NOMBRE: Bruno Farfán

SECCIÓN: 1

PUNTAJE:

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC3675 — Aprendizaje Reforzado — 1' 2025

Tarea 1 – Respuesta Pregunta 1

a)

Resultados replicar el experimento de la Figura 2.2 del libro:

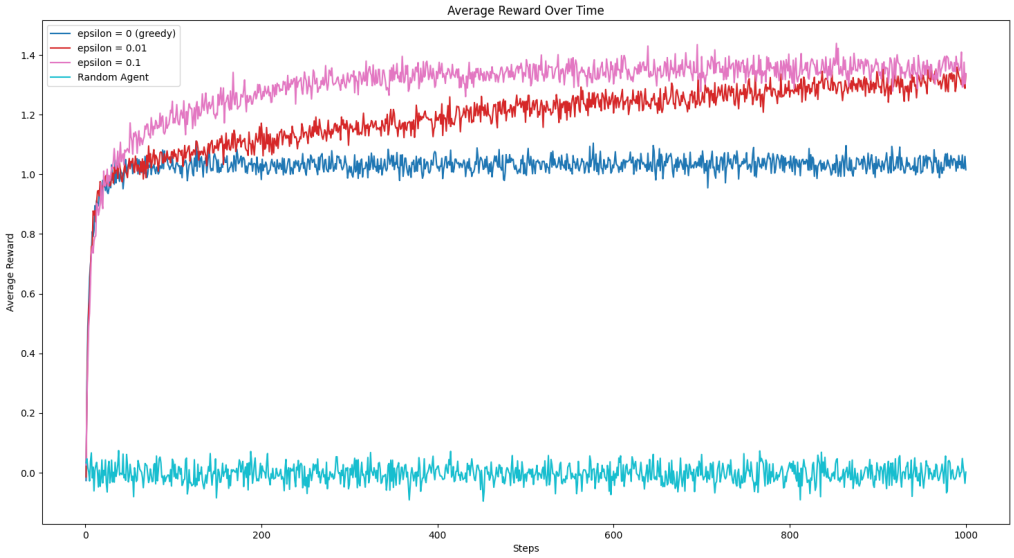

Figure 1: Recompensa promedio de los agentes random y de SimpleBandit con diferentes valores de ϵ

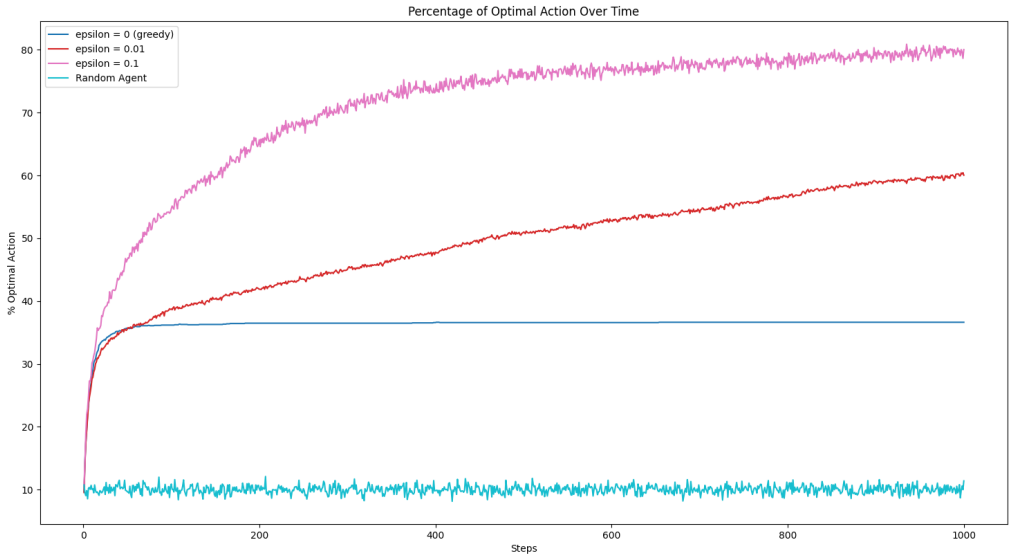

Figure 2: Porcentaje de veces que se elige la acción óptima

Los resultados obtenidos efectivamente son los esperados, ya que los gráficos son muy similares a los del libro, y estos hacen sentido con los respectivos valores de ϵ, ya que se observa una mayor velocidad en el aprendizaje usando $\epsilon = 0.1$ que con $\epsilon = 0.01$.

b)

