A Value Function Basis for Multi-Step Prediction

Andrew Jacobsen, Vincent Liu, Roshan Shariff,
Adam White, Martha White
Summer 2019

Preamble

Preamble

- [illegible]
- [illegible]

- want know many things
- Too lazy to learn them

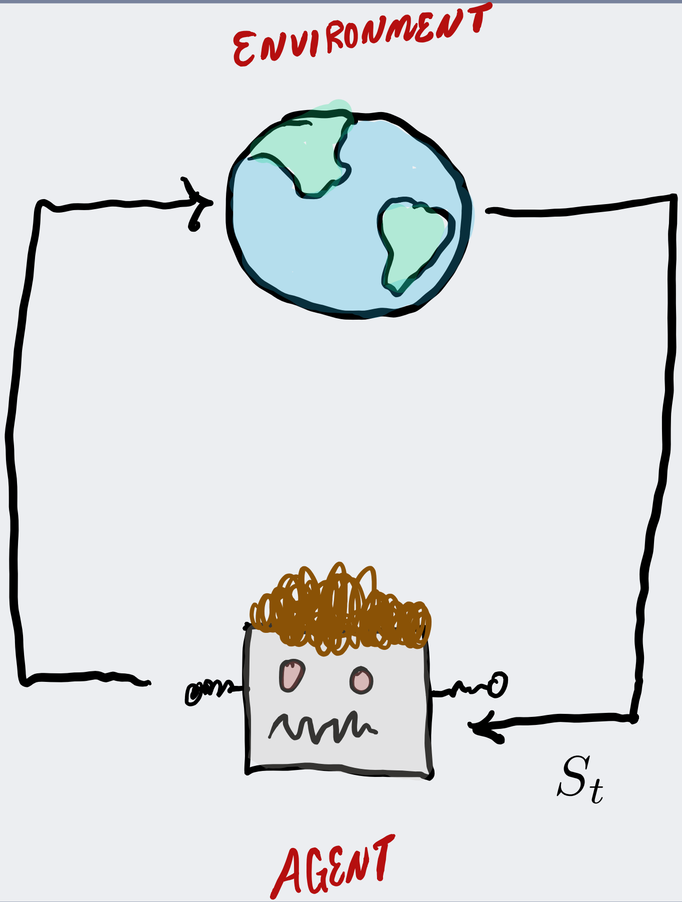
How can we ***know*** *as much as possible,*

while ***learning*** *as little as possible?*

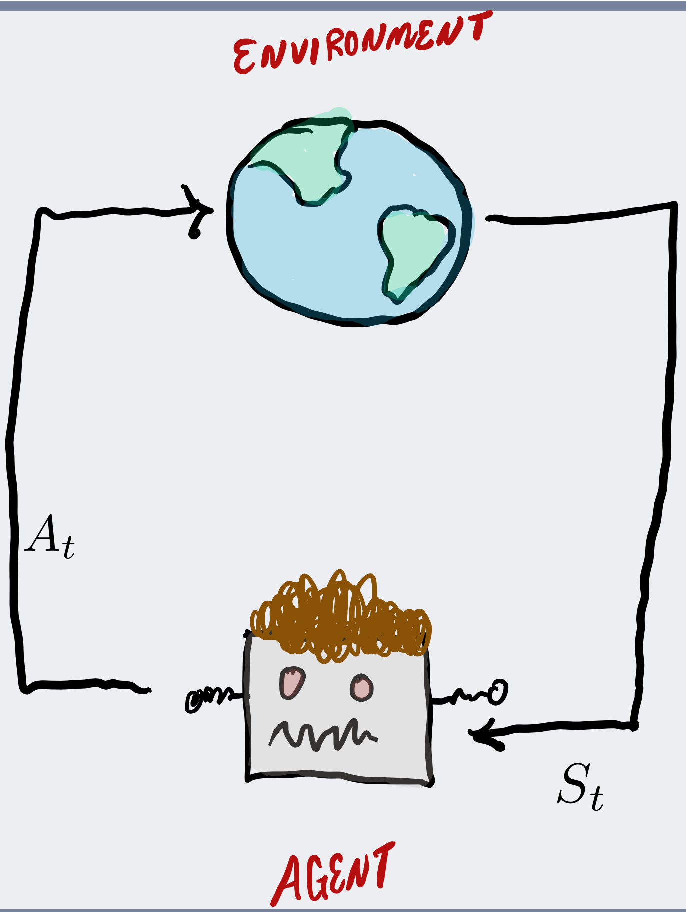
Overview

- Background
- Predicting at Every Time-scale
- Experimental Results
- A Basis of GVF Predictions

