남이 써놓은 자료를 계속 보자니, 왜 이소리가 여기서 나오는건지 이해가 잘 안 되는 부분이 있는 것 같다. 영어 논문을 복사해서 처음부터 끝까지 해석해보자.

제목. **Policy Gradient Methods** for Reinforcement Learning with **Function Approximation**

function approximation은 실제 함수를 여러가지 샘플을 뽑아서 근사값으로 가정한다는 말일 것이다. 그렇다면 어떤 함수를 approximate한다는 것일까? state-action value function? state-value function?

Abstract

Function approximation is essential to reinforcement learning, but the standard approach of approximating a value function and determining a policy from it has so far proven theoretically intractable.

Fuctional approximation은 강화학습에 있어서 필수적이지만, 일반적인 접근법으로 value-function을 근사하고 이를 바탕으로 policy를 결정하는 것은 이론적으로 난해한 것으로 지금까지 증명되었다.

In this paper we explore an alternative approach in which the policy is explicitly represented by its own function approximator, independent of the value function, and is updated according to the gradient of expected reward with respect to the policy parameters.

이 논문에서 우리는 새로운 어프로치를 발견했다 – policy가 명시적으로 그 자신의 function approximator로 표현되어 value function과 무관하고, 그리고 정책 파라미터에 관한 기대 reward의 그래디언트에 의해 업데이트 되는.

Williams's REINFORCE method and actor-critic methods are examples of this approach. Our main new result is to show that the gradient can be written in a form suitable for estimation from experience aided by an approximate action-value or advantage function.

윌리엄의 REINFORCE 방식과 actor-critic 방식이 이 접근법의 예가 될 수 있다. 우리의 주된 새로운 결론은 그래디언트가 approximate action-value 또는 advantage function에 의해 만들어진 experienc로부터 추측되는데 알맞은 형태로 쓰여질 수 있음을 보여주기 위함이다.

윌리엄의 Reinforce 방식?

앞서 유도했었던 이 식, monte carlo gradient estimation을 나타내는 이 식을 이용하는 방식이다. f(x)의 기대값을 하나의 유려한 함수로 나타내기는 엄청 힘든 법이다. 위를 이용하면, f(x)를 몰라도 확률분포 그래디언트와 f(x)값을 곱한 값을 다수 샘플링해 평균값을 구하면 전체 그래디언트를 얻을 수 있다는 이야기 인 것 같다. 그럼 pg는 Reinforcement 방식과 뭐가 다르다는 걸까?

Using this result, we prove for the first time that a version of policy iteration with arbitrary differentiable function approximation is convergent to a locally optimal policy.

이 결과를 이용해, 우리는 임의의 미분가능한 function approximation에 의한 policy iteration의 한 버전이 로컬 옵티멀로 수렴함을 증명했다.

본문

Large applications of reinforcement learning (RL) require the use of generalizing function approximators such neural networks, decision-trees, or instance-based methods. The dominant approach for the last decade has been the value-function approach, in which all function approximation effort goes into estimating a value function, with the action-selection policy represented implicitly as the "greedy" policy with respect to the estimated values (e.g., as the policy that selects in each state the action with highest estimated value).

대규모 RL 응용은 nn이나, decision tree나 인스턴스 기반 방법 같은 일반화된?하는? function approximators의 사용이 필요하다. 지난 십년간 가장 주된 접근법은 모든 fuction approximation 을 위한 노력이 value function을 추정하는데 투자되는 value-fuction 접근법이고, 추정된 value에 대한 greedy policy로 action-selection policy가 암묵적으로 표현되었다.(예: 각 상태에서 가장 높은 추정가치를 가진 action을 선택하는 policy)

The value-function approach has worked well in many applications, but has several limitations.

이것은 많은 응용에서 잘 동작해왔지만 몇가지 한계가 있다.

First, it is oriented toward finding deterministic policies, whereas the optimal policy is often stochastic, selecting different actions with specific probabilities (e.g., see Singh, Jaakkola, and Jordan, 1994).

