Function Approximation

BLOE 498/598

Value functions vs. Policies

Policy: $\pi(s) \rightarrow a$

Difficult to engineer from scratch.

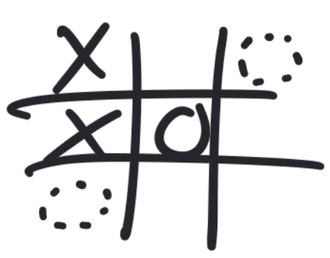
- Value functions are often more important than policies.
  - We can "easily" construct π from V.
  - It's easy to come up with a starting policy.

How Does AlphaGo work?

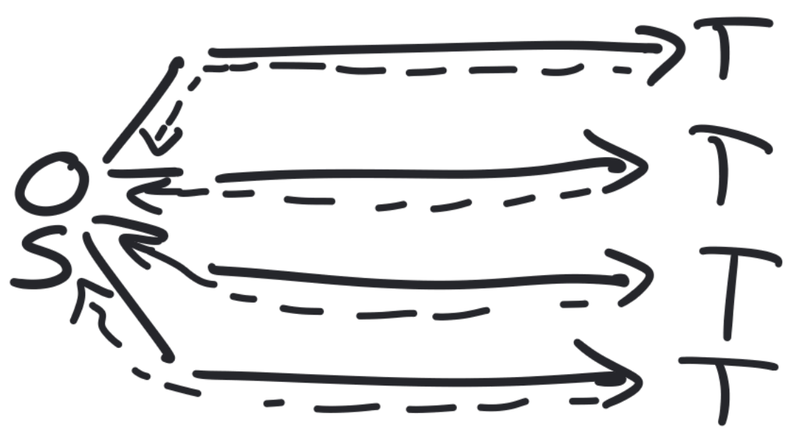
Began with a "human policy"!

$\pi_h \longrightarrow V_{\pi_h}$

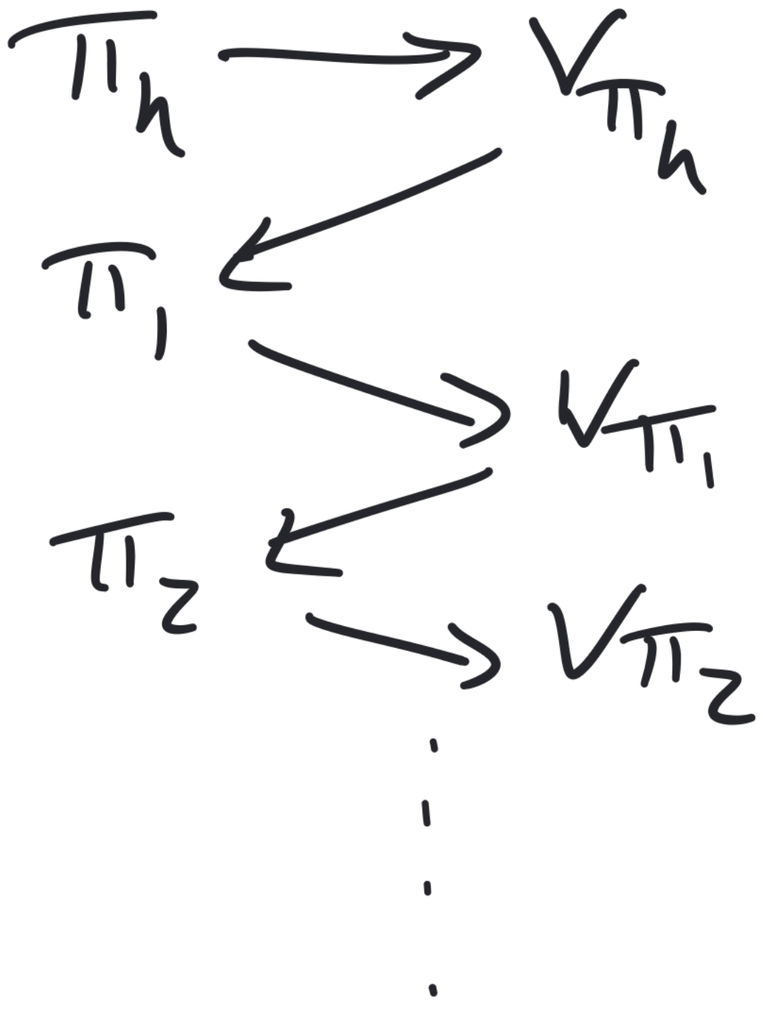
"Knowledge gradient" is very flat for Go.



Monte Carlo tree search.



RL for AlphaGo



AlphaGo Zero

No human policy

Why was Go so hard?

The state space is enormous.

Go is 19x19 grid.