Para que el porcentaje de veces que se elige la acción óptima sea cercano al 90%, se necesita que el agente elija la opción óptima cada vez que no se opta por explorar. Sin embargo, como el agente aún está aprendiendo, algunas veces ocurrirá que eligirá una acción sub-óptima pensando que es la óptima, de esta forma bajando levemente el porcentaje de elección de la acción óptima.
En el infinito se esperaría que el agente eligiera la acción óptima el $90\% + \frac{1}{10} \cdot 10\% = 91\%$ de las veces.

c)

Se implementa el agente con un *step size* fijo y se replica el experimento de la Figura 2.5 del libro. El resultado es el siguiente:

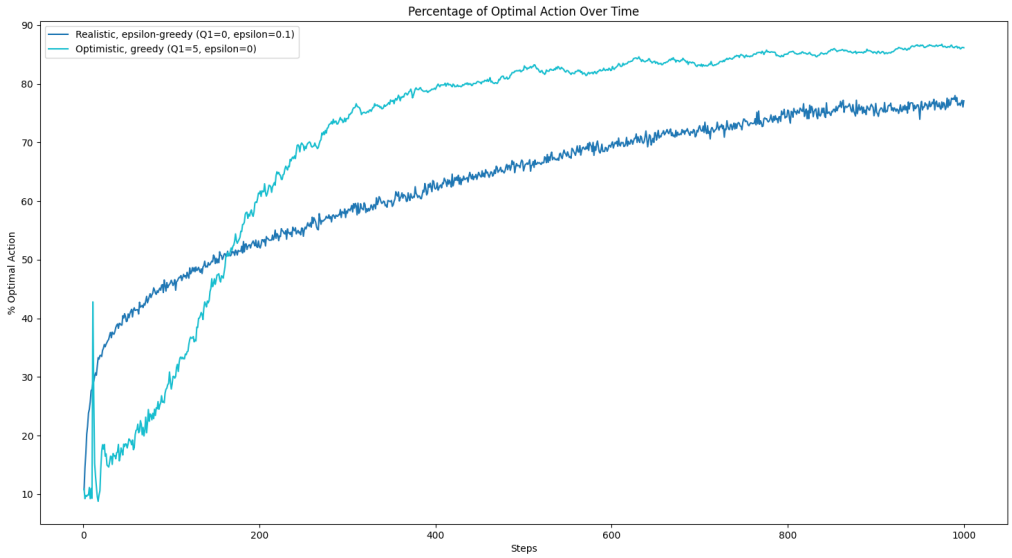

Figure 3: Porcentaje de veces que se elige la acción óptima

Nuevamente los resultados fueron los esperados, ya que coinciden con lo obtenido en el libro. Además haciendo sentido que el agente *greedy* explore mucho involuntariamente, debido a que al principio constantemente sobreestima lo que obtendrá de cada acción.

d)

En la inicialización optimista con $Q_1 = 5$ y $\epsilon = 0$, se observa una rápida subida seguida de una caída en la selección de la acción óptima. Esto ocurre porque al inicio, todas las acciones tienen valores inflados, lo que induce una fase de exploración implícita, donde el agente prueba diferentes acciones hasta identificar la mejor. La subida inicial refleja este proceso de aprendizaje acelerado. Sin embargo, cuando el agente comienza a explotar la mejor acción (subida repentina), los valores de $Q(a)$ se ajustan a las recompensas reales, haciendo que baje su recompensa esperada, y algunas acciones subóptimas aún pueden parecer temporalmente mejores, causando una caída momentánea antes de estabilizarse.

e)

Debido a que se tiene un *step size* fijo $Q(a)$ nunca se estabilizará completamente, sino que estará oscilando permanentemente y nunca llegará a converger en 100%. Luego si hay acciones que tienen valores de recompensa esperados similares, al tener un agente que oscila constantemente, puede ocurrir que de vez en cuando (aprox. el 15% de las veces) termine eligiendo una opción subóptima pensando que era la óptima.

f)

Resultados de repetir el experimento de la Fgira 2.5 del libro con $\mu = 4$:

