Reinforcement Learning in Practice I
The k-Armed Bandit Problem

Antoine SYLVAIN

EPITA

2021

Contents

Greedy algorithms

- Make the locally optimal choice at each stage

Greedy algorithms

- Make the locally optimal choice at each stage
- Full-exploitation

Greedy algorithms

- Make the locally optimal choice at each stage
- Full-exploitation
- High probability of getting stuck in a local optimum

ϵ-Greedy algorithm

- Greedy most of the time

ϵ-Greedy algorithm

- Greedy most of the time
- We have a probability ϵ to explore rather than exploit

ϵ-Greedy algorithm

- Greedy most of the time
- We have a probability ϵ to explore rather than exploit
- With a probability ϵ : select an arm randomly
- With a probability $1 - \epsilon$: select the best known action : $\hat{a}_t^* = argmax_{a \in \mathcal{A}} \mathcal{Q}(a)$

ϵ-Greedy algorithm

- Greedy most of the time
- We have a probability ϵ to explore rather than exploit
- With a probability ϵ : select an arm randomly
- With a probability $1 - \epsilon$: select the best known action : $\hat{a}_t^* = argmax_{a \in \mathcal{A}} \mathcal{Q}(a)$
- $\epsilon = 0 \rightarrow$ greedy algorithm (full exploitation)
- $\epsilon = 1 \rightarrow$ random algorithm (full exploration)

ϵ-Greedy action value

$$Q_t(a) = \frac{1}{N_t(a)} \sum_{\tau=1}^{t} r_\tau$$

with
$N_t(a)$ how many times the action a has been chosen

Contents

Upper Confidence Bounds

- Exploration is a good way to try new options

Upper Confidence Bounds

- Exploration is a good way to try new options
- But it is a pity if your random try select a bad action you already tried

Upper Confidence Bounds

- Exploration is a good way to try new options
- But it is a pity if your random try select a bad action you already tried
- UCB: favor option with high uncertainty, assuming they still have potential

Upper Confidence Bounds

- $\hat{\mathcal{U}}_t(a)$ is the upper confidence bound of the reward value, so that the true value is below with bound with high probability
- $\mathcal{Q}(a) \leq \hat{\mathcal{Q}}_t(a) + \hat{\mathcal{U}}_t(a)$

Upper Confidence Bounds

- $\hat{\mathcal{U}}_t(a)$ is the upper confidence bound of the reward value, so that the true value is below with bound with high probability
- $\mathcal{Q}(a) \leq \hat{\mathcal{Q}}_t(a) + \hat{\mathcal{U}}_t(a)$ with
  $\mathcal{Q}(a)$ the true mean value,
  $\hat{\mathcal{Q}}_t(a)$ the sample mean value

Upper Confidence Bounds

- $\hat{\mathcal{U}}_t(a)$ is the upper confidence bound of the reward value, so that the true value is below with bound with high probability
- $\mathcal{Q}(a) \leq \hat{\mathcal{Q}}_t(a) + \hat{\mathcal{U}}_t(a)$ with
  $\mathcal{Q}(a)$ the true mean value,
  $\hat{\mathcal{Q}}_t(a)$ the sample mean value
- $\hat{\mathcal{U}}_t(a)$ is a function of $N_t(a)$ (number of tries of a)
- the bigger $N_t(a)$, the smaller $\hat{\mathcal{U}}_t(a)$

Upper Confidence Bounds

- $\hat{\mathcal{U}}_t(a)$ is the upper confidence bound of the reward value, so that the true value is below with bound with high probability
- $\mathcal{Q}(a) \leq \hat{\mathcal{Q}}_t(a) + \hat{\mathcal{U}}_t(a)$ with
  $\mathcal{Q}(a)$ the true mean value,
  $\hat{\mathcal{Q}}_t(a)$ the sample mean value
- $\hat{\mathcal{U}}_t(a)$ is a function of $N_t(a)$ (number of tries of a)
- the bigger $N_t(a)$, the smaller $\hat{\mathcal{U}}_t(a)$
- So, how do we calculate $\hat{\mathcal{U}}_t(a)$?

Hoeffding's Inequality

- Let $X_1, ..., X_t$ be independent and identically distributed random variables in $[0, 1]$

Hoeffding's Inequality

- Let $X_1, ..., X_t$ be independent and identically distributed random variables in $[0, 1]$
- For $u > 0$, we have :
  $\mathbb{P}[\mathbb{E}[X] > \bar{X}_t + u] \leq e^{-2tu^2}$
  with $\bar{X}_t = \sum_{\tau=1}^{t} X_t$ the sample mean
  and $u = \mathcal{U}_t(a)$

Hoeffding's Inequality

- Let $X_1, ..., X_t$ be independent and identically distributed random variables in $[0, 1]$
- For $u > 0$, we have :
  $\mathbb{P}[\mathbb{E}[X] > \bar{X}_t + u] \leq e^{-2tu^2}$
  with $\bar{X}_t = \sum_{\tau=1}^{t} X_t$ the sample mean
  and $u = \mathcal{U}_t(a)$
- $\mathbb{P}[\mathcal{Q}(a) > \hat{\mathcal{Q}}_t(a) + \mathcal{U}_t(a)] \leq e^{-2t\mathcal{U}_t(a)^2}$

