Bootstrapping en Apprentissage par Renforcement

Rhouma Haythem

Avril 2025

1 Introduction

En apprentissage par renforcement, le terme *bootstrapping* désigne le fait de **mettre à jour une estimation à partir d'une autre estimation**, plutôt que de s'appuyer uniquement sur des « données complètes » (par exemple, le retour total jusqu'à la fin de l'épisode).

Concrètement, au lieu d'attendre la fin de l'épisode pour calculer la somme de toutes les récompenses futures, on utilise une *cible* qui contient déjà une valeur estimée, comme $V(S_{t+1})$ ou $\max_a Q(S_{t+1}, a)$.

Cette idée est au coeur des méthodes TD (Temporal Difference) et de Q-Learning.

2 Retour Monte Carlo : pas de bootstrapping

Les méthodes Monte Carlo mettent à jour la valeur d'un état en utilisant le *retour complet* observé à partir de cet état jusqu'à la fin de l'épisode. Pour un état S_t, on définit le retour G_t comme :

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots + \gamma^{T-t-1} R_T, \tag{1}$$

où T est le temps de fin de l'épisode.

La mise à jour Monte Carlo est alors :

$$V(S_t) \leftarrow V(S_t) + \alpha\big(G_t - V(S_t)\big), \tag{2}$$

ou, en forme pondérée :

$$V(S_t) \leftarrow (1-\alpha)V(S_t) + \alpha G_t. \tag{3}$$

Ici, la **cible** G_t est entièrement calculée à partir des **récompenses réelles**. Il n'y a aucune valeur estimée à l'intérieur de G_t. Par conséquent, il **n'y a pas de bootstrapping** dans les méthodes Monte Carlo.

3 Intuition du bootstrapping

Dans les méthodes TD et Q-Learning, on n'attend pas la fin de l'épisode. On utilise une cible de la forme :

$$\text{cible} = R_{t+1} + \gamma V(S_{t+1}), \tag{4}$$

ou encore :

$$\text{cible} = R_{t+1} + \gamma \max_a Q(S_{t+1}, a). \tag{5}$$

Dans ces deux cas, la cible contient :

— une partie *observée*, la récompense immédiate R_{t+1},
— une partie *estimée*, la valeur $V(S_{t+1})$ ou $Q(S_{t+1}, a)$.

On **corrige** donc $V(S_t)$ ou $Q(S_t, A_t)$ en utilisant une cible qui n'est pas totalement vraie, mais qui repose déjà sur nos propres estimations. C'est cela, le **bootstrapping** :

nouvelle estimation ← ancienne estimation corrigée à partir d'une autre estimation.

4 Bootstrapping dans TD(0) et TD(1)

4.1 TD(1)

La mise à jour TD(1) standard est :

$$V(S_t) \leftarrow V(S_t) + \alpha\big(R_{t+1} + \gamma V(S_{t+1}) - V(S_t)\big). \tag{6}$$

En forme pondérée :

$$V(S_t) \leftarrow (1-\alpha)V(S_t) + \alpha\big(R_{t+1} + \gamma V(S_{t+1})\big). \tag{7}$$

Nous voyons que la cible est :

$$\text{cible TD(1)} = R_{t+1} + \gamma V(S_{t+1}),$$

où $V(S_{t+1})$ est une **estimation**. On met donc à jour $V(S_t)$ en utilisant une autre estimation : c'est du bootstrapping.

4.2 TD(0) comme cas extrême

On peut définir un TD(0) (ici dans la famille n-step) comme :

$$V(S_t) \leftarrow V(S_t) + \alpha\big(\gamma V(S_{t+1}) - V(S_t)\big), \tag{8}$$

ou

$$V(S_t) \leftarrow (1-\alpha)V(S_t) + \alpha\gamma V(S_{t+1}). \tag{9}$$

La cible est alors simplement :

$$\text{cible TD(0)} = \gamma V(S_{t+1}),$$

c'est-à-dire uniquement du bootstrapping sur $V(S_{t+1})$, sans utiliser R_{t+1}.

5 Bootstrapping dans TD(n)

La généralisation TD(n) introduit le retour n-step :

$$G_t^{(n)} = \sum_{k=1}^{n} \gamma^{k-1} R_{t+k} + \gamma^n V(S_{t+n}). \tag{10}$$

On met alors à jour $V(S_t)$ par :

$$V(S_t) \leftarrow V(S_t) + \alpha\big(G_t^{(n)} - V(S_t)\big), \tag{11}$$

ou, en forme pondérée :

$$V(S_t) \leftarrow (1-\alpha)V(S_t) + \alpha\Big(\sum_{k=1}^{n} \gamma^{k-1} R_{t+k} + \gamma^n V(S_{t+n})\Big). \tag{12}$$

Remarque importante :

— La partie $\sum_{k=1}^{n} \gamma^{k-1} R_{t+k}$ contient uniquement des récompenses réelles.
— La partie $\gamma^n V(S_{t+n})$ est du **bootstrapping** car $V(S_{t+n})$ est une estimation.