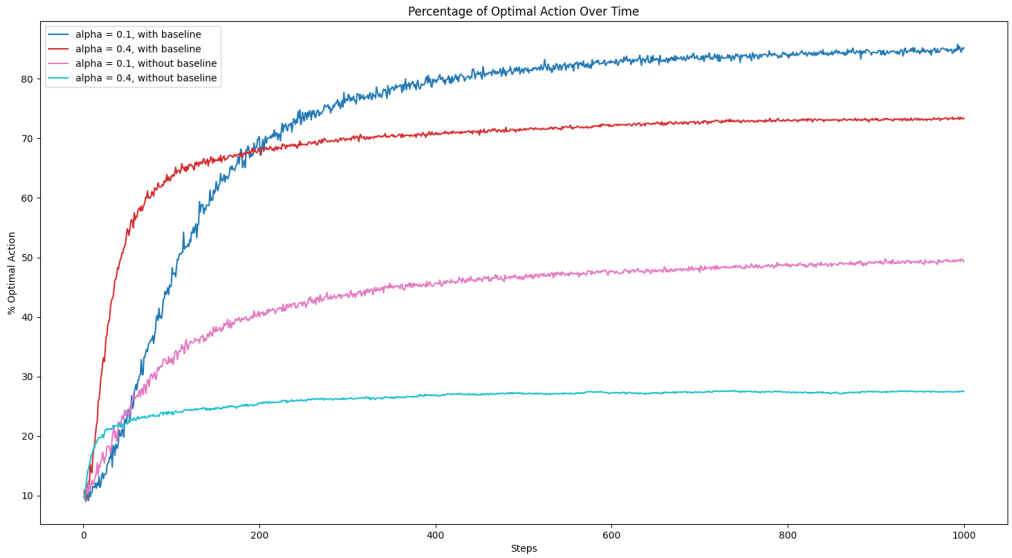

Figure 4: Porcentaje de veces que se elige la acción óptima

En este último experimento también se obtuvieron los resultados esperados, ya que una vez más coinciden con el libro.
Además se observa que un mayor α hace que los agentes aprendan más rápido, sin embargo, no les permite estabilizarse y llegar a porcentajes más altos de precisión en la elección de la acción óptima.

g)

El resultado de realizar el mismo experimento pero con $\mu = 0$ es el siguiente:

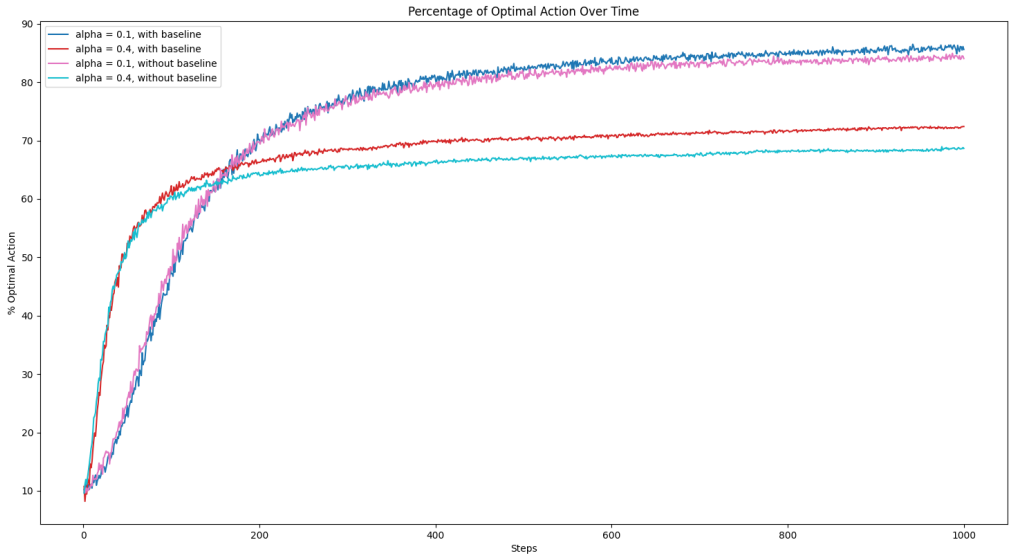

Figure 5: Porcentaje de veces que se elige la acción óptima

Como se puede ver en el gráfico con $\mu = 0$, los resultados con y sin *baseline* son muy similares. Esto sucede porque, al tener una distribución de recompensas centrada en 0, el promedio de recompensas $\bar{R}$ en la ecuación con *baseline* también tiende a 0, haciendo que la actualización de preferencias se parezca a la ecuación sin *baseline*. En otras palabras, cuando $\mu = 0$, el término $(R_t - \bar{R})$ en la actualización de $H(a)$ es prácticamente igual a R_t, lo que hace que la corrección del *baseline* pierda relevancia. Por esta razón, la diferencia entre usar o no *baseline* se reduce considerablemente en este caso.