Estudio Comparativo de Algoritmos en el Problema de los K-Brazos

Esteban Becerra, Carlos Cruzado, Anastasiya Ruzhytska
esteban.becerraf@um.es, carlos.cruzadoe1@um.es, anastasiya.r.r@um.es

6 de marzo de 2025

Resumen

Este trabajo presenta un estudio comparativo de diversos algoritmos utilizados en el problema de los K-Brazos (Multi-Armed Bandit). Se analizan los métodos *Epsilon-Greedy*, *UCB1*, *UCB2*, *Softmax* y *Gradient Bandit*, aplicados a diferentes distribuciones de recompensa: Normal, Bernoulli y Binomial.

Se presentan experimentos que evalúan el rendimiento de cada método en términos de recompensa promedio, porcentaje de selección del brazo óptimo y *regret* acumulado. Los resultados indican que la efectividad de cada algoritmo varía según la distribución de recompensas y el contexto de aplicación.

1. Introducción

El problema del *K-Brazos* es fundamental en el campo del Aprendizaje por Refuerzo, modelando situaciones en las que un agente debe tomar decisiones repetidas en un entorno incierto. El objetivo principal es encontrar una estrategia de exploración-explotación eficiente que maximice la recompensa total obtenida a lo largo del tiempo.

Este problema tiene múltiples aplicaciones, como la selección de anuncios en publicidad digital, sistemas de recomendación, ensayos clínicos, y optimización de estrategias financieras. La dificultad radica en encontrar un equilibrio entre explorar nuevas opciones y explotar las que ya han demostrado ser óptimas.

En este trabajo, se estudian cinco algoritmos para resolver el problema de los K-Brazos, comparándolos en diferentes escenarios de recompensa.

2. Fundamentos Teóricos

2.1. Definición Matemática

Sea un conjunto de K brazos, cada uno con una distribución de recompensa desconocida $P_i(r)$, donde $i \in \{1, 2, \ldots, K\}$. En cada paso temporal t, el agente selecciona un brazo A_t y recibe una recompensa R_t extraída de la distribución correspondiente.

El objetivo es maximizar la recompensa acumulada:

$$G_T = \sum_{t=1}^{T} R_t \tag{1}$$

Minimizando el *regret*:

$$R_T = T\mu^* - \sum_{t=1}^{T} \mathbb{E}[R_t] \tag{2}$$

Donde $\mu^* = \max_i \mathbb{E}[R_i]$ representa la recompensa esperada óptima.

2.2. Distribuciones de Recompensa

Se han implementado tres tipos de distribuciones de recompensa para evaluar los algoritmos:

- **Bernoulli:** $R_i \sim Bernoulli(p_i)$. Se usa cuando las recompensas son binarias (éxito o fracaso).
- **Binomial:** $R_i \sim Bin(n, p_i)$. Extiende la distribución Bernoulli a múltiples intentos.
- **Normal:** $R_i \sim \mathcal{N}(\mu_i, \sigma^2)$. Modela entornos con variabilidad continua.

3. Implementación de Algoritmos

3.1. Epsilon-Greedy

El algoritmo ϵ-Greedy es una estrategia de exploración-explotación que selecciona la mejor acción conocida con probabilidad $1 - \epsilon$ y una acción aleatoria con probabilidad ϵ. Es un método simple que permite evitar el sesgo excesivo hacia la explotación temprana de los brazos con mayores recompensas observadas.

Matemáticamente, en cada paso t, el algoritmo elige la acción A_t de la siguiente manera:

$$A_t = \begin{cases} \arg\max_a Q_t(a) & \text{con probabilidad } 1 - \epsilon \\ \text{acción aleatoria} & \text{con probabilidad } \epsilon \end{cases} \tag{3}$$

Donde $Q_t(a)$ es el valor estimado de la acción a en el tiempo t. Se han evaluado valores de $\epsilon = \{0,0, 0,01, 0,1\}$ para analizar el impacto de diferentes niveles de exploración.

