My problem

甄景贤 (King-Yin Yan)

General.Intelligence@Gmail.com

January 20, 2018

The problem I want to solve is to combine these 2 paradigms:

- reinforcement learning (also known as dynamic programming)
- logical reasoning

Part 1

The first part is pretty standard: It consists of a dynamical system:

$$x_{t+1} = F(x_t, u_t) \tag{1}$$

where u is the **control**. The problem is to seek an optimal trajectory of x_t. In artificial intelligence, x is the **mental state** of an intelligent agent, F is its **knowledge-base**. The reward at state x is given by $L(x)$ where L corresponds to the **Lagrangian** in the Hamiltonian formulation. In other words we are trying to maximize the **action** which is the time integral of the Lagrangian:

$$\boxed{\text{action}} \quad A = \int L(x(t))dt \tag{2}$$

The optimality condition is given by the **Hamilton-Jacobi equation**, its discrete version is called the **Bellman equation**. We are not just trying to find the optimal trajectory x_t, but we are also *learning* F itself, because F represents the knowledge of the agent and is modifiable.

We use a (deep) neural network to represent F, that is to say:

weight matrix for each layer　　total # of layers

$$x_{t+1} = F(x) = \sigma(W_1 \sigma(W_2 ... \sigma(W_L \; x))) \tag{3}$$

So our dynamical system is a deep neural network joined from end to end to form a loop, also called a **recurrent** neural network. The dynamical state x changes from each iteration (ie, 1 pass) of the neural network.

So far, all this is pretty standard. It belongs to the currently very hot research topic of "deep reinforcement learning".

Part 2

在逻辑中，$\Psi_1 \vdash \Psi_2$ 代表 **逻辑推导**，其中 Ψ_1 是**前提**，Ψ_2 是**结论**。对应於 Boolean lattice 可以记作：

$$\boxed{\text{logic}} \quad \Psi_1 \vdash \Psi_2 \quad \Leftrightarrow \quad \Psi_1 \leq \Psi_2 \quad \boxed{\text{lattice}} \tag{4}$$

（注意方向相反，是惯例）

我的目的是：令神经网络 F 做 $\vdash$ 的工作，换句话说 F approximates $\vdash$。

F 的作用是在 Boolean lattice 中**向下移一步**，对应於逻辑中的**一步推论** (single-step deduction)。

我的问题是：既然 F 有这 lattice monotone automorphism 的结构，那么 F 作为一个 neural network，它的 learning algorithm 应该可以**加快**。换句话说，可以**交替**使用 Bellman update (for Part 1) 和 "lattice update" (for Part 2)，similar to the **Method of Alternating Projections** of convex sets. 因为我们需要同时符合 Parts 1 and 2 的两个条件。在机器学习的术语中，我们的 search space 缩小了 (dimensionality reduction)，因为原本的 search space 是 F 的 function space，新的 search space 是 quotient 了 Boolean lattice 的结构。

神经网络 $F: V \to V$ 作用在 vector space 上，但 $\vdash: L \to L$ 作用在 Boolean lattice 上，所以需要将 Boolean lattice represent 到 vector space 上，亦即 $\rho: L \to V$。

Satisfying the following commutative diagram:

- L 代表 Boolean lattice, with order relation $\geq$.
- $f: L \to L$ 是 monotonous automorphism，即 $f(a) \geq f(b)$ if $a \geq b$.
- V 是 vector space，ρ is a representation $\rho: L \to V$.
- F 是在 ρ 之下保持 $\geq$ 关系的映射，亦同时是上一节的 neural network function。

$$\begin{array}{ccc} L & \xrightarrow{f} & L \\ {\scriptstyle\rho}\downarrow & & \downarrow{\scriptstyle\rho} \\ V & \xrightarrow{F} & V \end{array} \tag{5}$$

但我暂时不清楚 ρ 的做法，和 F 如何 construct。