

Log-optimal Investment in Markovian Environments

Csaba Szepesvári

Computer and Automation Research Institute of the
Hungarian Academy of Sciences
Kende u. 13-17, Budapest 1111, Hungary
E-mail: szcsaba@sztaki.hu

Morgen Stanley Quantitative and Financial Mathematics
Conference
21 October, 2005

Co-workers: **Remi Munos, András Antos**



Outline

- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment
 - FX Markets
 - Stock Market
- 3 Solution Methods for MDPs
 - Classics
 - Approximate Methods
 - Does it Work?
- 4 Application to Log-optimal Investment
- 5 Conclusions

Outline

- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment
 - FX Markets
 - Stock Market
- 3 Solution Methods for MDPs
 - Classics
 - Approximate Methods
 - Does it Work?
- 4 Application to Log-optimal Investment
- 5 Conclusions

Outline

- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment
 - FX Markets
 - Stock Market
- 3 Solution Methods for MDPs
 - Classics
 - Approximate Methods
 - Does it Work?
- 4 Application to Log-optimal Investment
- 5 Conclusions

Outline

- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment
 - FX Markets
 - Stock Market
- 3 Solution Methods for MDPs
 - Classics
 - Approximate Methods
 - Does it Work?
- 4 Application to Log-optimal Investment
- 5 Conclusions

Outline

- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment
 - FX Markets
 - Stock Market
- 3 Solution Methods for MDPs
 - Classics
 - Approximate Methods
 - Does it Work?
- 4 Application to Log-optimal Investment
- 5 Conclusions

Outline

- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment
 - FX Markets
 - Stock Market
- 3 Solution Methods for MDPs
 - Classics
 - Approximate Methods
 - Does it Work?
- 4 Application to Log-optimal Investment
- 5 Conclusions

Markovian Decision Problems

Definition

$(\mathcal{X}, \mathcal{A}, P, r)$ MDP:

- State space $\mathcal{X} (\subset \mathbb{R}^d)$
- Action space \mathcal{A}
- Transition probabilities $P(\cdot|x, a)$
- Reward function $r(x, a)$.

Markovian Decision Problems

Definition

$(\mathcal{X}, \mathcal{A}, P, r)$ MDP:

- State space $\mathcal{X} (\subset \mathbb{R}^d)$
- Action space \mathcal{A}
- Transition probabilities $P(\cdot|x, a)$
- Reward function $r(x, a)$.

Markovian Decision Problems

Definition

$(\mathcal{X}, \mathcal{A}, P, r)$ MDP:

- State space $\mathcal{X} (\subset \mathbb{R}^d)$
- Action space \mathcal{A}
- Transition probabilities $P(\cdot|x, a)$
- Reward function $r(x, a)$.

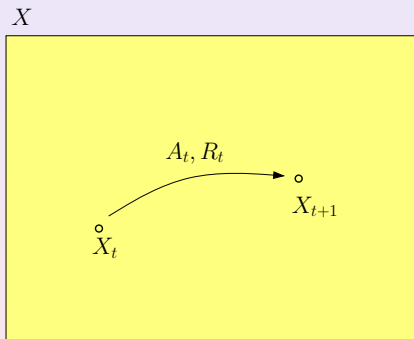
Markovian Decision Problems

Definition

$(\mathcal{X}, \mathcal{A}, P, r)$ MDP:

- State space $\mathcal{X} (\subset \mathbb{R}^d)$
- Action space \mathcal{A}
- Transition probabilities $P(\cdot|x, a)$
- Reward function $r(x, a)$.

