Aprendizaje por Refuerzo: Introducción

Alvaro J. Riascos Villegas
Universidad de los Andes y Quantil

Agosto 2024

Contenido

1. Introducción
2. Bandido Multibrazo
   - Estrategia ϵ-codiciosa
   - Experimento
   - Implementación
   - Caso no Estacionario
   - Dependencia de valores iniciales
   - Incertidumbre
   - Estudio Compartivo Algoritmos
3. Caso Asociativo
4. Estado del Arte

Características Principales

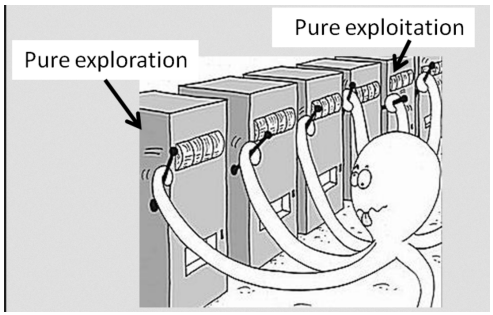
- RL es una teoría que busca descubrir principios simples y generales que permitan caracterizar la inteligencia.
- Idea principal: Aprender de interactuar con un ambiente.
- Vamos a estudiar formas computacionales de hacer esto.
- Las caracteristicas principales son: aprender de experimentar (explotacion y exploracion) y recompensas diferidas (*credit assignment*).
- Cuando se tiene un modelo del ambiente se puede simular (*model based*). Cuando no se tiene un modelo del ambiente se debe interactuar con el (*model free*).
- Los métodos basados en modelos sirven de guía de los métodos sin modelo.

Contenido

1. Introducción
2. Bandido Multibrazo
   - Estrategia ϵ-codiciosa
   - Experimento
   - Implementación
   - Caso no Estacionario
   - Dependencia de valores iniciales
   - Incertidumbre
   - Estudio Compartivo Algoritmos
3. Caso Asociativo
4. Estado del Arte

Caso No Asociativo

- Caso muy particular: El ambiente siempre es el mismo y no cambia con la interacción con el aprendiz (i.e., solo hay un estado del mundo).
- Tenemos K-brazos (acciones):



- **La recompensa $R_t(a)$ de tomar una accion $A_t = a$ en el momento t es una variable aleatoria:** $R_t(a)$. El valor esperado es:

$$q_t(a) = E[R_t(a) \mid A_t = a] \tag{1}$$

Caso No Asociativo

- El objetivo es elegir la mejor acción en cada periodo t.
- No conocemos la distribución de la recompensa $R_t(a)$ para ninguna acción, de lo contrario sería en principio fácil resolver:

$$\max_a q_t(a) \tag{2}$$

Caso No Asociativo

- Sin embargo, si $R_t(a)$ es un proceso estacionario podemos estimar $q_t(a)$ por el promedio de las acciones antes de t:

$$Q_t(a) = \frac{\sum_{i=1}^{t-1} R_i(a) I_{[A_i=a]}}{\sum_{i=1}^{t-1} I_{[A_i=a]}} \tag{3}$$

  y cambiar el problema por:

$$\max_a Q_t(a) \tag{4}$$

- En el caso estacionario escribimos $q_t = q_*$

Estrategia ϵ-codiciosa

- Como $Q_t(a)$ es apenas una estimación del verdadero valor, la estrategia anterior (codiciosa) puede no ser óptima en el largo plazo.
- Si en cada periodo, con probabiliad ϵ exploramos otras acciones, esto puede mejorar la probabilidad de elegir de forma óptima en el largo plazo.
- ϵ es una medida de la incertidumbre que se tiene del estimador $Q_t(a)$.
- La estrategia que elige en cada periodo de forma codiciosa con probabilidad $1 - \epsilon$ y explora con probabilidad ϵ la llamamos ϵ-codiciosa.

Experimento

- La siguiente figura muestra la distribución $R_t(a)$ para diez valores de $q_\star(a)$.