3.2. Métodos UCB (Upper Confidence Bound)

Los métodos UCB buscan balancear exploración y explotación de manera más inteligente que ϵ-Greedy, utilizando un criterio basado en intervalos de confianza para seleccionar la acción con el mayor potencial.

- **UCB1**: Se basa en la desigualdad de Hoeffding para construir una cota superior de confianza en la recom-

pensa esperada de cada brazo. La regla de selección está dada por:

$$A_t = \arg\max_a \left[Q_t(a) + \sqrt{\frac{2 \ln t}{N_t(a)}} \right] \quad (4)$$

donde $N_t(a)$ es el número de veces que se ha seleccionado el brazo a hasta el instante t.

- **UCB2**: Es una variante de UCB1 que introduce un mecanismo basado en fases. Define intervalos de ejecución $\tau(k)$ para cada brazo y actualiza su confianza con menor frecuencia, reduciendo la sobre-exploración:

$$A_t = \arg\max_a \left[Q_t(a) + \sqrt{\frac{(1+\alpha)\ln(e \cdot t/\tau(k))}{2\tau(k)}} \right] \quad (5)$$

donde $\alpha \in (0, 1)$ controla la velocidad de exploración y $\tau(k)$ define la cantidad de veces que un brazo es seleccionado en cada fase.

3.3. Softmax y Gradient Bandit

Estos métodos implementan estrategias probabilísticas de selección de acciones basadas en valores estimados de recompensa.

- **Softmax**: En lugar de seleccionar directamente el brazo con mayor recompensa estimada, asigna probabilidades de selección según una función exponencial normalizada:

$$P(A_t = a) = \frac{\exp(Q_t(a)/\tau)}{\sum_b \exp(Q_t(b)/\tau)} \quad (6)$$

donde τ es un parámetro de temperatura que controla el grado de exploración. Valores altos de τ favorecen una selección más uniforme entre todos los brazos, mientras que valores bajos tienden a una selección más explotativa.

- **Gradient Bandit**: En lugar de estimar valores de acción directamente, aprende una preferencia relativa para cada brazo, actualizándola mediante ascenso en gradiente en función de la recompensa obtenida:

$$H_{t+1}(A_t) = H_t(A_t) + \alpha(R_t - \bar{R}_t)(1 - P(A_t)) \quad (7)$$
$$H_{t+1}(a) = H_t(a) - \alpha(R_t - \bar{R}_t)P(a), \quad \forall a \neq A_t \quad (8)$$

donde $H_t(a)$ representa la preferencia del brazo a en el tiempo t, α es la tasa de aprendizaje, y $\bar{R}_t$ es la recompensa promedio obtenida hasta el instante t.

4. Resultados

4.1. Comparación de Rendimiento

Para evaluar el rendimiento de los distintos algoritmos, se realizaron experimentos con $K = 10$ brazos, cada uno siguiendo distribuciones de recompensa Normal, Bernoulli y Binomial. Cada configuración de algoritmo se ejecutó durante 1000 pasos y se repitió 500 veces para obtener métricas confiables.

A continuación, se presentan los resultados específicos para cada algoritmo y tipo de distribución.

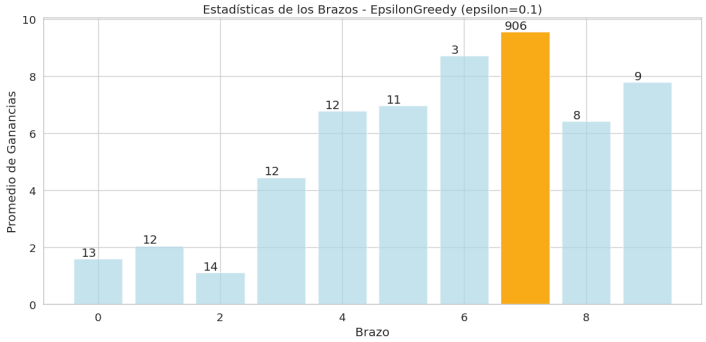

Figura 1: Estadísticas de selección de brazos para e-greedy con $\epsilon = 0{,}1$ en distribución Normal

4.1.1. Epsilon-Greedy

El algoritmo *Epsilon-Greedy* mostró diferencias significativas dependiendo de la elección del parámetro ϵ.

