Search spaces and gradient descent

We want to find the optimal policy given some MDP. But how should we search for this policy? We could search within set of potentially optimal policies, the $|A|^{|S|}$ discrete policies, or we could search within the set of possible value functions, $\mathbb{R}^{|S|}$, or maybe some other space. Which space allows us to find the optimal policy in the cheapest manner?

Phrased differently, this is a question about how the space we are searching within effects the search dynamics: its rate of convergence and the possible trajectories.

Naively, we know that smaller search spaces are better. We would rather search for our keys in a single room, rather than many. But added structure (for example, continuity) can be exploited to yield faster search, even when there are infinitely more states to search.

Value iteration

In RL it is possible to transform the hard combinatorial problem of searching through the $|A|^{|S|}$ possible discrete policies, into an easier (how do we know it is easier?!? PROOF) problem, a search through all possible policies ?!?.

Policy iteration

When transforming between two spaces, how does the optimisation space change? Does my abstraction make optimisation easier?

Model iteration

Search through possible models, τ, r, calculate the optimal policy $\pi^*_{\tau,r}$ and then update τ, r based on $\| V_{\tau,r}(\pi^*) - V(\pi^*) \|$.

Searching through models that yield similar returns when playing the same policy.

More generally, I am interested in how searches in different spaces, whether the value, the policy, or some parameters, …

Let's focus on gradient descent.

$$w_{t+1} = w_t - \eta \nabla f(w_t) \tag{1}$$

$$\tag{2}$$

It's dynamics are deptendent on the topology of its loss landscpace, which is determined by the search space and .

$$\max_{V} \mathop{\mathbb{E}}_{s \sim D} V(s) \tag{3}$$

$$\max_{\pi} \mathop{\mathbb{E}}_{s \sim D} V^{\pi}(s) \tag{4}$$

$$\max_{\theta} \mathop{\mathbb{E}}_{s \sim D} V_{\theta}(s) \tag{5}$$

$$\max_{\theta} \mathop{\mathbb{E}}_{s \sim D} V^{\pi_{\theta}}(s) \tag{6}$$

$$\max_{\phi} \mathop{\mathbb{E}}_{s \sim D} V^{\pi_{\theta_{\phi}}}(s) \tag{7}$$

$$\max_{\varphi} \mathop{\mathbb{E}}_{s \sim D} V^{\pi_{\theta_{\phi_{\varphi}}}}(s) \tag{8}$$

$$\tag{9}$$

We can pick the space we optimise in. Why would we want to pick one space over another?

- In which spaces can we do gradient descent?
- In which spaces can we do convex optimisation?
- In which spaces does momentum work well?
- …

Topology and dynamics

Ok, so if we parameterise our search space. We have now changed the topology of our search space.
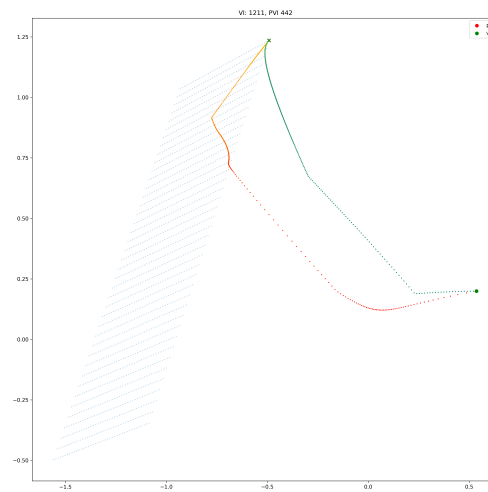
Q: How can we rationally pick the topology of our search space to accelerate learning?

- A well connected space? For all possible policies, there exists θ_1, θ_2 s.t. $\| \theta_1 - \theta_2 \|_2$ is small. (but that doesnt necessarily help… depends on the landscapce imposed by $\nabla_{\theta} V$)
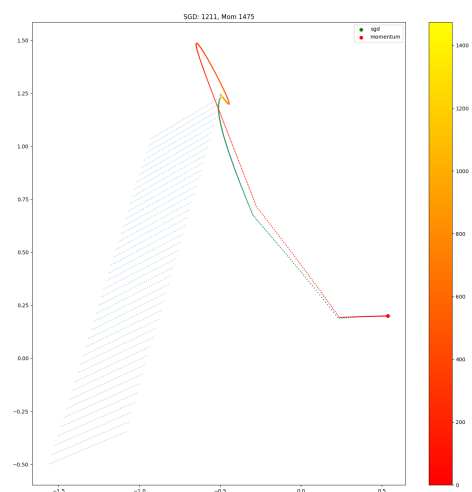- ???

See these gradient flows for example;

Pics?!?

Here are some examples …???

Figure 1: The optimisation dynamics of value iteration versus parameterised value iteration.

Figure 2: The optimisation dynamics of value iteration versus value iteration with momentum.

If we overparameterise the search space, then we can move between solutions in new ways. We can "tunnel" from A to B, without crossing C.

Intuition: Every point is closer, under some measure of distance?!? But. Momentum seems like it might be a bad thing here?

Accleration and parameterisation

Intuition. Something weird happens with momentum in overparameterised spaces.

It is necessary to consider the trajectory to study momentum. It depends on what has happened in the past. Can we construct a space of possible trajectories? What properties do trajectories have? They are connected by the update fn.

Continuous flow and its discretisation

A linear step of size, α, in parameter space, ie by gradient descent, is not necessrily a linear step in parameter space.
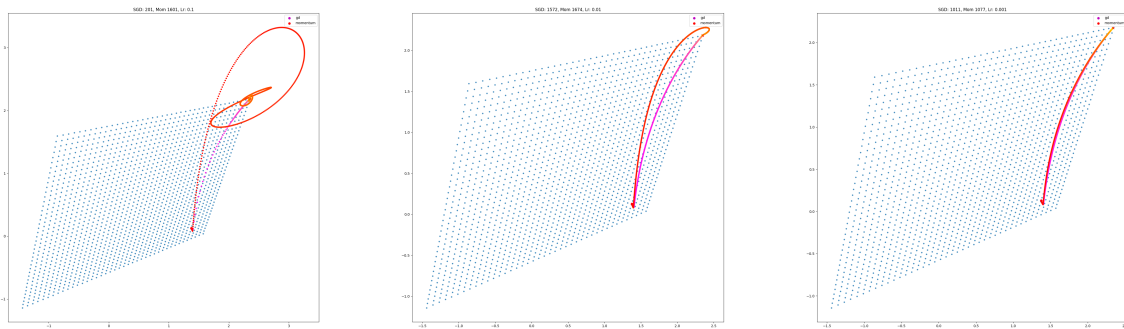
This is consistent with acceleration of gradient descent being a phenomena only possible in the discrete time setting. On symplectic optimization.

This phenomena can be explained by the exponential decay of the momentum terms.

$$m_{t+1} = m_t + \gamma \nabla f(w_t) \tag{10}$$

$$w_{t+1} = w_t - \eta(1 - \gamma)m_{t+1} \tag{11}$$

$$\tag{12}$$

As $\eta \to 0$, $(1 - \gamma) \cdot m_{t+1} \to \nabla f(w_t)$.

TODO, prove it.