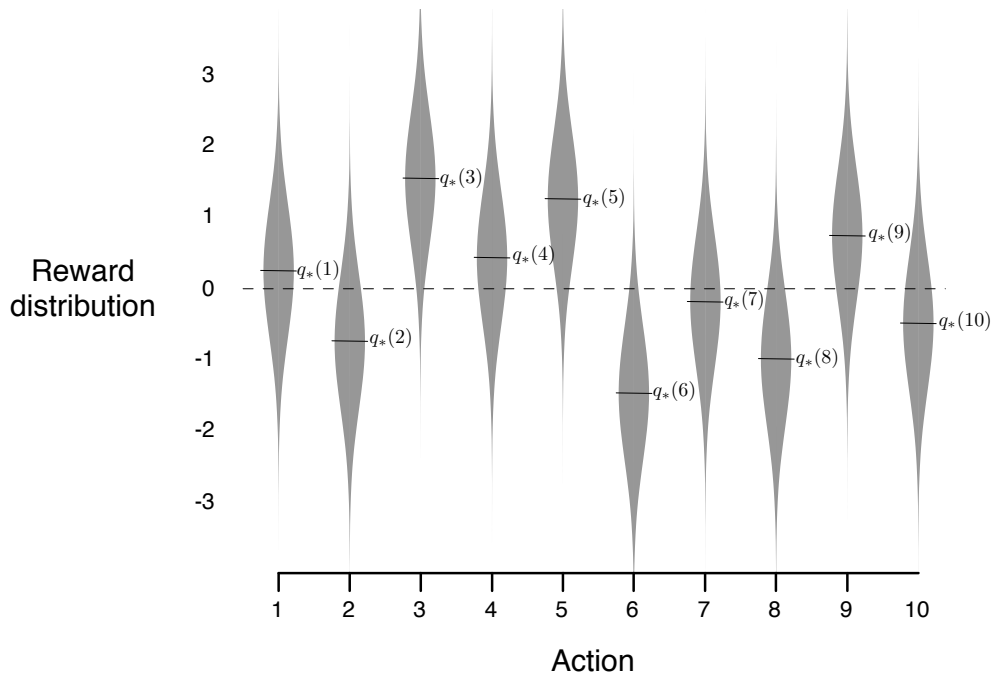


Figure 2.1: An example bandit problem from the 10-armed testbed. The true value $q_*(a)$ of each of the ten actions was selected according to a normal distribution with mean zero and unit variance, and then the actual rewards were selected according to a mean $q_*(a)$ unit variance normal distribution, as suggested by these gray distributions.

Experimento: Resultados

- En la realidad no conocemos las anteriores distribuciones, solo que tenemos diez acciones disponibles (i.e., nos enfrentamos a un bandido de 10 brazos).
- La siguiente figura muestra los resultados de simular 2000 problemas (bandido de 10 brazos): cada simulacion se obtiene como indica el texto de la figura anterior.
- En cada problema de estos se usan estrategias ϵ-codiciosa por 1000 periodos.