Process View



$$\pi : X \rightarrow A$$

$$V^\pi(x) = E[\sum_{t=0}^{\infty} \gamma^t R_t | X_0 = x, \pi]$$

$$Q^\pi(x, a) = E[\sum_{t=0}^{\infty} \gamma^t R_t | X_0 = x, A_0 = a, \pi]$$

$$0 < \gamma < 1$$

Reinforcement Learning

Goal: Finding an optimal policy

- .. in an unknown MDP by just observing a trajectory
- .. when a generative model of the MDP is given
 - ..large MDP
- .. when a model of the MDP is given

Reinforcement Learning

Goal: Finding an optimal policy

- .. in an unknown MDP by just observing a trajectory
- .. when a generative model of the MDP is given
 - ..large MDP
- .. when a model of the MDP is given

Reinforcement Learning

Goal: Finding an optimal policy

- .. in an unknown MDP by just observing a trajectory
- .. when a generative model of the MDP is given
 - ..large MDP
- .. when a model of the MDP is given

Reinforcement Learning

Goal: Finding an optimal policy

- .. in an unknown MDP by just observing a trajectory
- .. when a generative model of the MDP is given
..large MDP
- .. when a model of the MDP is given

Outline

- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment**
 - FX Markets**
 - Stock Market
- 3 Solution Methods for MDPs
 - Classics
 - Approximate Methods
 - Does it Work?
- 4 Application to Log-optimal Investment
- 5 Conclusions

Simple FX Example

- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- $p_{12}(t)$ – amount of dollar purchased for 1 euro
- W_t – wealth (calc'ed in dollars)
- α_t – relative portfolio; proportion of wealth in **euros**

Simple FX Example

- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- $p_{12}(t)$ – amount of dollar purchased for 1 euro
- W_t – wealth (calc'ed in dollars)
- α_t – relative portfolio; proportion of wealth in **euros**

Simple FX Example

- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- $p_{12}(t)$ – amount of dollar purchased for 1 euro
- W_t – wealth (calc'ed in dollars)
- α_t – relative portfolio; proportion of wealth in **euros**

Simple FX Example

- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- $p_{12}(t)$ – amount of dollar purchased for 1 euro
- W_t – wealth (calc'ed in dollars)
- α_t – relative portfolio; proportion of wealth in **euros**

Simple FX Example

- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- $p_{12}(t)$ – amount of dollar purchased for 1 euro
- W_t – wealth (calc'ed in dollars)
- α_t – relative portfolio; proportion of wealth in **euros**

Simple FX Example

- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- $p_{12}(t)$ – amount of dollar purchased for 1 euro
- W_t – wealth (calc'ed in dollars)
- α_t – relative portfolio; proportion of wealth in **euros**

FX: Dynamics and Bid-Ask Spread

- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- Dynamics of dollar's exchange rate:

$$\frac{p_{12}(t+1)}{p_{12}(t)} = \rho_{t+1}$$

- Bid-ask spread:

$$p_{12}(t+1)p_{21}(t+1) = \eta_{t+1}^2 < 1$$

FX: Dynamics and Bid-Ask Spread

- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- Dynamics of dollar's exchange rate:

$$\frac{p_{12}(t+1)}{p_{12}(t)} = \rho_{t+1}$$

- Bid-ask spread:

$$p_{12}(t+1)p_{21}(t+1) = \eta_{t+1}^2 < 1$$

FX: Dynamics and Bid-Ask Spread

- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- Dynamics of dollar's exchange rate:

$$\frac{p_{12}(t+1)}{p_{12}(t)} = \rho_{t+1}$$

- Bid-ask spread:

$$p_{12}(t+1)p_{21}(t+1) = \eta_{t+1}^2 < 1$$

FX: Dynamics and Bid-Ask Spread

- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- Dynamics of dollar's exchange rate:

$$\frac{p_{12}(t+1)}{p_{12}(t)} = \rho_{t+1}$$

- Bid-ask spread:

$$p_{12}(t+1)p_{21}(t+1) = \eta_{t+1}^2 < 1$$

FX: Dynamics and Bid-Ask Spread

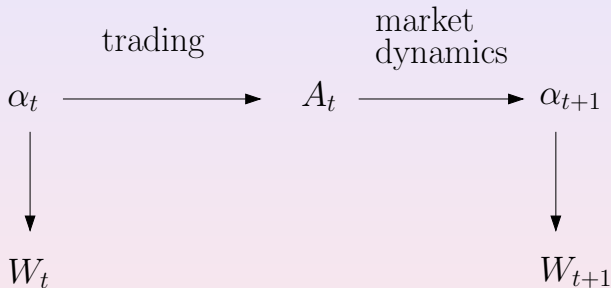
- 2-currency exchange rates:
 - dollar: $p_{12}(t)$
 - euro: $p_{21}(t)$
- Dynamics of dollar's exchange rate:

$$\frac{p_{12}(t+1)}{p_{12}(t)} = \rho_{t+1}$$

- Bid-ask spread:

$$p_{12}(t+1)p_{21}(t+1) = \eta_{t+1}^2 < 1$$

FX: Dynamics



$$\alpha_{t+1} = \frac{A_t \rho_{t+1}}{(1-A_t) + A_t \rho_{t+1}} \stackrel{\text{def}}{=} f_0(A_t, \rho_{t+1})$$

FX: Rewards

$$r_t = \log \frac{W_{t+1}}{W_t} = \log((1 - A_t) + A_t \rho_{t+1}) + \mathbb{I}(A_t \geq \alpha_t) \log \left(\frac{\alpha_t + \eta_{t+1}^2 (1 - \alpha_t)}{A_t + \eta_{t+1}^2 (1 - A_t)} \right)$$

.. if we buy euro: ultimately we will suffer some conversion loss

FX: Rewards

$$r_t = \log \frac{W_{t+1}}{W_t} = \log((1 - A_t) + A_t \rho_{t+1}) + \mathbb{I}(A_t \geq \alpha_t) \log \left(\frac{\alpha_t + \eta_{t+1}^2 (1 - \alpha_t)}{A_t + \eta_{t+1}^2 (1 - A_t)} \right)$$

.. if we buy euro: ultimately we will suffer some conversion loss

Markovian Dynamics

- $(\phi_t, \rho_t, \eta_t^2)$ – Markovian dynamics

- MDP:

- State: $X_t = (\phi_t, \rho_t, \eta_t^2, o_t)$
- Actions: $A = \{0, 1\}$
- Rewards: $\eta = r(\phi_t, a_t, \rho_{t-1}, \eta_{t-1}^2)$
- Time-evolution: $X_{t+1} = f(X_t, A_t, W_t)$, W_t “noise”

Markovian Dynamics

- $(\phi_t, \rho_t, \eta_t^2)$ – Markovian dynamics
- MDP:
 - State: $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$
 - Actions: $A = [0, 1]$
 - Rewards: $r_t = r(\alpha_t, a_t, \rho_{t+1}, \eta_{t+1}^2)$.
 - Time-evolution: $X_{t+1} = f(X_t, A_t, W_t)$, W_t “noise”

Markovian Dynamics

- $(\phi_t, \rho_t, \eta_t^2)$ – Markovian dynamics
- MDP:
 - State: $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$
 - Actions: $A = [0, 1]$
 - Rewards: $r_t = r(\alpha_t, a_t, \rho_{t+1}, \eta_{t+1}^2)$.
 - Time-evolution: $X_{t+1} = f(X_t, A_t, W_t)$, W_t “noise”

Markovian Dynamics

- $(\phi_t, \rho_t, \eta_t^2)$ – Markovian dynamics
- MDP:
 - State: $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$
 - Actions: $A = [0, 1]$
 - Rewards: $r_t = r(\alpha_t, a_t, \rho_{t+1}, \eta_{t+1}^2)$.
 - Time-evolution: $X_{t+1} = f(X_t, A_t, W_t)$, W_t “noise”