- **Distribución Normal**: Como se muestra en la Figura Figure 1, $\epsilon = 0{,}1$ alcanza rápidamente una alta tasa de selección del brazo óptimo, lo que se traduce en un bajo *regret* acumulado (Figura Figure 2). Sin embargo, $\epsilon = 0{,}01$ converge más lentamente, y $\epsilon = 0$ (pura explotación) queda atrapado en un brazo subóptimo.
- **Distribución Bernoulli**: Aquí, el rendimiento de $\epsilon = 0{,}1$ es más variable debido a la naturaleza discreta de las recompensas. Se observa que $\epsilon = 0{,}01$ logra un *regret* menor en algunos casos, pero su tasa de selección del brazo óptimo es inferior a la de $\epsilon = 0{,}1$.
- **Distribución Binomial**: En este caso, $\epsilon = 0{,}1$ tarda más en converger en comparación con la Normal, aunque sigue siendo superior a $\epsilon = 0{,}01$. La exploración es clave en esta distribución debido a la alta varianza de las recompensas.

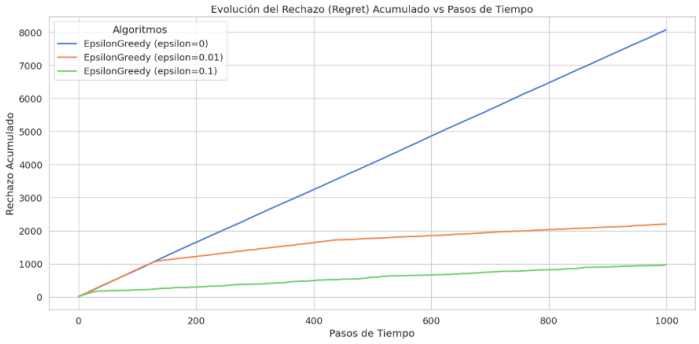

Figura 2: Regret acumulado para e-greedy en distribución Normal

4.1.2. UCB1 y UCB2

Los métodos *Upper Confidence Bound* mostraron un comportamiento más robusto en todas las distribuciones.

- **Distribución Normal**: La Figura Figure 3 muestra que UCB1 con $c = 0{,}5$ logra la mejor convergencia en términos de recompensa promedio, mientras que valores más altos ($c = 1$ o $c = 2$) provocan exploración excesiva, retrasando la estabilización en el brazo óptimo. UCB2 con

$\alpha = 0{,}9$ también alcanza buenas recompensas, pero su convergencia es más lenta debido a su estrategia basada en fases.

- **Distribución Bernoulli**: En este caso, UCB2 con $\alpha = 0{,}5$ supera a UCB1 en la mayoría de los escenarios, logrando menor *regret* acumulado (Figura Figure 4). Esto se debe a que la exploración controlada por fases de UCB2 permite identificar mejor los brazos óptimos en una distribución discreta.
- **Distribución Binomial**: UCB1 con $c = 0{,}5$ sigue siendo competitivo, aunque UCB2 con $\alpha = 0{,}9$ obtiene un rendimiento más estable debido a la varianza en las recompensas. En contraste, $\alpha = 0{,}1$ presenta un desempeño inferior, ya que explora demasiado poco y tarda más en converger.