Experimento: Resultados

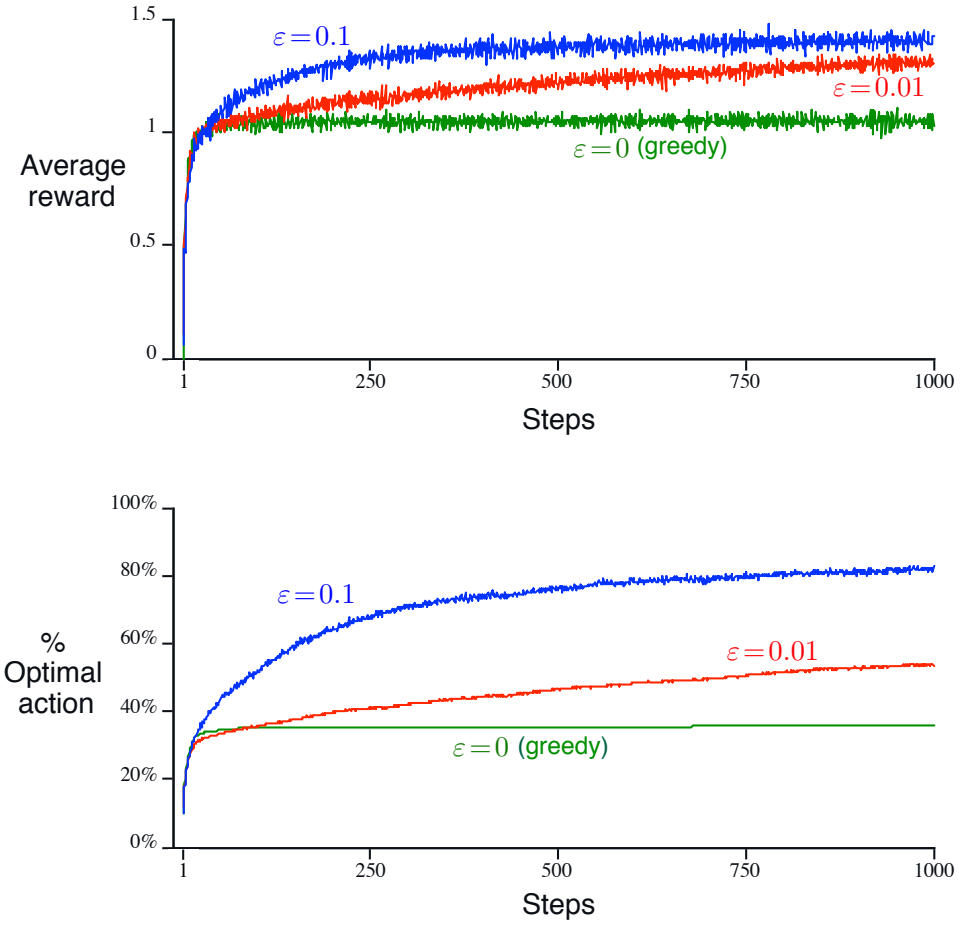


Figure 2.2: Average performance of ε-greedy action-value methods on the 10-armed testbed. These data are averages over 2000 runs with different bandit problems. All methods used sample averages as their action-value estimates.

- La primera gráfica muestra la recompensa promedio y la segunda el porcentaje de veces que cada estrategia seleccionó la estrategia óptima.

Implementación Incremental

- El objetivo es hacer la estrategia anterior computacionalmente eficiente:
  1. En el número de cálculos a realizar.
  2. En memoria.
- Es fácil ver que:

$$Q_{t+1}(a) = Q_t(a) + \frac{I_{[A_t=a]}}{\sum_{i=1}^{t} I_{[A_i=a]}}(R_t(a) - Q_t(a)) \qquad (5)$$

- La forma general de esta estrategia es:

  Nueva estimación $\leftarrow$ Estimación anterior $+$ Tamaño del salto $\times$ (Recompensa $-$ Estimación anterior)

- Obsérvese que entre más iteraciones, menor es el peso que se le da a la actualización.

Estrategia ϵ codiciosa: Pseudocódigo

A simple bandit algorithm

Initialize, for $a = 1$ to k:

$Q(a) \leftarrow 0$

$N(a) \leftarrow 0$

Loop forever:

$$A \leftarrow \begin{cases} \arg\max_a Q(a) & \text{with probability } 1-\varepsilon \quad \text{(breaking ties randomly)} \\ \text{a random action} & \text{with probability } \varepsilon \end{cases}$$

