

# Learning Tetris Using the Noisy Cross-Entropy Method

István Szita<sup>1</sup>, András Lőrincz<sup>2</sup>

*Department of Information Systems, Eötvös Loránd University  
Pázmány Péter sétány 1/C, Budapest, Hungary H-1117*

## 【强化学习算法 7】CEM



张楚珩

清华大学 交叉信息院博士在读

7 人赞同了该文章

CEM 指的是 cross entropy method，本来是一类优化方法，但是大家引用的时候通常指的是这篇文章提到的算法。

原文传送门：

Szita, István, and András Lőrincz. "Learning Tetris using the noisy cross-entropy method." *Neural computation* 18.12 (2006): 2936-2941.

**特色：**把policy到performance的映射看做一个黑盒子，用无导数的方法和线性的策略来解。这个2006年发表的方法看起来粗糙，但在玩俄罗斯方块这个游戏上，多年来打败了无数高级的算法。在Mujoco的很多任务上，这个算法依然能有很不错的表现，可以说很惊艳了。

**分类：**Model-free、**Derivative-free**、On-policy（不太清楚无导数方法还分不分这个）、Continuous State Space、Continuous Action Space、Not Support High-dim Input（因为是线性策略）、Deterministic Policy

**过程：**

1. 用一个linear deterministic policy来解MDP控制问题，认为  $\pi_M(s) = M\phi(s)$ ；目标是最大化  $S(M) = \mathbb{E}_{\pi_M}[\sum_t r_t]$ ；把要解的问题看做一个黑盒子  $M \rightarrow S(M)$

2. Cross-entropy方法主要想法就是维护一个可能最优解的分布，然后根据采样并且查询黑盒子的采样值，更新该分布。把  $w = \text{vec}(M)$  看做一个向量，每轮都采n个样本  $w_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$   $i \in [n]$ ，然后进行rollout得到相应的值  $S(w_1), S(w_2), \dots, S(w_n)$ ，取值最大的前  $|I|$  个，得到它们index的集合  $I \subseteq [n]$ ，然后对分布进行更新

$$\mu_{t+1} := \frac{\sum_{i \in I} w_i}{|I|},$$

$$\sigma_{t+1}^2 := \frac{\sum_{i \in I} (w_i - \mu_{t+1})^T (w_i - \mu_{t+1})}{|I|} + Z_{t+1},$$

后面加入噪声  $z_t$  是为了防止方差过快地收敛。

编辑于 2018-10-01

算法（书籍）

算法

强化学习 (Reinforcement Learning)

▲ 赞同 7



● 添加评论

🔗 分享

♥ 喜欢

★ 收藏



文章被以下专栏收录



强化学习前沿  
读呀读paper

进入专栏