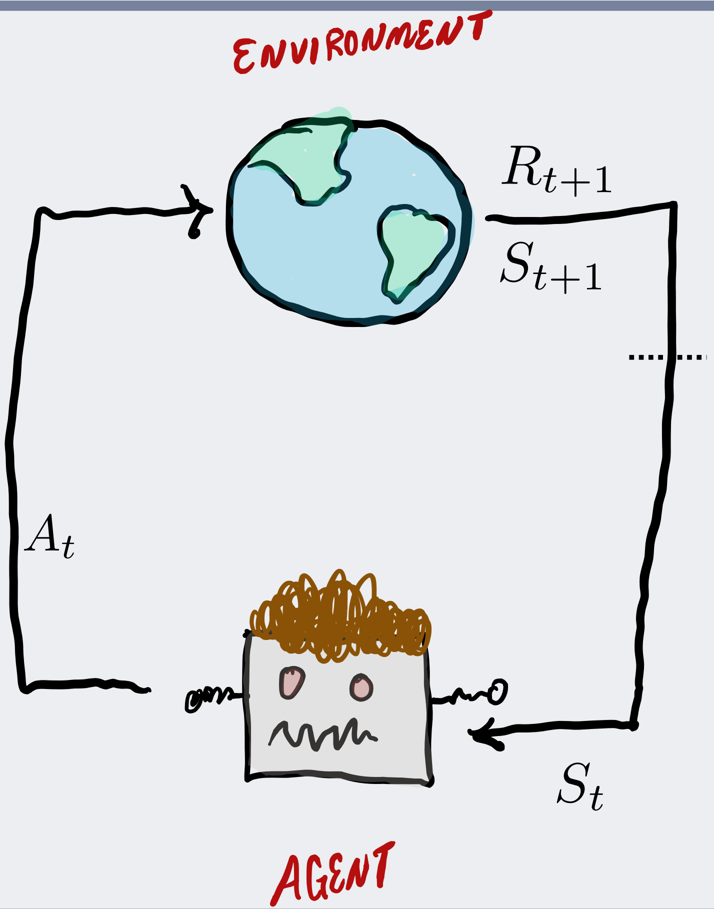
Background



Background



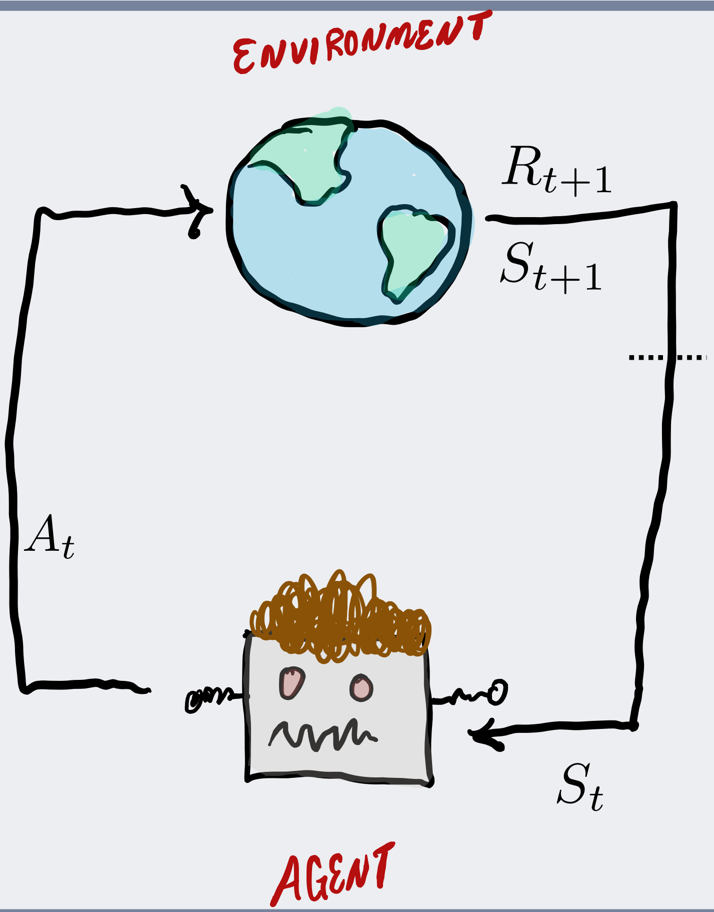
Background



Background

$$G_{t,\gamma} = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

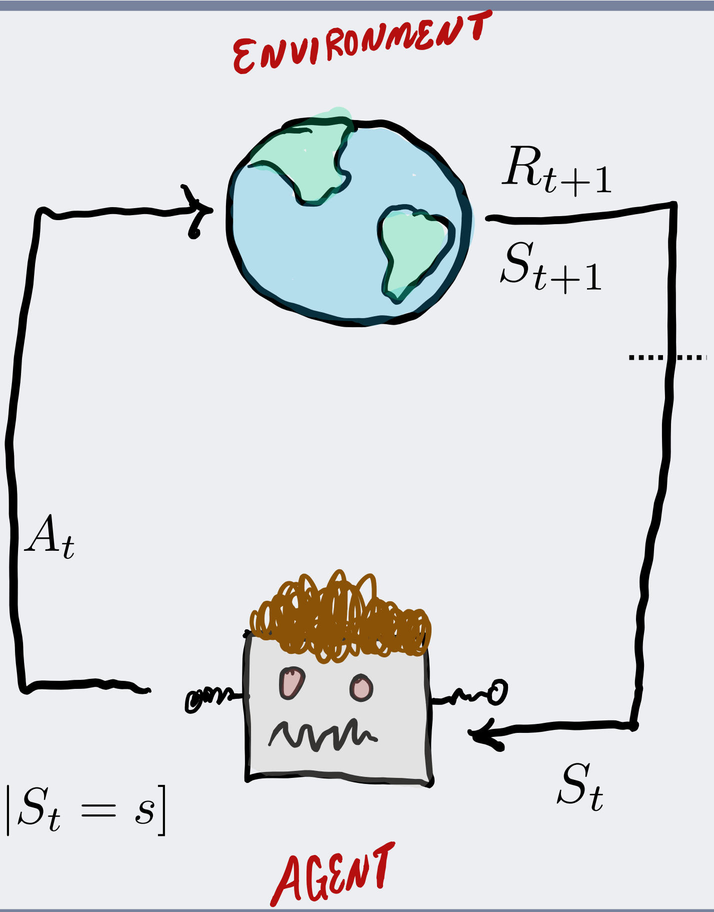
$$v_\gamma(s) = \mathbb{E}[G_{t,\gamma} | S_t = s, A_{t:\infty} \sim \pi]$$