Markovian Dynamics

- $(\phi_t, \rho_t, \eta_t^2)$ – Markovian dynamics
- MDP:
 - State: $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$
 - Actions: $A = [0, 1]$
 - Rewards: $r_t = r(\alpha_t, \mathbf{a}_t, \rho_{t+1}, \eta_{t+1}^2)$.
 - Time-evolution: $X_{t+1} = f(X_t, A_t, W_t)$, W_t “noise”

Markovian Dynamics

- $(\phi_t, \rho_t, \eta_t^2)$ – Markovian dynamics
- MDP:
 - State: $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$
 - Actions: $A = [0, 1]$
 - Rewards: $r_t = r(\alpha_t, \mathbf{a}_t, \rho_{t+1}, \eta_{t+1}^2)$.
 - Time-evolution: $X_{t+1} = f(X_t, A_t, W_t)$, W_t “noise”

Outline

- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment**
 - FX Markets
 - Stock Market**
- 3 Solution Methods for MDPs
 - Classics
 - Approximate Methods
 - Does it Work?
- 4 Application to Log-optimal Investment
- 5 Conclusions

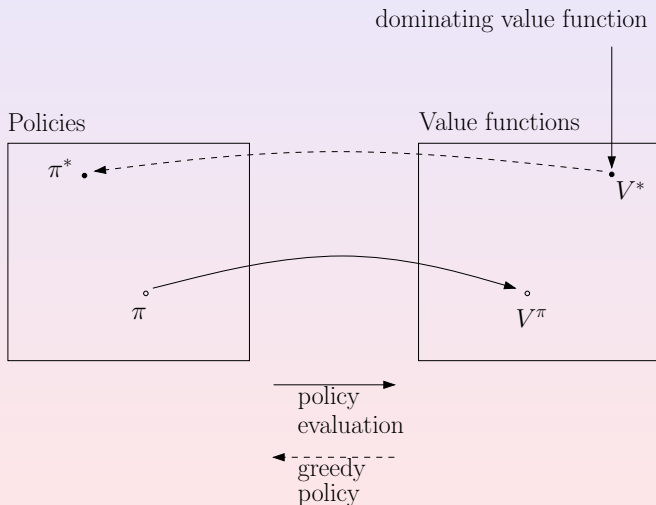
Stock Market

.. similar equations can be given:)

Outline

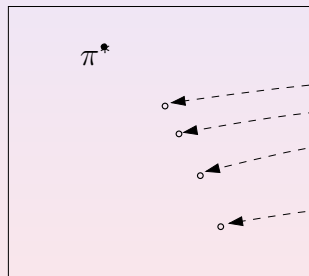
- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment
 - FX Markets
 - Stock Market
- 3 Solution Methods for MDPs**
 - Classics**
 - Approximate Methods
 - Does it Work?
- 4 Application to Log-optimal Investment
- 5 Conclusions