첫째로, 이것은 deterministic policy를 찾는 것을 지향하는 반면에 몇몇 optimal policy는 stochastic하여 특정 확률로 action을 선택한다. (예로 Singh, Jaakkola, and Jordan논문을 보아라)

Second, an arbitrarily small change in the estimated value of an action can cause it to be, or not be, selected. Such discontinuous changes have been identified as a key obstacle to establishing convergence assurances for algorithms following the value-function approach (Bertsekas and Tsitsiklis, 1996).

둘째로, 추정된 action value값에 약간의 변화만 생겨도 선택이 되느냐 마느냐를 결정하게 된다. 이런 비연속적인 변화가 value-function 접근법을 따르는 알고리즘의 수렴을 보장하는 데에 핵심 장애물이라고 알려져있다. (Bertsekas and Tsitsiklis, 1996).

For example, Q-Iearning, Sarsa, and dynamic programming methods have all been shown unable to converge to any policy for simple MDPs and simple function approximators (Gordon, 1995, 1996; Baird, 1995; Tsitsiklis and van Roy, 1996; Bertsekas and Tsitsiklis, 1996).

Qlearning 외 등등의 방법들이 간단한 MDP나 function apporximator에 수렴하지 못하는 것으로 보여졌다. (Gordon, 1995, 1996; Baird, 1995; Tsitsiklis and van Roy, 1996; Bertsekas and Tsitsiklis, 1996). \* MDP : Markov Decision Procss

This can occur even if the best approximation is found at each step before changing the policy, and whether the notion of "best" is in the mean-squared-error sense or the slightly different senses of residual-gradient, temporal-difference, and dynamic-programming methods.

이것은 policy를 바꾸기 전 매 스텝마다 최선의 approximation이 발견되어도, 그리고 이 최선이 mse 관점인지 residual gradient, temporal-difference, dynamic-programming method의 살짝 다른 관점인지 여부와 관계없이 발생할 수 있다.

Dynamic programming : 주어진 문제를 여러 개의 하위 문제로 나누어 푼 다음 결합해 최종 문제를 해결하는 방식. 벨먼-포드, 데익스트라, 플로이드-워셜 알고리즘. 등등…

In this paper we explore an alternative approach to function approximation in RL.

새방법 발견했다.

Rather than approximating a value function and using that to compute a deterministic policy, we approximate a stochastic policy directly using an independent function approximator with its own parameters.

deterministic policy를 계산하기 위해 value function을 근사하기보다, 우리는 스스로의 패러미터를 가진 독립적인 function approximator를 이용해 stochastic policy를 근사한다.

For example, the policy might be represented by a neural network whose input is a representation of the state, whose output is action selection probabilities, and whose weights are the policy parameters.

예를들어 policy는 입력으로 state 표현, 출력으로 action 선택확률, 그리고 weight으로 policy parameter을 가진 nn으로 표현될 수도 있다.

Let \theta denote the vector of policy parameters and \rho the performance of the corresponding policy (e.g., the average reward per step). Then, in the policy gradient approach, the policy parameters are updated approximately proportional to the gradient:

그래디언트에 비례해서 policy는 업데이트된다.

where is a positive-definite step size. If the above can be achieved, then can

usually be assured to converge to a locally optimal policy in the performance measure

. Unlike the value-function approach, here small changes in can cause only small

changes in the policy and in the state-visitation distribution.

…는 보통 로컬 옵티멀 policy로 수렴이 보장된다. value function 접근법과 다르게 theta 변화는 policy와 state 방문 분포에 작은 변화만을 줄 수 있다.

In this paper we prove that an unbiased estimate of the gradient (1) can be obtained from experience using an approximate value function satisfying certain properties. Williams's (1988, 1992) REINFORCE algorithm also finds an unbiased estimate of the gradient, but without the assistance of a learned value function. REINFORCE learns much more slowly than RL methods using value functions and has received relatively little attention. Learning a value function and using it to reduce the variance of the gradient estimate appears to be ess~ntial for rapid learning. Jaakkola, Singh and Jordan (1995) proved a result very similar to ours for the special case of function approximation corresponding to tabular POMDPs. Our result strengthens theirs and generalizes it to arbitrary differentiable function approximators. Konda and Tsitsiklis (in prep.) independently developed a very simialr result to ours. See also Baxter and Bartlett (in prep.) and Marbach and Tsitsiklis (1998).