Background

$$G_{t,\gamma} = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

$$v_\gamma(s) = \mathbb{E}[G_{t,\gamma} | S_t = s, A_{t:\infty} \sim \pi]$$



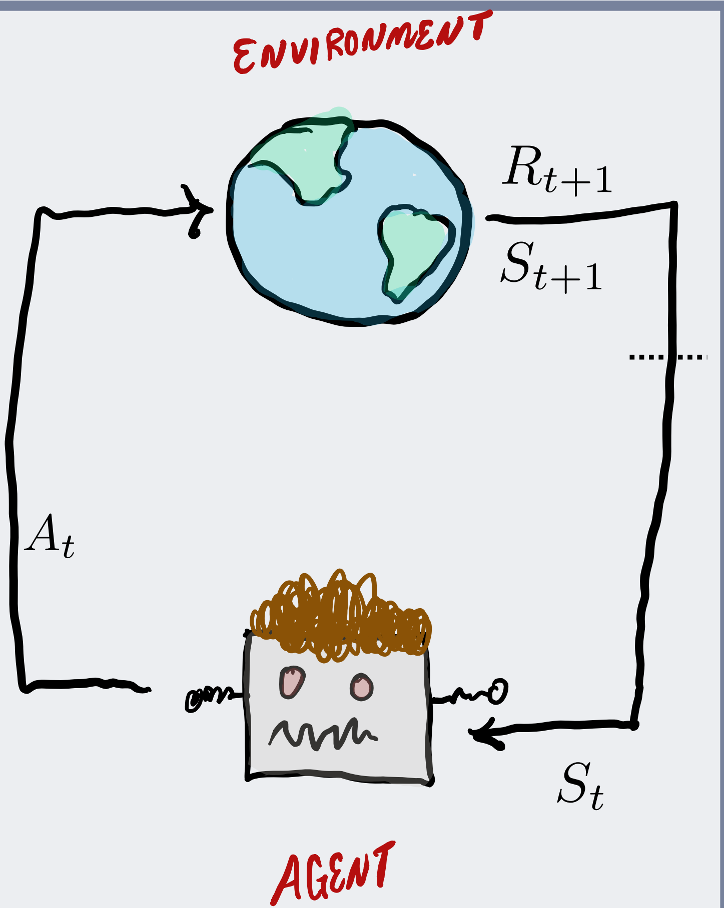
General Value Functions

$$v_\gamma(s) = \mathbb{E}[\sum_{k=0}^{\infty} C_{t+k+1} \left(\Pi_{i=1}^{k} \gamma(S_{t+k})\right) | S_t = s]$$

Background

$$G_{t,\gamma} = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

$$v_\gamma(s) = \mathbb{E}[G_{t,\gamma} | S_t = s, A_{t:\infty} \sim \pi]$$



General Value Functions

$$v_\gamma(s) = \mathbb{E}[\sum_{k=0}^{\infty} C_{t+k+1} \gamma^k | S_t = s]$$

$$y_1, y_2, \ldots, y_t, \ldots$$

$$y_1, y_2, \ldots, y_t, \ldots$$

$$G_{t,\gamma_1} = \sum_{k=0}^{\infty} \gamma_1^k y_{t+k+1}$$

$$G_{t,\gamma_2} = \sum_{k=0}^{\infty} \gamma_2^k y_{t+k+1}$$

$$G_{t,\gamma_3} = \sum_{k=0}^{\infty} \gamma_3^k y_{t+k+1}$$

$y_1, y_2, \ldots, y_t, \ldots$

$$G_{t,\gamma_1} = \sum_{k=0}^{\infty} \gamma_1^k y_{t+k+1} = \langle (1, \gamma_1, \gamma_1^2, \ldots)^\top, (y_{t+1}, y_{t+2}, \ldots)^\top \rangle$$

$$G_{t,\gamma_2} = \sum_{k=0}^{\infty} \gamma_2^k y_{t+k+1} = \langle (1, \gamma_2, \gamma_2^2, \ldots)^\top, (y_{t+1}, y_{t+2}, \ldots)^\top \rangle$$

$$G_{t,\gamma_3} = \sum_{k=0}^{\infty} \gamma_3^k y_{t+k+1} = \langle (1, \gamma_3, \gamma_3^2, \ldots)^\top, (y_{t+1}, y_{t+2}, \ldots)^\top \rangle$$

$$y_1, y_2, \ldots, y_t, \ldots$$

$$G_{t,\gamma_1} = \sum_{k=0}^{\infty} \gamma_1^k y_{t+k+1} = \langle (1, \gamma_1, \gamma_1^2, \ldots)^\top, (y_{t+1}, y_{t+2}, \ldots)^\top \rangle$$

$$G_{t,\gamma_2} = \sum_{k=0}^{\infty} \gamma_2^k y_{t+k+1} = \langle (1, \gamma_2, \gamma_2^2, \ldots)^\top, (y_{t+1}, y_{t+2}, \ldots)^\top \rangle$$

$$G_{t,\gamma_3} = \sum_{k=0}^{\infty} \gamma_3^k y_{t+k+1} = \langle (1, \gamma_3, \gamma_3^2, \ldots)^\top, (y_{t+1}, y_{t+2}, \ldots)^\top \rangle$$

$$\begin{pmatrix} G_{t,\gamma_1} \\ G_{t,\gamma_2} \\ G_{t,\gamma_3} \end{pmatrix} = \begin{pmatrix} 1 & \gamma_1 & \gamma_1^2 & \gamma_1^3 & \cdots \\ 1 & \gamma_2 & \gamma_2^2 & \gamma_2^3 & \cdots \\ 1 & \gamma_3 & \gamma_3^2 & \gamma_3^3 & \cdots \end{pmatrix} \vec{y}$$

$$y_1, y_2, \ldots, y_t, \ldots$$

$$G_{t,\gamma_1} = \sum_{k=0}^{\infty} \gamma_1^k y_{t+k+1} = \langle (1, \gamma_1, \gamma_1^2, \ldots)^\top, (y_{t+1}, y_{t+2}, \ldots)^\top \rangle$$

$$G_{t,\gamma_2} = \sum_{k=0}^{\infty} \gamma_2^k y_{t+k+1} = \langle (1, \gamma_2, \gamma_2^2, \ldots)^\top, (y_{t+1}, y_{t+2}, \ldots)^\top \rangle$$

$$G_{t,\gamma_3} = \sum_{k=0}^{\infty} \gamma_3^k y_{t+k+1} = \langle (1, \gamma_3, \gamma_3^2, \ldots)^\top, (y_{t+1}, y_{t+2}, \ldots)^\top \rangle$$

$$\begin{pmatrix} G_{t,\gamma_1} \\ G_{t,\gamma_2} \\ G_{t,\gamma_3} \end{pmatrix} = \begin{pmatrix} 1 & \gamma_1 & \gamma_1^2 & \gamma_1^3 & \cdots \\ 1 & \gamma_2 & \gamma_2^2 & \gamma_2^3 & \cdots \\ 1 & \gamma_3 & \gamma_3^2 & \gamma_3^3 & \cdots \end{pmatrix} \vec{y} = \Gamma \vec{y}$$