Big Picture

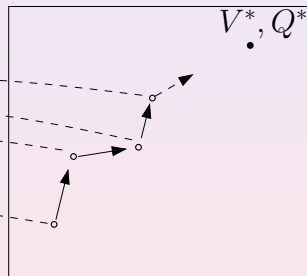


Value Iteration

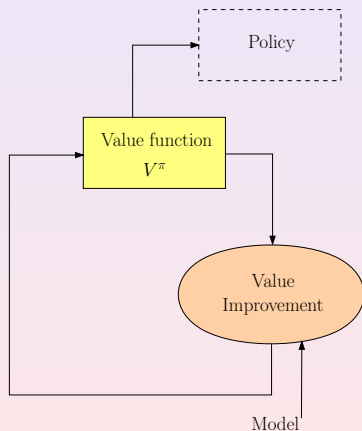
Policies



Value functions

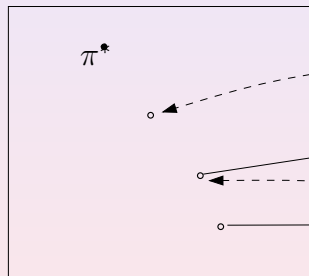


Value Iteration – Algorithmic View

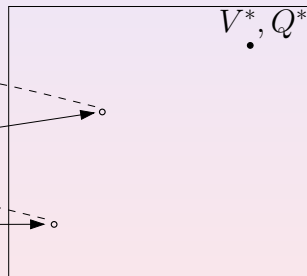


Policy Iteration

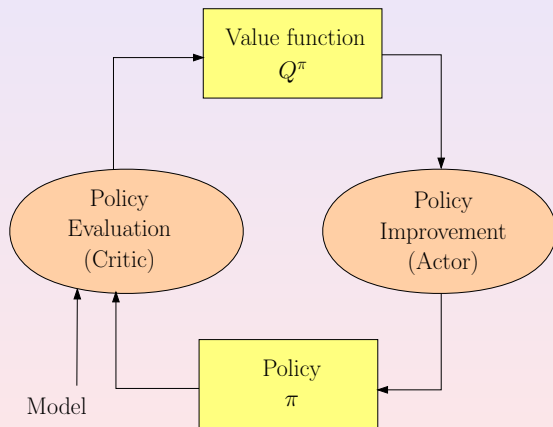
Policies



Value functions



Policy Iteration – Algorithmic View



Value- and Policy Iteration

Good

- Exact algorithms (asymptotically correct)
- Geometric convergence rate

Bad

What if model is unknown?

Value- and Policy Iteration

Good

- Exact algorithms (asymptotically correct)
- Geometric convergence rate

Bad

What if model is unknown?

Value- and Policy Iteration

Good

- Exact algorithms (asymptotically correct)
- Geometric convergence rate

Bad

- Requires model (analytic form)

What if model is unknown?

Value- and Policy Iteration

Good

- Exact algorithms (asymptotically correct)
- Geometric convergence rate

Bad

- Requires model (analytic form)
- Integration over state-space

What if model is unknown?

Value- and Policy Iteration

Good

- Exact algorithms (asymptotically correct)
- Geometric convergence rate

Bad

- Requires model (analytic form)
- Integration over state-space

What if model is unknown?

Value- and Policy Iteration

Good

- Exact algorithms (asymptotically correct)
- Geometric convergence rate

Bad

- Requires model (analytic form)
- Integration over state-space

What if model is unknown?

Value- and Policy Iteration

Good

- Exact algorithms (asymptotically correct)
- Geometric convergence rate

Bad

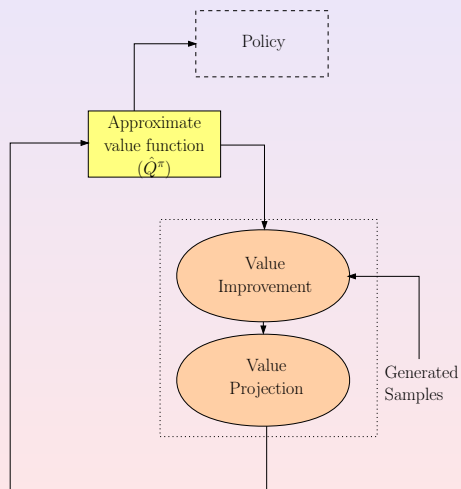
- Requires model (analytic form)
- Integration over state-space

What if model is unknown?