$R \leftarrow bandit(A)$

$N(A) \leftarrow N(A) + 1$

$Q(A) \leftarrow Q(A) + \frac{1}{N(A)}\left[R - Q(A)\right]$

Decaimiento exponencial

- En el algoritmo anterior, entre más iteraciones, menor es el peso que se le da a la actualización.
- Con decaimiento exponencial se utiliza un parámetro $\alpha \in (0, 1)$ y la actualización:

$$Q_{t+1}(a) = Q_t(a) + \alpha(R_t(a) - Q_t(a)) \tag{6}$$

- Esto corresponde a darle más peso a las últimas recompensas:

$$Q_{t+1}(a) = \alpha R_t(a) + (1 - \alpha)Q_t(a) \tag{7}$$

$$= (1 - \alpha)^t Q_1(a) + \alpha \sum_{i=1}^{t} (1 - \alpha)^{t-i} R_i(a) \tag{8}$$

Dependencia de valores iniciales

- Los resultados del algoritmo ϵ-codicioso dependen de la estimación inicial $Q_1(a)$. Este sesgo puede desaparecer solo despues de muchas iteraciones.
- Una forma de fomentar la exploración es iniciar el algoritmo con valores muy optimistas de $Q_1(a)$ (lejos de cero). Por ejemplo si $Q_1(a)$ es muy alto, elegir cualquier a implica que $Q_2(a)$ se actualiza a un menor valor. Luego, en la próxima iteración se elige un nuevo a (aún en modo explotación). De esa forma se eligen todos los a más rápidamente.

Inicios Optimistas

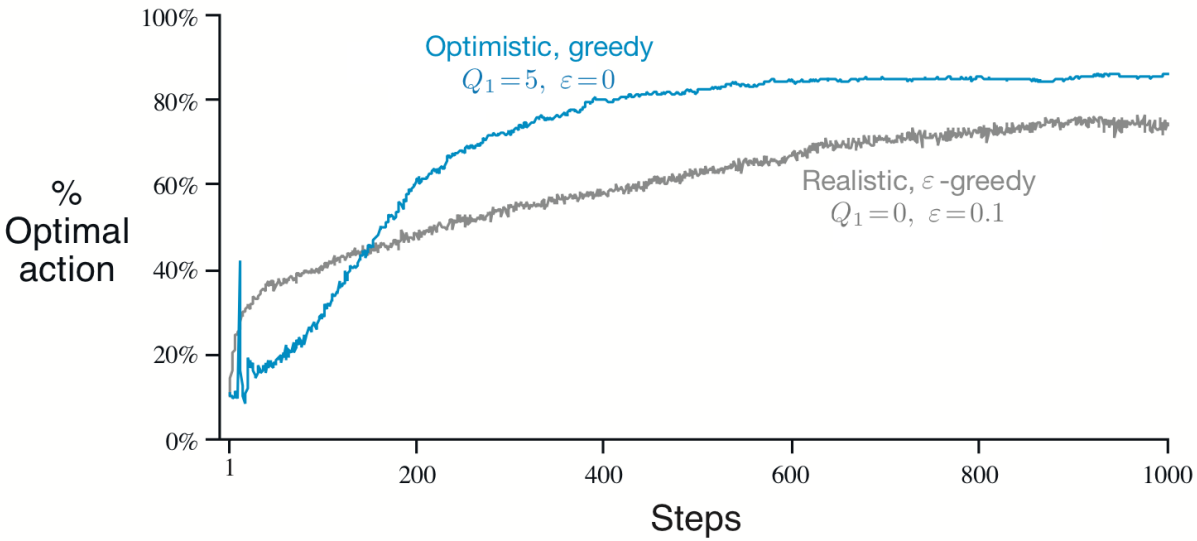


Figure 2.3: The effect of optimistic initial action-value estimates on the 10-armed testbed. Both methods used a constant step-size parameter, $\alpha = 0.1$.

Incertidumbre

- Hemos visto que la estrategia ϵ- codiciosa es una forma de reponder a la incertidumbre de la estrategia codiciosa.
- Dos formas de abordar este problema:
  1. UCB (Upper Confidence Bound).
  2. Una aproximación Bayesiana (i.e., Thompson Sampling).