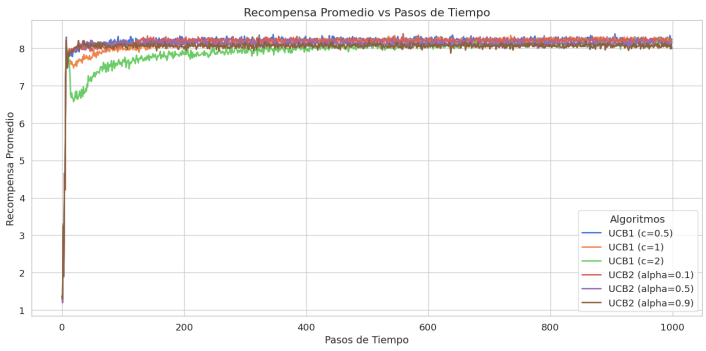

Figura 3: Recompensa promedio para UCB en distribución Normal

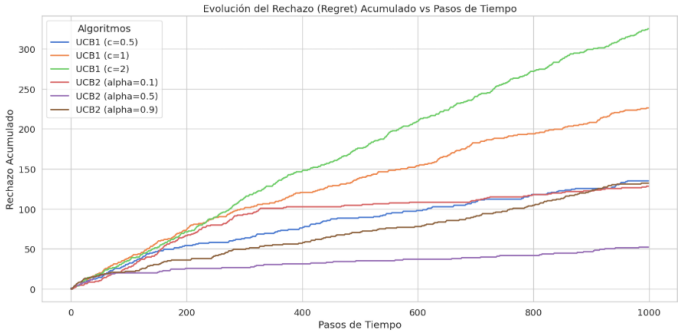

Figura 4: Regret acumulado para UCB en distribución Bernoulli

4.1.3. Softmax

El rendimiento de *Softmax* estuvo altamente influenciado por el parámetro de temperatura τ.

- **Distribución Normal**: Como se observa en la Figura Figure 5, valores de τ cercanos a 1 lograron el mejor equilibrio entre exploración y explotación. Un τ demasiado bajo causó un estancamiento temprano en brazos subóptimos. Destaca aquí $\tau = 1$.
- **Distribución Bernoulli**: Softmax con $\tau = 0{,}1$ mostró menor efectividad en comparación con UCB2 y Epsilon-Greedy, debido a la naturaleza discreta de la recompensa, pero aun así fue la mejor opción en cuanto a Softmax.
- **Distribución Binomial**: El impacto de τ fue similar al caso Normal. Aquí, valores altos de τ provocaron un *regret* bastante bajo (Figura Figure 6).

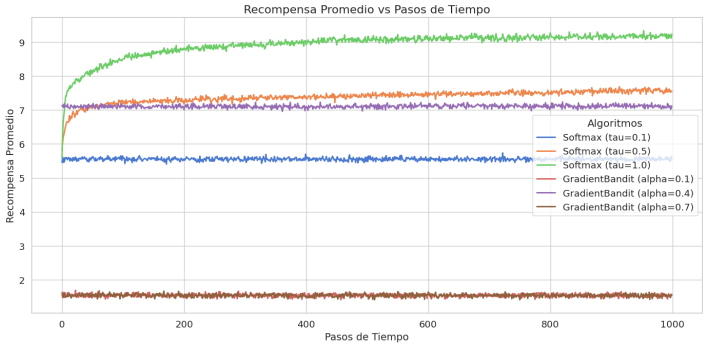

Figura 5: Recompensa promedio para Ascenso de Gradiente en distribución Normal

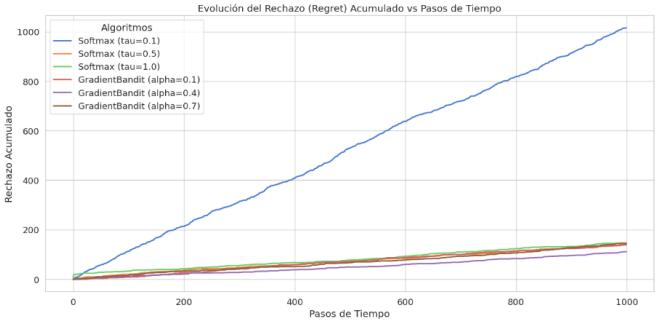

Figura 6: Regret acumulado para Ascenso de Gradiente en distribución Binomial

4.1.4. Gradient Bandit

El método *Gradient Bandit* también tuvo un rendimiento bastante aceptable.