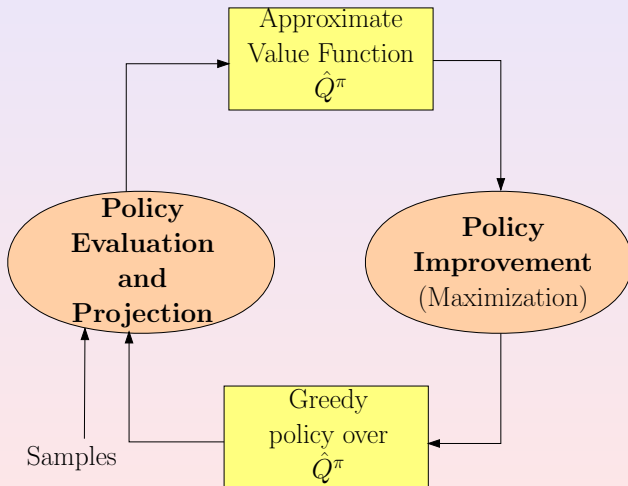
Outline

- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment
 - FX Markets
 - Stock Market
- 3 Solution Methods for MDPs**
 - Classics
 - Approximate Methods**
 - Does it Work?
- 4 Application to Log-optimal Investment
- 5 Conclusions

Fitted Value Iteration



Fitted Policy Iteration

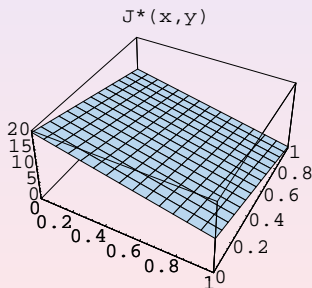
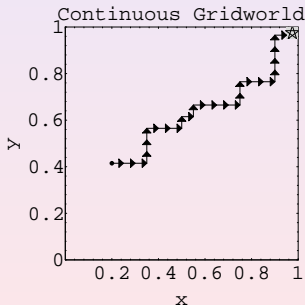


Outline

- 1 Introduction
 - Markovian Decision Problems
- 2 Log-optimal Investment
 - FX Markets
 - Stock Market
- 3 Solution Methods for MDPs**
 - Classics
 - Approximate Methods
 - Does it Work?**
- 4 Application to Log-optimal Investment
- 5 Conclusions

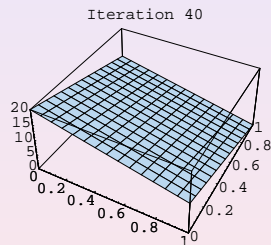
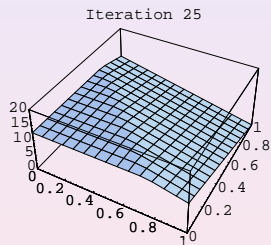
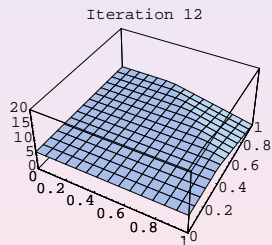
Fitted Value Iteration for Navigation Problems¹

From: Boyan & Moore: “Generalization in Reinforcement Learning: Safely Approximating the Value Function”, *NIPS-7*, 1995.



¹With thanks to Justin Boyan

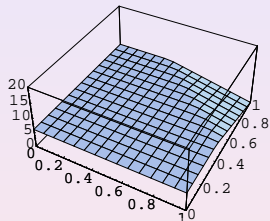
Navigation II.



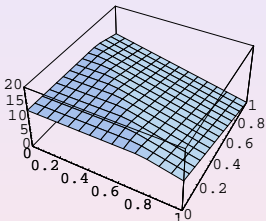
Value Iteration at Work

Navigation II.

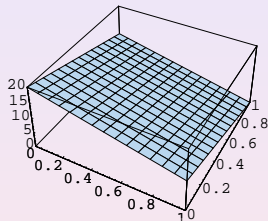
Iteration 12



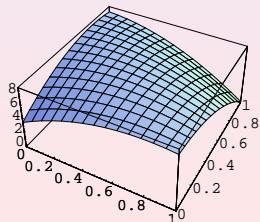
Iteration 25



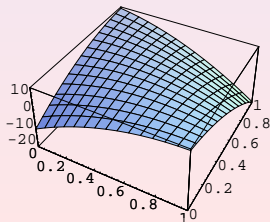
Iteration 40



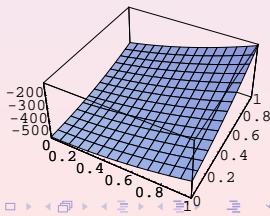
Iteration 17



Iteration 43



Iteration 127



Averagers – A Solution

$$V_{t+1} = \Pi_{\mathcal{F}} T V_t$$

- Requirement: $\Pi_{\mathcal{F}} T$ is sup-norm contraction
- Averagers (Gordon '95): Kernel averaging (fixed kernel), weighted k -nearest neighbors, Bézier patches, linear interpolation on a triangular (or tetrahedral, etc.) mesh, bilinear interpolation on a square (or cubical, etc.), ...