Want: $\hat{y} \approx \vec{y}$

Want: $\hat{y} \approx \vec{y}$

$$\Gamma = \begin{pmatrix} 1 & \gamma_1 & \gamma_1^2 & \gamma_1^3 & \cdots \\ 1 & \gamma_2 & \gamma_2^2 & \gamma_2^3 & \cdots \\ 1 & \gamma_3 & \gamma_3^2 & \gamma_3^3 & \cdots \end{pmatrix}$$

$$\Gamma^\top \vec{\theta} = \theta_1 \vec{\gamma}_1 + \theta_2 \vec{\gamma}_2 + \theta_3 \vec{\gamma}_3 \approx \vec{y}$$

$$\Gamma = \begin{pmatrix} 1 & \gamma_1 & \gamma_1^2 & \gamma_1^3 & \cdots \\ 1 & \gamma_2 & \gamma_2^2 & \gamma_2^3 & \cdots \\ 1 & \gamma_3 & \gamma_3^2 & \gamma_3^3 & \cdots \end{pmatrix}$$

Want: $\hat{y} \approx \vec{y}$

$$\Gamma^\top \vec{\theta} = \theta_1 \vec{\gamma}_1 + \theta_2 \vec{\gamma}_2 + \theta_3 \vec{\gamma}_3 \approx \vec{y}$$

$$\implies \Gamma\Gamma^\top \vec{\theta} \approx \Gamma \vec{y}$$

Want: $\hat{y} \approx \vec{y}$

$$\Gamma = \begin{pmatrix} 1 & \gamma_1 & \gamma_1^2 & \gamma_1^3 & \cdots \\ 1 & \gamma_2 & \gamma_2^2 & \gamma_2^3 & \cdots \\ 1 & \gamma_3 & \gamma_3^2 & \gamma_3^3 & \cdots \end{pmatrix}$$

$$\Gamma^\top \vec{\theta} = \theta_1 \vec{\gamma}_1 + \theta_2 \vec{\gamma}_2 + \theta_3 \vec{\gamma}_3 \approx \vec{y}$$

$$\implies \Gamma\Gamma^\top \vec{\theta} \approx \Gamma \vec{y}$$

$$\implies \boxed{\vec{\theta} = (\Gamma\Gamma^\top)^{-1} \Gamma \vec{y}}$$

Want: $\hat{y} \approx \vec{y}$

$$\Gamma = \begin{pmatrix} 1 & \gamma_1 & \gamma_1^2 & \gamma_1^3 & \cdots \\ 1 & \gamma_2 & \gamma_2^2 & \gamma_2^3 & \cdots \\ 1 & \gamma_3 & \gamma_3^2 & \gamma_3^3 & \cdots \end{pmatrix}$$

$$\Gamma^\top \vec{\theta} = \theta_1 \vec{\gamma}_1 + \theta_2 \vec{\gamma}_2 + \theta_3 \vec{\gamma}_3 \approx \vec{y}$$

$$\implies \Gamma \Gamma^\top \vec{\theta} \approx \Gamma \vec{y}$$

$$\implies \boxed{\vec{\theta} = (\Gamma \Gamma^\top)^{-1} \underline{\Gamma \vec{y}}}$$

$$\underline{\Gamma \vec{y}} \rightarrow \begin{pmatrix} G_{t,\gamma_1} \\ G_{t,\gamma_2} \\ G_{t,\gamma_3} \end{pmatrix}$$

$$\Gamma = \begin{pmatrix} 1 & \gamma_1 & \gamma_1^2 & \gamma_1^3 & \cdots \\ 1 & \gamma_2 & \gamma_2^2 & \gamma_2^3 & \cdots \\ 1 & \gamma_3 & \gamma_3^2 & \gamma_3^3 & \cdots \end{pmatrix}$$

Want: $\hat{y} \approx \vec{y}$

$$\Gamma^\top \vec{\theta} = \theta_1 \vec{\gamma}_1 + \theta_2 \vec{\gamma}_2 + \theta_3 \vec{\gamma}_3 \approx \vec{y}$$

$$\implies \Gamma\Gamma^\top \vec{\theta} \approx \Gamma \vec{y}$$

$$\implies \boxed{\vec{\theta} = (\underline{\Gamma\Gamma^\top})^{-1} \underline{\Gamma\vec{y}}}$$

$$[\Gamma\Gamma^\top]_{i,j} = \sum_{k=0}^{\infty} \gamma_i^k \gamma_j^k = \frac{1}{1-\gamma_i\gamma_j}$$

$$\begin{pmatrix} G_{t,\gamma_1} \\ G_{t,\gamma_2} \\ G_{t,\gamma_3} \end{pmatrix}$$

$$\hat{y} = \Gamma^\top \vec{\theta}$$

$$\vec{\theta} = (\Gamma\Gamma^\top)^{-1} \begin{pmatrix} G_{t,\gamma_1} \\ G_{t,\gamma_2} \\ G_{t,\gamma_3} \end{pmatrix}$$

$$\hat{y} = \Gamma^\top \vec{\theta} \qquad \vec{\theta} = (\Gamma\Gamma^\top)^{-1} \begin{pmatrix} G_{t,\gamma_1} \\ G_{t,\gamma_2} \\ G_{t,\gamma_3} \end{pmatrix}$$

$$y_{t+n} \approx \hat{y}[n] = \sum_{i=1}^{k} \theta_i \gamma_i^{n-1}$$

$$\hat{y} = \Gamma^\top \vec{\theta} \qquad \vec{\theta} = (\Gamma\Gamma^\top)^{-1} \begin{pmatrix} G_{t,\gamma_1} \\ G_{t,\gamma_2} \\ G_{t,\gamma_3} \end{pmatrix}$$

$$y_{t+n} \approx \hat{y}[n] = \sum_{i=1}^{k} \theta_i \gamma_i^{n-1}$$

$$G_{t,\gamma} = \langle \vec{\gamma}, \vec{y} \rangle \approx \langle \vec{\gamma}, \hat{y} \rangle = \sum_{i=1}^{k} \frac{\theta_i}{1 - \gamma\gamma_i}$$

$$\hat{y} = \Gamma^\top \vec{\theta}$$

$$\vec{\theta} = (\Gamma\Gamma^\top)^{-1} \begin{pmatrix} v_{\gamma_1}(s) \\ v_{\gamma_2}(s) \\ v_{\gamma_3}(s) \end{pmatrix}$$

$$\mathbb{E}[y_{t+n} | S_t = s] \approx \hat{y}[n] = \sum_{i=1}^{k} \theta_i \gamma_i^{n-1}$$

$$v_\gamma(s) = \langle \vec{\gamma}, \vec{y} \rangle \approx \langle \vec{\gamma}, \hat{y} \rangle = \sum_{i=1}^{k} \frac{\theta_i}{1 - \gamma \gamma_i}$$

$$\hat{y} = \Gamma^\top \vec{\theta} \qquad \vec{\theta} = (\Gamma\Gamma^\top)^{-1} \begin{pmatrix} v_{\gamma_1}(s) \\ v_{\gamma_2}(s) \\ v_{\gamma_3}(s) \end{pmatrix}$$