Incertidumbre

- UCB (Upper Confidence Bound): La estrategia ϵ-codicioso cuando elige de forma aleatoria, lo hace de forma uniforme. Sin embargo, sería mejor si se elige con mayor probabilidad aquellas acciones que tiene un buen balance de rentabilidad e incertidumbre:

$$A_t = argmax_a\{Q_t(a) + c\sqrt{\frac{ln(t)}{N_t(a)}}\} \tag{9}$$

  donde t es el número de veces que se ha disparado alguna arma y $N_t(a)$ es el número de veces que la arma a se ha disparado.

- El segundo término es una medida de la incertidumbre. Entre mas incierta la acción más incentivo a elegirla. La incertidumbre aumenta cuando se ha disparado pocas veces esa arma.

UCB

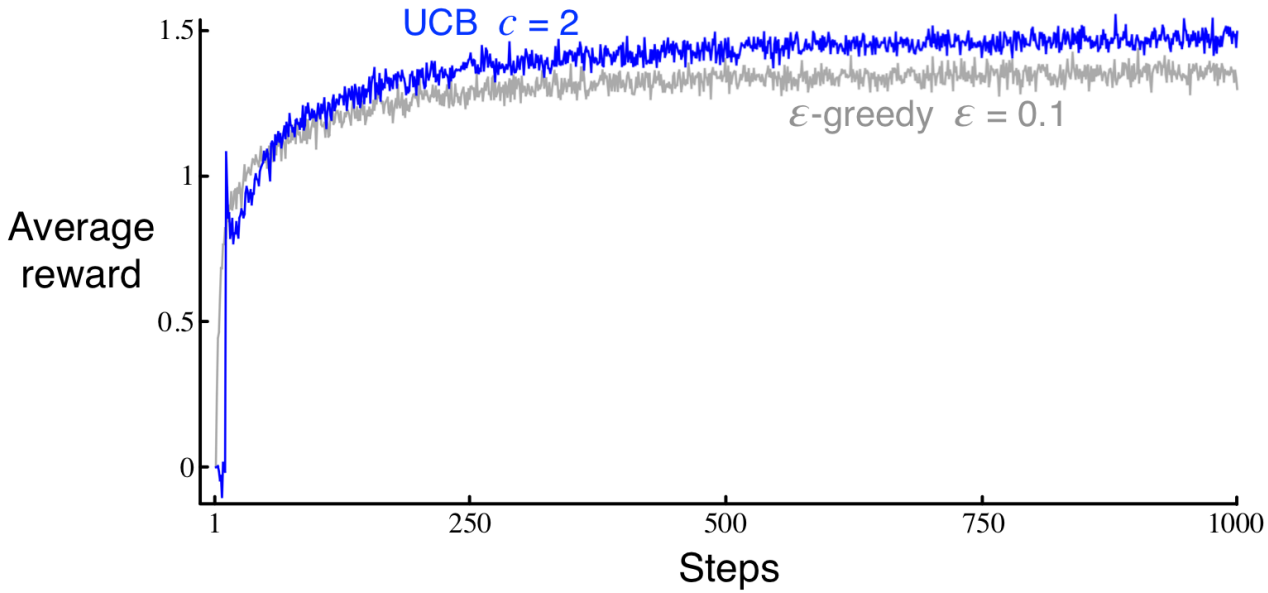


Figure 2.4: Average performance of UCB action selection on the 10-armed testbed. As shown, UCB generally performs better than ε-greedy action selection, except in the first k steps, when it selects randomly among the as-yet-untried actions.