Hoeffding's Inequality

- Let $X_1, ..., X_t$ be independent and identically distributed random variables in $[0, 1]$
- For $u > 0$, we have :
  $\mathbb{P}[\mathbb{E}[X] > \bar{X}_t + u] \leq e^{-2tu^2}$
  with $\bar{X}_t = \sum_{\tau=1}^{t} X_t$ the sample mean
  and $u = \mathcal{U}_t(a)$
- $\mathbb{P}[\mathcal{Q}(a) > \hat{\mathcal{Q}}_t(a) + \mathcal{U}_t(a)] \leq e^{-2t\mathcal{U}_t(a)^2}$
- $\mathcal{U}_t(a) = \sqrt{\frac{-logp}{2N_t(a)}}$
  with $p = e^{-2t\mathcal{U}_t(a)^2}$

UCB1

- Set $p = t^{-4}$ to reduce the threshold in time
- $\mathcal{U}_t(a) = \sqrt{\frac{2 log t}{N_t(a)}}$

UCB1

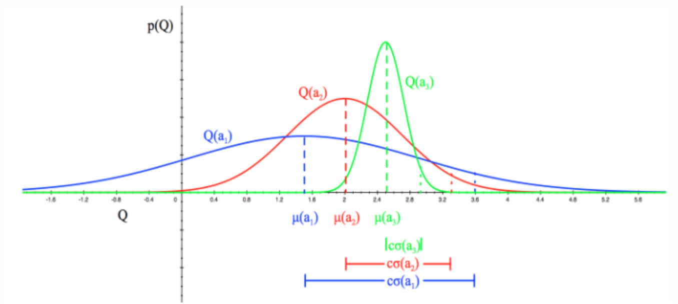
- Set $p = t^{-4}$ to reduce the threshold in time
- $\mathcal{U}_t(a) = \sqrt{\frac{2 log t}{N_t(a)}}$
- $a_t = argmax_{a \in \mathcal{A}} \mathcal{Q}(a) + \mathcal{U}_t(a)$

Bayesian UCB

- Here, we expect the mean reward to be Gaussian

Bayesian UCB

- Here, we expect the mean reward to be Gaussian
- We can set the upper bound by setting $\hat{\mathcal{U}}_t(a)$ to be a multiple of the standard deviation
- For example, with $\hat{\mathcal{U}}_t(a)$ being twice the standard deviation, we set the upper bound as a 95% interval



Bayesian UCB

- At each time step t, we sample an expected reward $\tilde{Q}(a)$

Bayesian UCB

- At each time step t, we sample an expected reward $\tilde{Q}(a)$
- We select the action $a_t = argmax_{a \in \mathcal{A}}(\tilde{Q}(a) + c\sigma(\alpha, \beta))$
- with

  α: success count

  β: fails count

  c: the chosen multiple of the standard deviation

Bayesian UCB

- At each time step t, we sample an expected reward $\tilde{Q}(a)$
- We select the action $a_t = argmax_{a \in \mathcal{A}}(\tilde{Q}(a) + c\sigma(\alpha, \beta))$
- with

  α: success count

  β: fails count

  c: the chosen multiple of the standard deviation
- We update α and β
- $\alpha_i \leftarrow \alpha_i + r_t$
- $\beta_i \leftarrow \beta_i + (1 - r_t)$

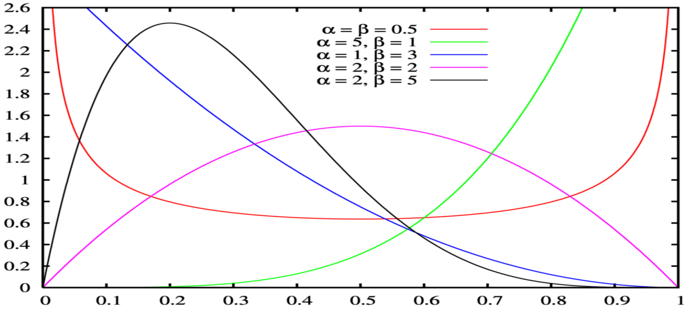
Contents

Thompson Sampling

- At each time step, we want to select action a whose probability is optimal:
- $\pi(a|h_t) = \mathbb{P}[Q(a) > Q(a')\forall a' \neq a|h_t]$
- with $\pi(a|h_t)$ the probability of selecting the action a knowing the history h_t

Thompson Sampling

- We assume that $Q(a)$ follows a beta distribution.
- $Beta(\alpha, \beta) \in [0, 1]$
- α: success count
- β: fails count



Thompson Sampling

- Initialization: $\alpha = \beta = 1$:
- We expect the reward probability to be 50% without much confidence

Thompson Sampling

- Initialization: $\alpha = \beta = 1$:
- We expect the reward probability to be 50% without much confidence
- If $\alpha = 1000$ and $\beta = 9000$, we have a strong confidence that the reward probability is close to 10%.

Thompson Sampling

- At each time step t, we sample an expected reward $\tilde{Q}(a)$ from the prior distribution $Beta(\alpha_i, \beta_i)$ for every action.

Thompson Sampling

- At each time step t, we sample an expected reward $\tilde{Q}(a)$ from the prior distribution $Beta(\alpha_i, \beta_i)$ for every action.
- We select the best action among the sample
- $a_t = argmax_{a \in \mathcal{A}} \tilde{Q}(a)$

Thompson Sampling

- At each time step t, we sample an expected reward $\tilde{Q}(a)$ from the prior distribution $Beta(\alpha_i, \beta_i)$ for every action.
- We select the best action among the sample
- $a_t = argmax_{a \in \mathcal{A}} \tilde{Q}(a)$
- We update the Beta distribution once the true reward has been observed
- $\alpha_i \leftarrow \alpha_i + r_t$
- $\beta_i \leftarrow \beta_i + (1 - r_t)$