"GVF Basis"

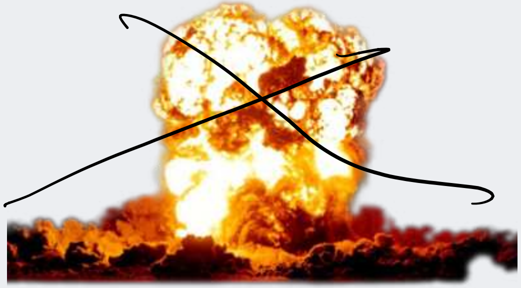
$$\mathbb{E}[y_{t+n} | S_t = s] \approx \hat{y}[n] = \sum_{i=1}^{k} \theta_i \gamma_i^{n-1}$$

$$v_\gamma(s) = \langle \vec{\gamma}, \vec{y} \rangle \approx \langle \vec{\gamma}, \hat{y} \rangle = \sum_{i=1}^{k} \frac{\theta_i}{1 - \gamma\gamma_i}$$

"Inferred GVF"

Results

Results

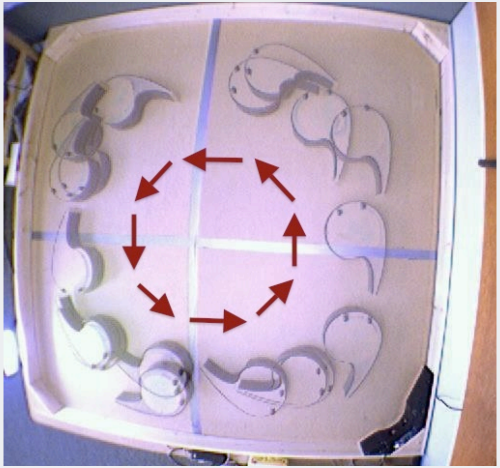


It works ~~AMAZING~~ OKAY

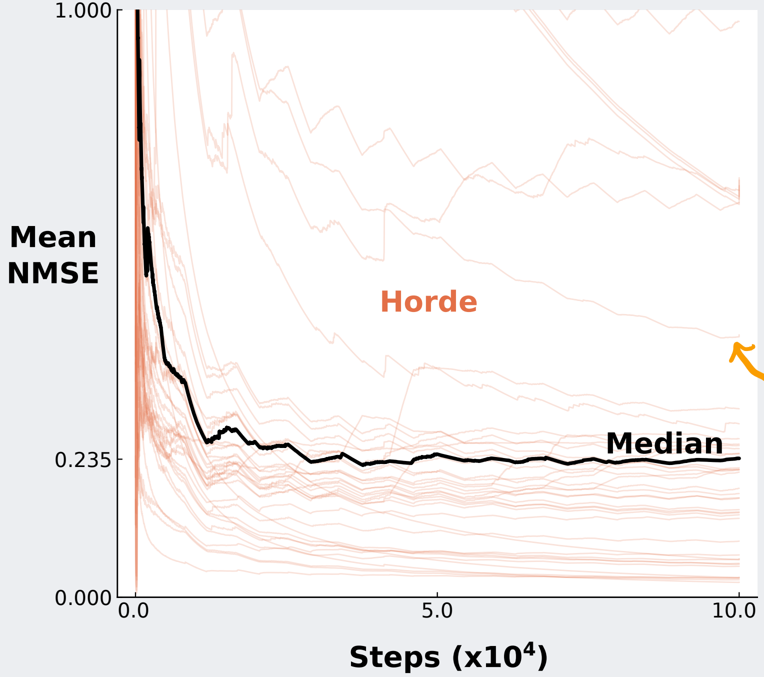
Task

For each sensor,

- Learn 7 GVFs, with discounts $\gamma_i = 1 - 2^{-i}$ for $i = 1, \ldots, 7$
- Infer the values of 100 GVFs with randomly selected discounts
- Infer the sensor reading 30 steps in the future



Predicting Sensor Readings: 100 GVFs

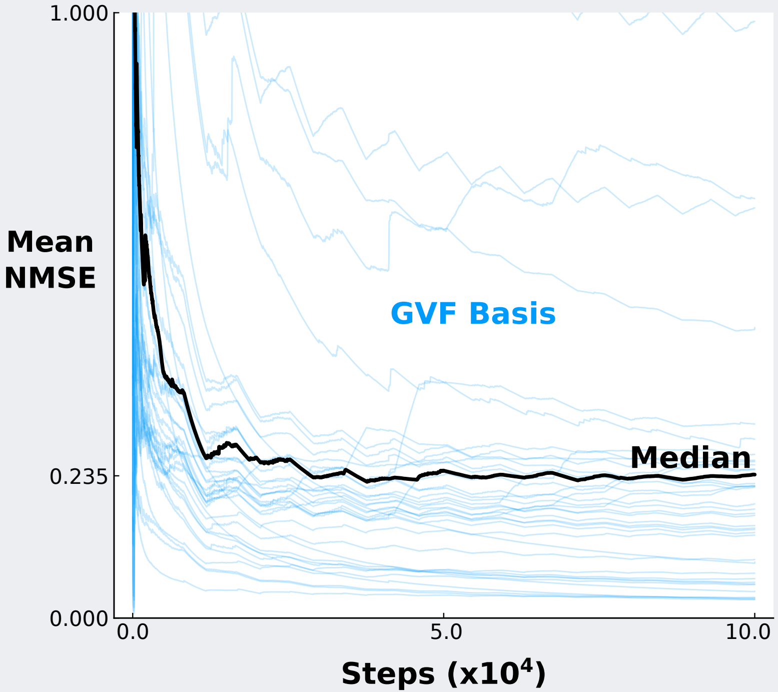


Horde: directly learn each GVF using TD

$$\mathrm{NMSE(T)} = \frac{1}{T}\frac{\sum_{t=1}^{T}(v_\gamma(S_t) - G_{t,\gamma})^2}{var(G_{:,\gamma})}$$

NMSE Averaged over test-Set GVFS for Some Sensor.