Plus n est grand, plus on se rapproche d'un retour de type Monte Carlo, mais il y a toujours du bootstrapping tant qu'on garde un terme $V(S_{t+n})$ dans la cible.

6 Bootstrapping dans Q-Learning et SARSA

6.1 Q-Learning

La mise à jour de Q-Learning est :

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\big[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)\big]. \tag{13}$$

En forme pondérée :

$$Q(S_t, A_t) \leftarrow (1-\alpha) Q(S_t, A_t) + \alpha\big[R_{t+1} + \gamma \max_a Q(S_{t+1}, a)\big]. \tag{14}$$

La cible contient le terme :

$$\gamma \max_a Q(S_{t+1}, a),$$

qui est une **estimation** de la meilleure valeur future possible. On corrige donc $Q(S_t, A_t)$ à partir de cette estimation, ce qui constitue du bootstrapping.

6.2 SARSA

De même, pour SARSA, la mise à jour est :

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\big[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)\big], \tag{15}$$

où cette fois, la cible utilise $Q(S_{t+1}, A_{t+1})$ (action réellement choisie sous la politique en cours). Là encore, ce terme est une estimation, donc il y a bootstrapping.

7 Comparaison : Monte Carlo vs TD (avec bootstrapping)

7.1 Monte Carlo

— Utilise un retour complet G_t basé uniquement sur des récompenses observées.
— Pas de bootstrapping : la cible ne contient aucune valeur estimée.
— Nécessite d'attendre la fin de l'épisode.
— Variance élevée, mais biais faible.

7.2 TD / Q-Learning

— Utilisent des cibles qui mélangent récompenses immédiates et valeurs estimées.
— Bootstrapping : on corrige une estimation par une autre estimation.
— Mise à jour possible *avant* la fin de l'épisode.
— Variance plus faible, mais introduit un biais supplémentaire.

8 Petit exemple numérique de bootstrapping

Considérons un état S_t avec :

$$V(S_t) = 5.0, \quad V(S_{t+1}) = 7.0, \quad R_{t+1} = 2.0,$$

et prenons :

$$\gamma = 0.9, \quad \alpha = 0.1.$$

8.1 Mise à jour Monte Carlo (hypothétique)

Supposons que le retour complet observé soit :

$$G_t = 15.0.$$

Alors :

$$V(S_t) \leftarrow V(S_t) + \alpha(G_t - V(S_t)) = 5.0 + 0.1(15.0 - 5.0) = 5.0 + 1.0 = 6.0. \tag{16}$$

Ici, la cible est $G_t = 15.0$, construite uniquement à partir de récompenses ; aucune valeur estimée n'apparaît dans la cible.

8.2 Mise à jour TD(1) avec bootstrapping

La cible TD(1) est :

$$\text{cible TD(1)} = R_{t+1} + \gamma V(S_{t+1}) = 2.0 + 0.9 \times 7.0 = 2.0 + 6.3 = 8.3.$$

La mise à jour est :

$$V(S_t) \leftarrow V(S_t) + \alpha(\text{cible} - V(S_t)) = 5.0 + 0.1(8.3 - 5.0) = 5.0 + 0.1 \times 3.3 = 5.0 + 0.33 = 5.33. \tag{17}$$

Ou, en forme pondérée :

$$V(S_t) \leftarrow (1 - \alpha)V(S_t) + \alpha \times 8.3 = 0.9 \times 5.0 + 0.1 \times 8.3 = 4.5 + 0.83 = 5.33. \tag{18}$$

On voit que :
— La cible Monte Carlo (15.0) est basée uniquement sur des récompenses réelles.
— La cible TD(1) (8.3) utilise la valeur estimée $V(S_{t+1}) = 7.0$: c'est précisément le **bootstrapping**.

9 Conclusion

Le *bootstrapping* est l'un des concepts centraux des méthodes TD et Q-Learning :
— On met à jour V ou Q en utilisant des cibles qui contiennent déjà des estimations.
— Cela permet des mises à jour plus rapides, avant la fin des épisodes.
— En contrepartie, on introduit un biais, mais on réduit souvent la variance.

Comprendre clairement la différence entre *retour complet* (Monte Carlo) et *cible bootstrappée* (TD, Q-Learning) est fondamental pour analyser les comportements, la stabilité et la convergence des algorithmes d'apprentissage par renforcement.