- **Distribución Normal**: Se observó que $\alpha = 0{,}4$ lograba el mejor balance entre estabilidad y convergencia (Figura Figure 7). Sin embargo, los otros valores de α hacían que la recompensa promedio fuera muy baja.
- **Distribución Bernoulli**: Gradient Bandit con $\alpha = 0{,}7$ y $\alpha = 0{,}1$ tuvieron un rendimiento muy bueno, con una alta recompensa y regret muy bajo.
- **Distribución Binomial**: Resultados parecidos a los de Bernoulli, muy buen rendimiento para los tres valores de α, como se puede ver en la imagen Figure 8.

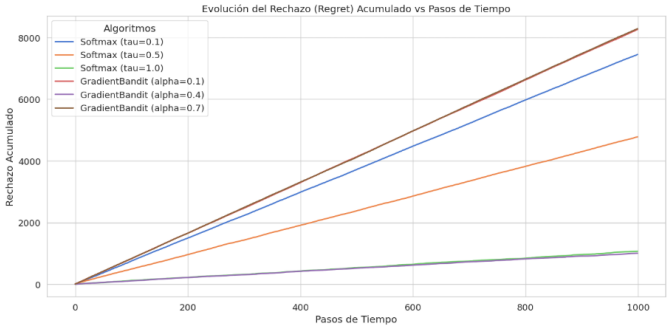

Figura 7: Regret acumulado para Ascenso de Gradiente en distribución Normal

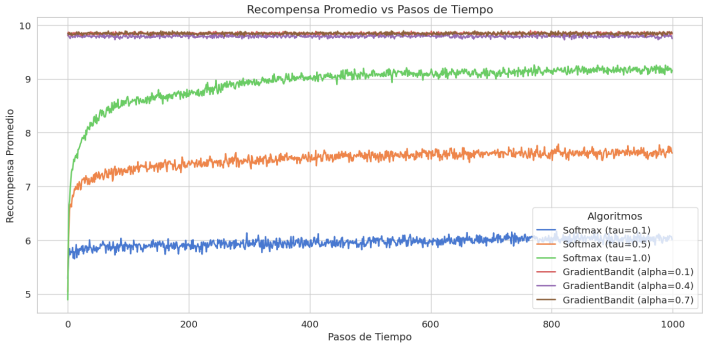

Figura 8: Recompensa promedio para Ascenso de Gradiente en distribución Binomial

5. Conclusiones

Con base en los experimentos realizados, se pueden extraer las siguientes conclusiones clave:

- **Epsilon-Greedy**: $\epsilon = 0{,}1$ es la mejor configuración para todas las distribuciones.
- **UCB1 y UCB2**: UCB1 con $c = 0{,}5$ es óptimo en distribuciones Normales, mientras que UCB2 con $\alpha = 0{,}5$ destaca en Bernoulli debido a su estrategia de exploración adaptativa y con $\alpha = 0{,}9$ en Binomial.
- **Softmax**: Funciona mejor en Normal y Binomial cuando $\tau = 1$. En Bernoulli, sin embargo, su rendimiento es inferior debido a la exploración excesiva.
- **Gradient Bandit**: $\alpha = 0{,}4$ ofrece un desempeño adecuado en Normal, aunque en el resto es preferible $\alpha = 0{,}7$.

En conclusión, la elección del algoritmo óptimo depende del tipo de distribución de recompensa y del balance entre exploración y explotación.

6. Referencias

Referencias

[1] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. MIT Press.

[2] Lattimore, T., & Szepesvári, C. (2020). *Bandit Algorithms*. Cambridge University Press.

[3] Transparencias de clase sobre el problema del Bandido Multi-Brazo, Universidad de Murcia.