Each square can have no, black, or white stones

$\Rightarrow$ # of possible states is 3^{19^2} > Everything

$V(s) \rightarrow \sum_i R_i$

↳ π choose the best action for every state

$\Rightarrow$ Function Approximation.

Artificial Neural Networks

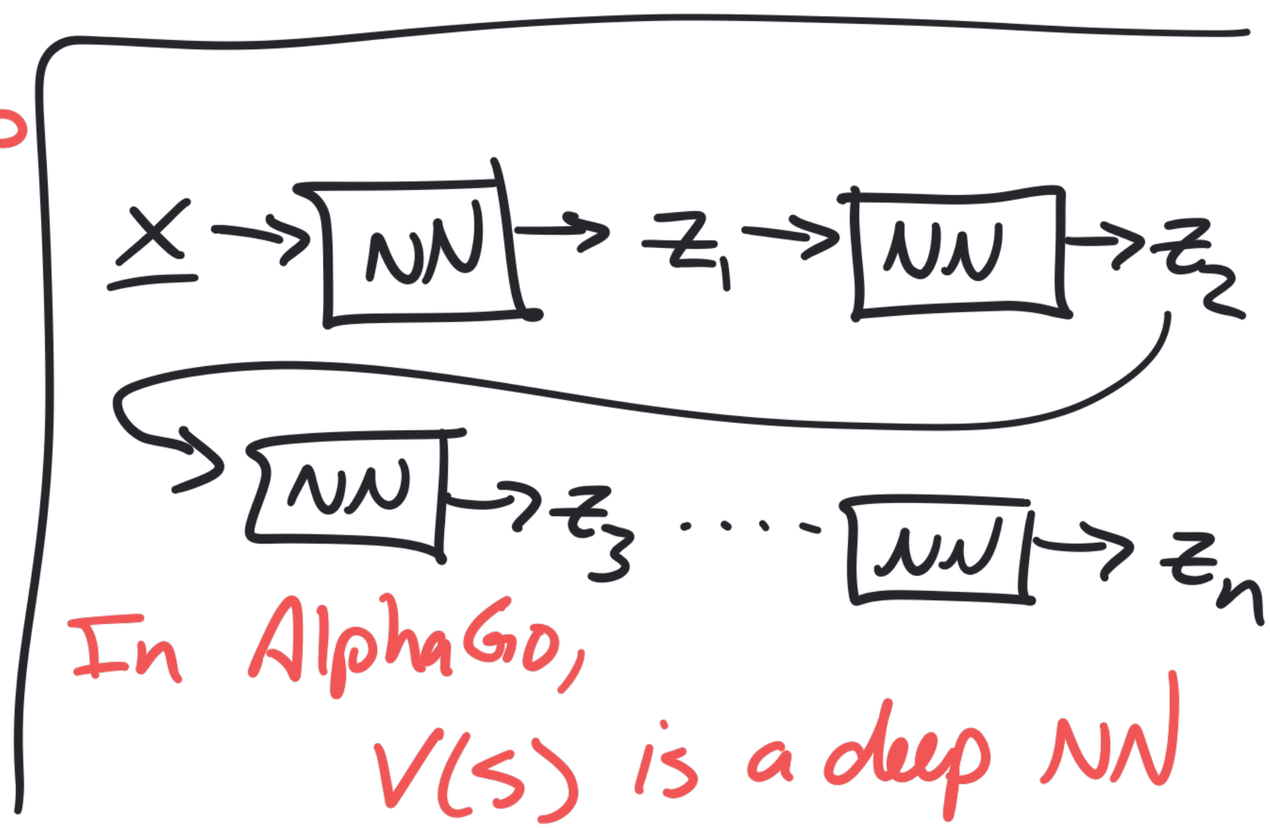
- ANN is an "overfitted" linear model with a nonlinearly transformed response.

$$\underline{x} \to \underline{A}\underline{x} + \underline{b} \to \underline{y} \xrightarrow{f} \to \underline{z}$$

$\underline{x} \in \mathbb{R}^n$, $\underline{A}\underline{x}+\underline{b} \in \mathbb{R}^p$, $\underline{y} \in \mathbb{R}^p$, $\underline{z} \in \mathbb{R}^p$

$\underline{A} \in \mathbb{R}^{p \times n}$, $\underline{b} \in \mathbb{R}^p$

n inputs
p outputs
pxn+p parameters



In AlphaGo, $V(s)$ is a deep NN

In AlphaGo:

$V(s)$ is a deep NN

$\pi(s)$ is also a deep NN

The value function is king.

Real-world RL.

- Self-driving cars
- Chat bots
- "Play" is expensive.
- How do we construct $V(s)$ from minimal data?