema+
  scale_y_continuous(name = "Densidad")+
  theme(axis.ticks.y = element_blank(), axis.text.y = element_blank())+
  scale_x_continuous(name = "Probabilidad de éxito", breaks = seq(0,1,.25)) +
  scale_color_discrete(breaks=c("Dia 0", "Dia 244", "Dia 61", "Dia 305", "Dia 122", "Dia 366"))+
  theme(legend.background = element_blank(), legend.key.height = unit(.1, "cm"), plot.margin = margin(10, 10, 10, 10))
  theme(axis.title.x = element_text(size = 10), axis.title.y = element_text(size = 10), axis.ticks.x = element_text(size = 10))

plot3_upper_dia

```

```

#Generación de 1000 años
set.seed(69)

datos_upper_año = actualizar(366, 1000, "upper")

#Obtención de la ganancia total al final de cada año

datos_upper_filter = filter(datos_upper_año, Dia == 366)

Exitos_upper_totales = as.data.frame(rowSums(datos_upper_filter[, 3:5]))
colnames(Exitos_upper_totales) = "x"

#Histograma de las ganancias promedio para los años

ggplot(Exitos_upper_totales, aes(x))+geom_histogram(fill = "#FF8F54", color = "black", binwidth = 10,
  scale_y_continuous(expand = expand_scale(add = c(0, 0)), name = "Frecuencia absoluta", limits = c(0, 1000)),
  scale_x_continuous(breaks = seq(108,220,8), name = "Unidades monetarias")+ tema

#Cálculo de la ganancia promedio y los intervalos

Media_upper = round(mean(Exitos_upper_totales$x),2)

Q_upper = sort(Exitos_upper_totales$x)

Q_upper_int = c(mean(Q_upper[25], Q_upper[26]), mean(Q_upper[975], Q_upper[976]))

```

Estrategia 7: Thompson sampling

```

#Generación de un día para la estrategia

thompson = function(y, temp, e){
  maquina1 = rbeta(1, 2 + y[1], 2 + y[4] - y[1])
  maquina2 = rbeta(1, 2 + y[2], 2 + y[5] - y[2])
  maquina3 = rbeta(1, 2 + y[3], 2 + y[6] - y[3])
  cual = cual_maximo(c(maquina1, maquina2, maquina3))
  vector = c(rbinom(1, 1, prob[cual]), cual)
  return(vector)
}

#Generación de 1 año

```

```

set.seed(69)

datos_thompson_dia = actualizar(366, 1,"thompson")

#Gráfico de la elección diaria de máquina

plot1_thompson_dia = ggplot(datos_thompson_dia)+
  geom_line(aes(x = Dia, y = Maquina1Total, colour = "Maquina 1"))+
  geom_line(aes(x = Dia, y = Maquina2Total, colour = "Maquina 2"))+
  geom_line(aes(x = Dia, y = Maquina3Total, colour = "Maquina 3"))+
  tema+
  scale_y_continuous(name = "Cantidad de veces jugadas", breaks = seq(0, 280, 40), limits = c(0, 280)) +
  scale_x_continuous(name="Dias",breaks = seq(0, 366, 61)) +
  scale_color_discrete(breaks=c("Maquina 1", "Maquina 2", "Maquina 3"),type = c("purple", "green", "red"))+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))+
  theme(legend.position = c(.25,.85), legend.background = element_blank(), legend.key = element_blank())

#Gráfico de la ganancia diaria

plot2_thompson_dia = ggplot(datos_thompson_dia)+
  geom_line(aes(x = Dia, y = Maquina1 + Maquina2 + Maquina3), color = "red")+
  scale_y_continuous(name="Unidades monetarias ganadas", breaks = c(seq(0, 240, 30)), limits = c(0, 240)) +
  scale_x_continuous(name="Dias", breaks = seq(0, 366, 61))+tema+
  theme(plot.margin = margin(1.5, .3, 1.5, .3, "cm"))

#Unión de los dos gráficos

grid.arrange(plot1_thompson_dia, plot2_thompson_dia, ncol = 2)