Estudio Compartivo Algoritmos

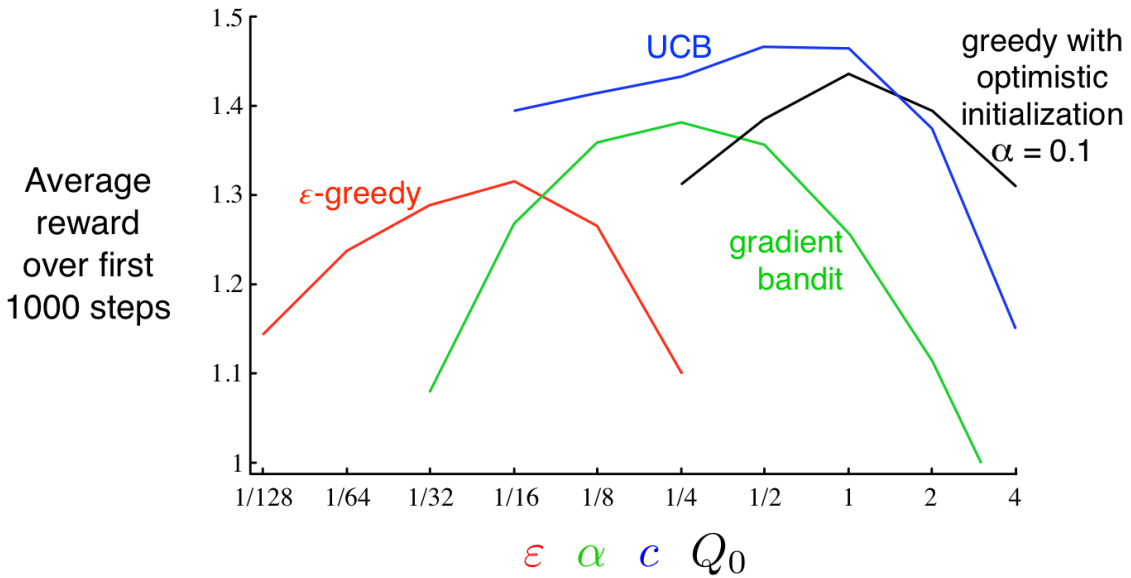


Figure 2.6: A parameter study of the various bandit algorithms presented in this chapter. Each point is the average reward obtained over 1000 steps with a particular algorithm at a particular setting of its parameter.

Contenido

1. Introducción
2. Bandido Multibrazo
   - Estrategia ϵ-codiciosa
   - Experimento
   - Implementación
   - Caso no Estacionario
   - Dependencia de valores iniciales
   - Incertidumbre
   - Estudio Compartivo Algoritmos
3. Caso Asociativo
4. Estado del Arte

Generalización

- Una generalización importante de lo que hemos hecho hasta este punto es introducir un estado S_t (variable aleatoria) que revela información relevante para tomar una acción: **contexto**.
- Un paso más allá es permitir que las acciones modifican el contexto o ambiente: $S_{t+1} = h(S_t, a_t), R_t(s, a)$. Este es el caso **asociativo**, el caso más general que se estudia en aprendizaje por refuerzo.

Contenido

1. Introducción
2. Bandido Multibrazo
   - Estrategia ϵ-codiciosa
   - Experimento
   - Implementación
   - Caso no Estacionario
   - Dependencia de valores iniciales
   - Incertidumbre
   - Estudio Compartivo Algoritmos
3. Caso Asociativo
4. Estado del Arte

AlphaZero

Game	White	Black	Win	Draw	Loss
Chess	*AlphaZero*	*Stockfish*	25	25	0
	Stockfish	*AlphaZero*	3	47	0
Shogi	*AlphaZero*	*Elmo*	43	2	5
	Elmo	*AlphaZero*	47	0	3
Go	*AlphaZero*	*AG0 3-day*	31	–	19
	AG0 3-day	*AlphaZero*	29	–	21

Table 1: Tournament evaluation of *AlphaZero* in chess, shogi, and Go, as games won, drawn or lost from *AlphaZero*'s perspective, in 100 game matches against *Stockfish*, *Elmo*, and the previously published *AlphaGo Zero* after 3 days of training. Each program was given 1 minute of thinking time per move.