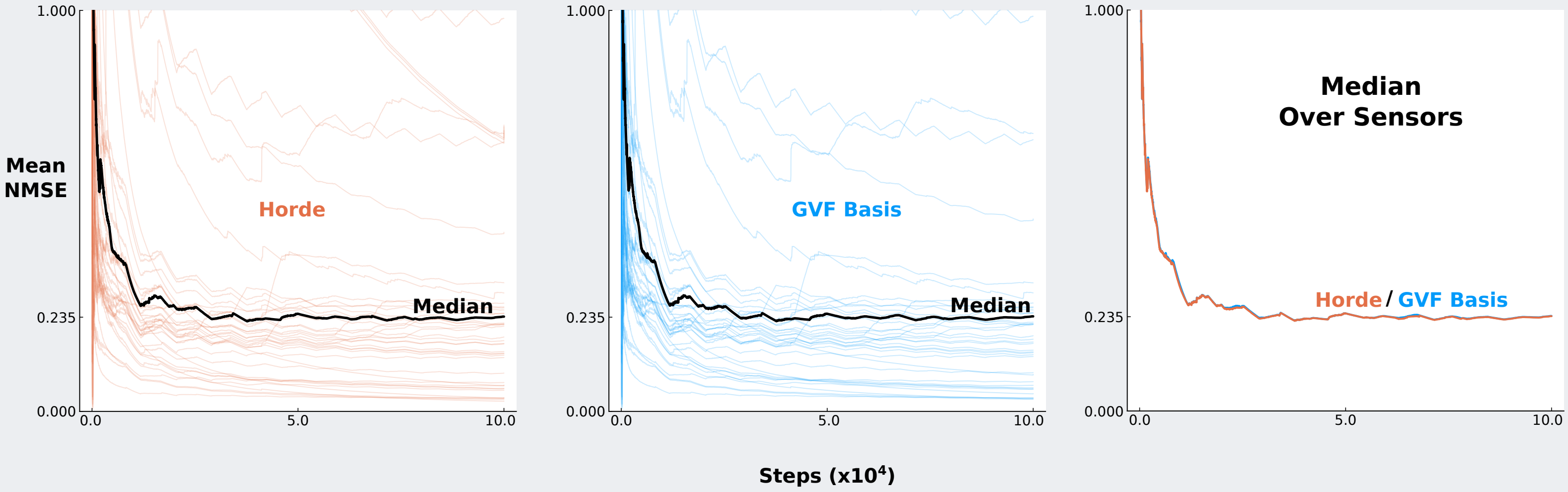
Predicting Sensor Readings: 100 GVFs



GVF Basis: Learn 7 GVFs,
infer the 100 GVFs of interest

$$\text{NMSE(T)} = \frac{1}{T}\frac{\sum_{t=1}^{T}(v_\gamma(S_t) - G_{t,\gamma})^2}{var(G_{:,\gamma})}$$

Predicting Sensor Readings: 100 GVFs

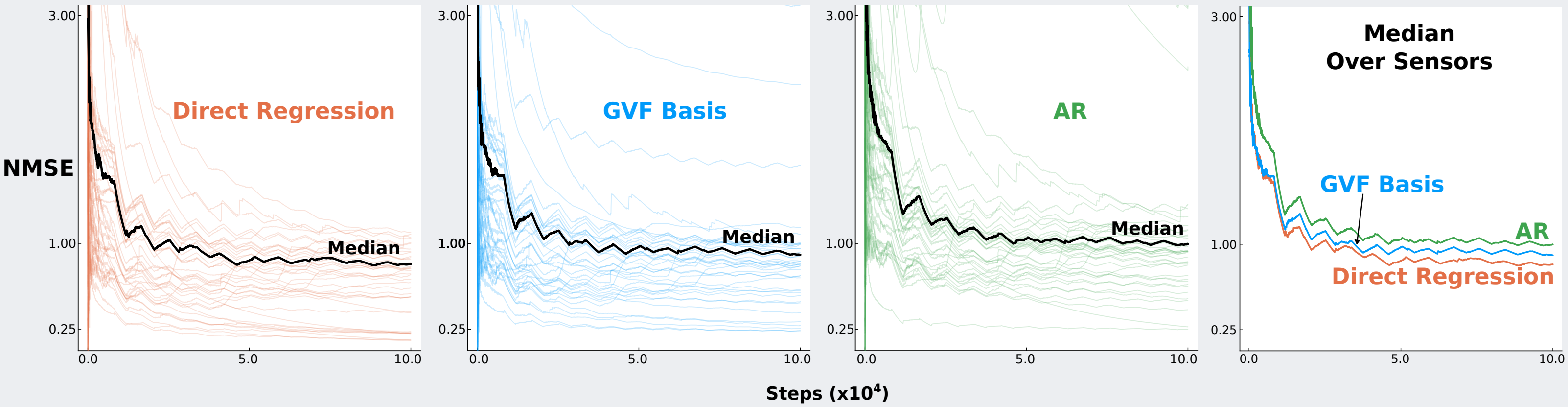


Horde: directly learn each GVF using TD

GVF Basis: Learn 7 GVFs, infer the 100 GVFs of interest

$$\text{NMSE(T)} = \frac{1}{T} \frac{\sum_{t=1}^{T} (v_\gamma(S_t) - G_{t,\gamma})^2}{var(G_{:,\gamma})}$$

Predicting Sensor Readings: 30-step



Direct Regression: tile-coded features, directly trained to predict 30 steps ahead