#Generación de las curvas de las binomiales para los días seleccionados

set.seed(69)
datos_thompson_post = matrix(seq(0,1,.001), nrow = 3003)
datos_thompson_post = cbind(datos_thompson_post, dbeta(seq(0, 1, .001), 2, 2))
Maquina1_thompson = matrix(nrow = 1001, ncol = 6)
Maquina2_thompson = matrix(nrow = 1001, ncol = 6)
Maquina3_thompson = matrix(nrow = 1001, ncol = 6)

for (i in 1:6) {
  Maquina1_thompson[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_thompson_dia$Maquina1[61*i], 2), nrow = 1001)
  Maquina2_thompson[,i] = matrix(dbeta(seq(0, 1, .001), 2 + datos_thompson_dia$Maquina2[61*i], 2), nrow = 1001)
  Maquina3_thompson[,i] = matrix(dbeta(seq(0,1,.001), 2+datos_thompson_dia$Maquina3[61*i], 2), nrow = 1001)
}

```

```

}
Maquinas_thompson = as.data.frame(rbind(Maquina1_thompson, Maquina2_thompson, Maquina3_thompson))
Maquinas_thompson = cbind(Maquinas_thompson, matrix(rep(c("Maquina 1", "Maquina 2", "Maquina 3"), nrow(Maquinas_thompson)), ncol = 3))
datos_thompson_post = cbind(datos_thompson_post, Maquinas_thompson)
colnames(datos_thompson_post) = c("seq", "Dia0", "Dia61", "Dia122", "Dia183", "Dia244", "Dia305", "Dia366")

#Gráfico de las distribuciones a posteriori para cada día

plot3_thompson_dia = ggplot(datos_thompson_post, aes(col = factor(Maquina))) +
  geom_line(aes(x = seq, y = Dia0 ,color = "Dia 0"))+
  geom_line(aes(x = seq, y = Dia61,color = "Dia 61"))+
  geom_line(aes(x = seq, y = Dia122,color = "Dia 122"))+
  geom_line(aes(x = seq, y = Dia183,color = "Dia 183"))+
  geom_line(aes(x = seq, y = Dia244,color = "Dia 244"))+
  geom_line(aes(x = seq, y = Dia305,color = "Dia 305"))+
  geom_line(aes(x = seq, y = Dia366, color = "Dia 366"))+
  facet_grid(~ Maquina)+
  tema+
  scale_y_continuous(name = "Densidad")+
  theme(axis.ticks.y = element_blank(), axis.text.y = element_blank())+
  scale_x_continuous(name = "Probabilidad de éxito", breaks = seq(0,1,.25)) +
  scale_color_discrete(breaks=c("Dia 0", "Dia 244", "Dia 61", "Dia 305", "Dia 122", "Dia 366"))+
  theme(legend.background = element_blank(), legend.key.height = unit(.1, "cm"), plot.margin = margin(10, 10, 10, 10))
  theme(axis.title.x = element_text(size = 10), axis.title.y = element_text(size = 10), axis.ticks.x = element_text(size = 10))

plot3_thompson_dia

#Generación de 1000 años

set.seed(69)

datos_thompson_año = actualizar(366, 1000, "thompson")

#Obtención de la ganancia total al final de cada año

datos_thompson_filter = filter(datos_thompson_año, Dia == 366)

Exitos_thompson_totales = as.data.frame(rowSums(datos_thompson_filter[, 3:5]))
colnames(Exitos_thompson_totales) = "x"

#Histograma de las ganancias promedio para los años

```

```

ggplot(Exitos_thompson_totales, aes(x))+geom_histogram(fill = "#FF8F54", color = "black", bin
  scale_y_continuous(expand = expand_scale(add = c(0, 0)), name = "Frecuencia absoluta", lim
  scale_x_continuous(breaks = seq(108,220,8), name = "Unidades monetarias")+ tema

#Cálculo de la ganancia promedio y los intervalos

Media_thompson = round(mean(Exitos_thompson_totales$x),2)

Q_thompson = sort(Exitos_thompson_totales$x)

Q_thompson_int = c(mean(Q_thompson[25], Q_thompson[26]), mean(Q_thompson[975], Q_thompson[976]))

```