swissQuant
Mastering Complexity

Mathematical Challenge August 2019

Policy optimization in approximate dynamic programming

References

[1] Francis A Longstaff and Eduardo S Schwartz. Valuing american options by simulation: a simple least-squares approach. *The review of financial studies*, 14(1):113–147, 2001.

[2] Jonas Mockus. *Bayesian approach to global optimization: theory and applications*, volume 37. Springer Science & Business Media, 2012.

[3] Warren B Powell. *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, volume 842. John Wiley & Sons, 2011.

[4] James C Spall. *Introduction to stochastic search and optimization: estimation, simulation, and control*, volume 65. John Wiley & Sons, 2005.

Description

Motivation

Approximate dynamic programming [3] aims at solving real-life dynamic optimization problems by approximating the dynamic programming approach. This type of problems are essentially ubiquitous, since they arise in all situations where an agent interacts with a system over time with the aim of optimizing its actions a w.r.t a given value/cost function C in a given state S

$$a^* = argmax_a \mathbf{E}\left[\sum_t C_t(S_t, a_t(S_t))|a\right] \tag{1}$$

a.o. when optimizing truckload in logistics, production and storage of energy, inventory management and the execution of large portfolio transactions.

While DP provides a theoretical approach to solve these problems by iteratively solving backwards the Bellman's equation:

$$a_t^*(S_t) = argmax_{a_t}\left(C_t(S_t,a_t) + \gamma \mathbf{E}\left[V_{t+1}(S_{t+1})|S_t,a_t\right]\right) \quad (2)$$

$$V_t(S_t) = C_t(S_t,a_t^*) + \gamma \mathbf{E}\left[V_{t+i}(S_{t+1})|S_t,a_t^*\right] \quad (3)$$

in practice the approach suffers from the so called "course of dimensionality" problem. I.e. the required computation explodes with increasing state and action dimensionality.

Technical Details

One common approach to tackle this issue is to iteratively approximate the value function V and the optimal policy a^*

$$a_t^{*,(n)}(S_t) = argmax_{a_t}\left(C_t(S_t,a_t) + \gamma \mathbf{E}\left[V_{t+1}^{(n)}(S_{t+1})|S_t,a_t\right]\right) \quad (4)$$

$$V_t^{(n+1)}(S_t) = C_t(S_t,a_t^{*,(n)}) + \gamma \mathbf{E}\left[V_{t+i}^{(n)}(S_{t+1})|S_t,a_t^*\right] \quad (5)$$

by sampling paths and proceeding forwards along those. In this challenge we will however focus on approaches not relying on the formulation of a value function but rather leveraging prior information about "good" policies, which are usually available either in the form of a reference policy or a family of policy candidates.
In the case a reference policy $a_t^{(r)}(S_t)$ is available, an approximation of the future values of being in a state S_t can be obtained as:

$$V_t^r(S_t) = \mathbf{E}\left[\sum_{t'>t} C(S_{t'},a_{t'}^{(r)}(S_{t'}))|S_t,a^{(r)}\right] \quad (6)$$

This approximation can be used once a given state or time horizon is reached. The optimal actions are then obtained for example solving the original problem on a shorter (rolled) horizon:

$$a_{t:t+T}^*(S_t) = argmax_{a_{t:t+T}}\mathbf{E}\left[\sum_{t'>t}^{t+T} C(S_{t'},a_{t'}) + V_{t+T}^{(r)}(S_{t+T})|a_{t:t'},S_t\right] \quad (7)$$

The obtained algorithms are usually fast but their performance are strongly application dependent.
Another class of approaches instead exploit prior knowledge about the optimal policy assuming a parametric from $a(S_t,\theta)$ and solving

$$\theta^* = argmax_\theta \mathbf{E}\left[F(\theta)\right] = argmax_\theta \mathbf{E}\left[\sum_t C_t(S_t,a(S_t,\theta))|a(\cdot,\theta)\right] \quad (8)$$

The problem (8) can be solved using stochastic search techniques [4]. In particular, when computing the function F is expensive, as it is often the case when F represents the value of a policy as in (8), the trade-off between searching for an optimal solution and limiting the computational effort has to be considered. For this purpose it can be beneficial to follow

a Bayesian optimization approach [2]. In general, this consists in considering samples of $F(\theta)$ as noisy observations of the function $f(\theta) = \mathbf{E}[F(\theta)]$, which can be used to update a prior belief. The posterior distribution can in turn be used to determine the θ which should be evaluated next. A common policy to chose the next information to gather consists in iteratively evaluating the parameter with the upper confidence bound (UCB).

$F(\theta) = f(\theta) + \epsilon$
Given the initial priori $p^{(0)}(f)$
for $n = 0, \cdots, N$ **do**
  $\theta_n = argmax_\theta \; q_\alpha \left(p^{(n)}(f(\theta))\right)$
  Evaluate a sample $F(\theta_n)$
  Update

$$p^{(n+1)}(f) = p(F(\theta_n)|f)p^{(n)}(f)$$

end for
$\theta^* = argmax_\theta \, \mathbf{E}_{p^{(N)}}[f(\theta)]$

Questions

- **Q1:** *Consider an American style derivative, compare the policy you would obtain using a least-square MC approach [1] with the one optimizing a policy based on a parametrisation $P(\tau; \theta)$ of the boundary of the exercise region*
- **Q2:** *The UCB rule does not seem to be fully aligned with the actual goal of choosing information in order to improve the current best estimate performance $maxf(\theta)$. A possible alternative consists in using a knowledge gradient approach [3]. Compare the two approaches in one case where the parameter $\theta \in \mathbf{R}$ and f is assumed to follow a Gaussian process distribution*

We look forward to your opinions and insights.

Best Quant Regards,

swissQuant Group Leadership Team