GVF Basis: Learn 7 GVFs (per sensor), infer 30 steps ahead

AR: history of observations given as input, directly trained to predict 30 steps ahead

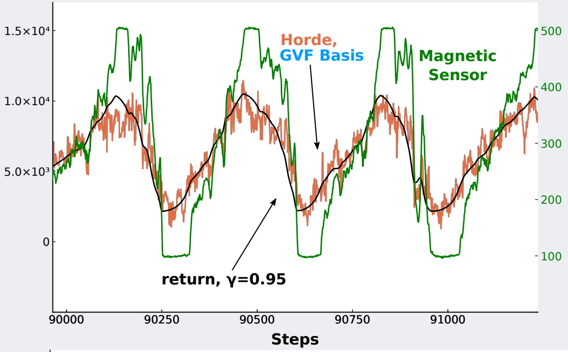
$$\text{NMSE(T)} = \frac{1}{T}\frac{\sum_{t=1}^{T}(\hat{y}_{t+30} - y_{t+30})^2}{var(y_:)}$$

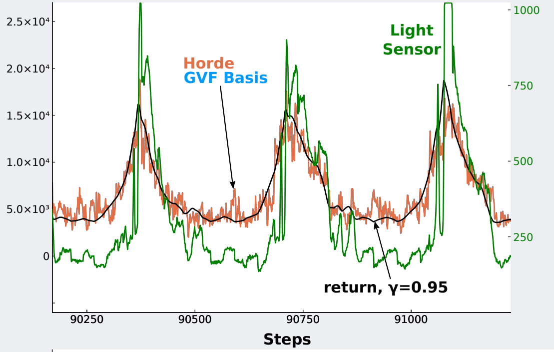
Predicting Sensor Readings

Magnetic Sensor Predictions

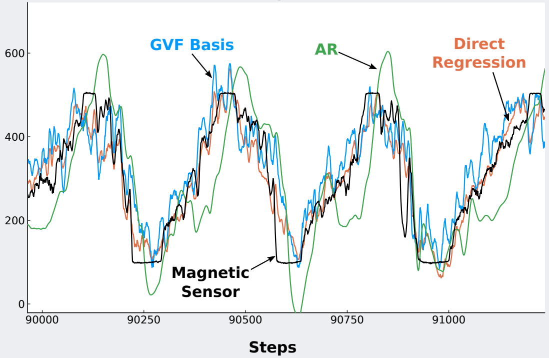
Light Sensor Predictions

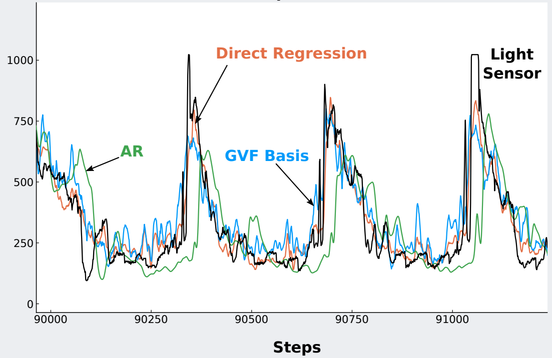
Predicting GVFs





Predicting 30 steps ahead





What should the discounts ideally be?

A simple case: Consider a finite-state Markov Reward Process with Transition matrix P: $P_{i,j} = Pr\{S_{t+1} = j | S_t = i\}$

Assume P Diagonalizable: $P = U\Gamma U^{-1}$ where $\Gamma_{i,i} = \gamma_i$

What should the discounts ideally be?

A simple case: Consider a finite-state Markov Reward Process with Transition matrix P: $P_{i,j} = Pr\{S_{t+1} = j | S_t = i\}$

Assume P Diagonalizable: $P = U \Gamma U^{-1}$ where $\Gamma_{i,i} = \gamma_i$

Let $\vec{V}_\gamma \in \mathbb{R}^{|S|}$ s.t. $\vec{V}_\gamma[s] = v_\gamma(s)$

Let $\vec{r}_t^{\,(n)} \in \mathbb{R}^{|S|}$ s.t. $\vec{r}_t^{\,(n)}[s] = \mathbb{E}[R_{t+n} | S_t = s]$

What should the discounts ideally be?

A simple case: Consider a finite-state Markov Reward Process with Transition matrix P: $P_{i,j} = Pr\{S_{t+1} = j | S_t = i\}$

Assume P Diagonalizable: $P = U \Gamma U^{-1}$ where $\Gamma_{i,i} = \gamma_i$

Let $\vec{V}_\gamma \in \mathbb{R}^{|S|}$ s.t. $\vec{V}_\gamma[s] = v_\gamma(s)$

Let $\vec{r}_t^{\,(n)} \in \mathbb{R}^{|S|}$ s.t. $\vec{r}_t^{\,(n)}[s] = \mathbb{E}[R_{t+n} | S_t = s]$

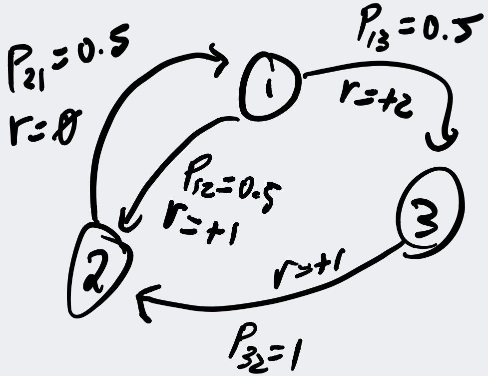
Then $\{\vec{V}_{\gamma_i} : \gamma_i = \Gamma_{i,i}\}$ Is a basis of the space $\{\vec{r}_t^{\,(n)} : n \in \mathbb{N}\}$

Thanks.
Questions?

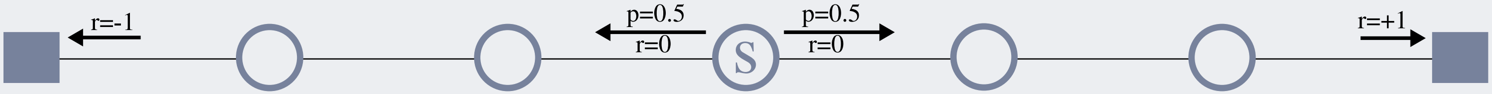
What should the discounts ideally be?

A simple case: Consider a finite-state Markov Reward Process with Transition matrix P: $P_{i,j} = Pr\{S_{t+1} = j | S_t = i\}$

Assume P Diagonalizable: $P = U\Gamma U^{-1}$ where $\Gamma_{i,i} = \gamma_i$



19 state Random walk

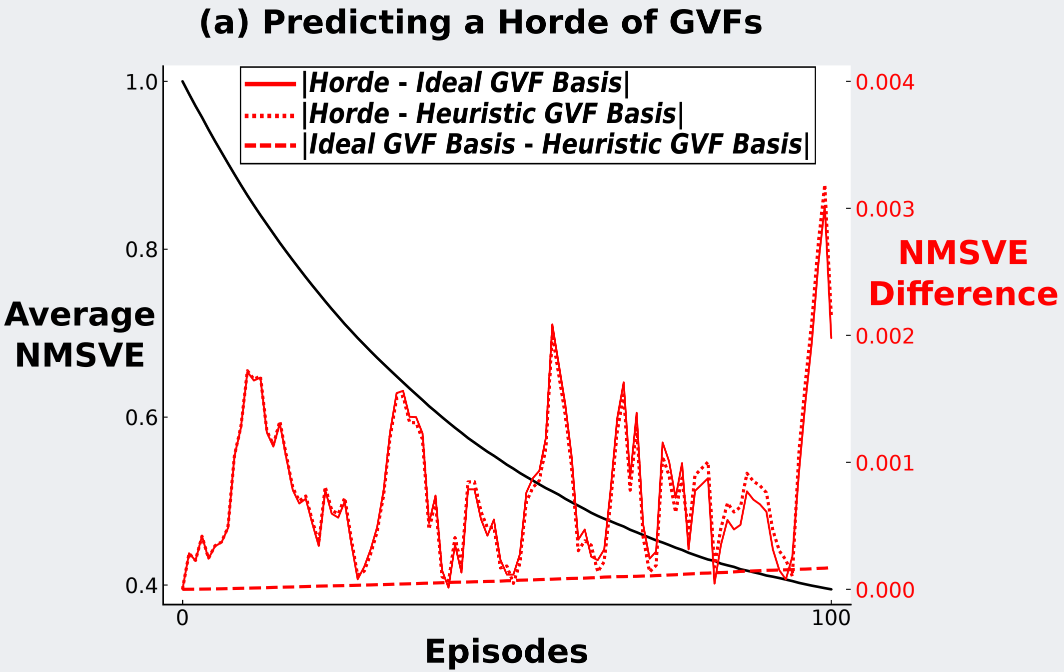


Task: Predict 10,000 value functions, with $\gamma_i \sim \mathrm{Uniform}(0,1)$

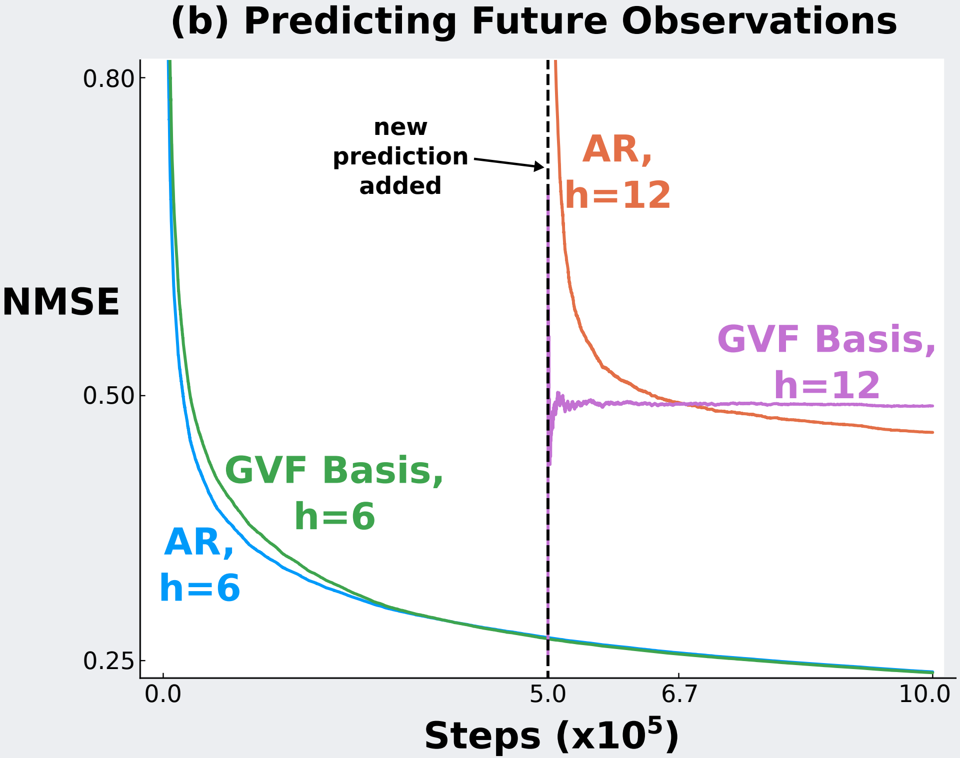
Learners:

- ***Horde:*** learn each value function directly
- ***Ideal GVF Basis:*** learn value functions with $\gamma_i = \Gamma_{i,i}$, infer the 10,000 value functions
- ***Heuristic GVF Basis:*** learn 19 value functions with discounts linearly spaced between (0,1), infer the 10,000 value functions

19 State Random Walk



Synthetic Tasks



What should the discounts ideally be?

A simple case: Consider a finite-state Markov Reward Process with Transition matrix P: $P_{i,j} = Pr\{S_{t+1} = j | S_t = i\}$

Assume P Diagonalizable: $P = U \Gamma U^{-1}$ where $\Gamma_{i,i} = \gamma_i$

$\exists$ a basis $\vec{u}_1, \ldots, \vec{u}_{|S|}$ s.t. $P\vec{u}_i = \gamma_i \vec{u}_i$