Averagers – A Solution

$$V_{t+1} = \Pi_{\mathcal{F}} T V_t$$

- Requirement: $\Pi_{\mathcal{F}} T$ is sup-norm contraction
- Averagers (Gordon '95): Kernel averaging (fixed kernel), weighted k -nearest neighbors, Bézier patches, linear interpolation on a triangular (or tetrahedral, etc.) mesh, bilinear interpolation on a square (or cubical, etc.), ...

Pushing the Edge – a Finite-Time Bound

Theorem²: Assume MDP is regular. Fix $\delta > 0$, $\epsilon > 0$, \mathcal{F} , ρ , μ . Assume that \mathcal{V} , the “capacity” of \mathcal{F} is finite. Assume that Bellman-errors for functions in \mathcal{F} can be uniformly bounded:

$$\sup_{g \in \mathcal{F}} \inf_{f \in \mathcal{F}} \|f - Tg\|_{p, \mu} \leq \epsilon.$$

Then, it is possible to select N, M, K such that after K iterations of the sampling based FVI algorithm run with (μ, N, M)

$$\|V^* - V^{\pi_K}\|_{p, \rho} \leq \frac{4C^{1/p}}{(1 - \gamma)^2} \epsilon$$

with probability at least $1 - \delta$. Further, N, M, K are polynomial in \mathcal{V} , R_{\max} , $1/\epsilon$, $\log |\mathcal{A}|$, $\log(1/\delta)$, $1/(1 - \gamma)$.

Here C is a constant related to how quickly future state distributions can **concentrate** away from ρ relative to μ .

²Munos & Szepesvári, ICML-2005

Extension to Fitted Policy Iteration

- Previous result required generative model
- Single sample path?

YES!

Extension to Fitted Policy Iteration

- Previous result required generative model
- Single sample path?

YES!

Extension to Fitted Policy Iteration

- Previous result required generative model
- Single sample path?

YES!

Log-optimal Investment – FX

- Fitted Value Iteration (with generative model):

⇒ +++

- Fitted Policy Iteration (single sample path):

⇒ - - -

- Trick:

- $X_t = (s_t, w_t, \pi_t^*, \alpha_t)$
- s_t, w_t, π_t^* – market state: external
- α_t – portfolio state: internal
- Systematic sampling of the portfolio-state

⇒ +++

Log-optimal Investment – FX

- Fitted Value Iteration (with generative model):

⇒ +++

- Fitted Policy Iteration (single sample path):

⇒ - - -

- Trick:

- $X_t = (s_t, w_t, \pi_t^T, \alpha_t)$
- s_t, w_t, α_t – market state: external
- π_t – portfolio state: internal
- Systematic sampling of the portfolio-state

⇒ +++

Log-optimal Investment – FX

- Fitted Value Iteration (with generative model):

⇒ +++

- Fitted Policy Iteration (single sample path):

⇒ - - -

- Trick:

- $X_t = (\phi_t, p_t, \eta_t^2, \alpha_t)$
- ϕ_t, p_t, η_t^2 – market state: external
- α_t – portfolio state: internal
- Systematic sampling of the portfolio-state

⇒ +++

Log-optimal Investment – FX

- Fitted Value Iteration (with generative model):

⇒ +++

- Fitted Policy Iteration (single sample path):

⇒ - - -

- Trick:

- $X_t = (\phi_t, \mu_t, \eta_t^2, \alpha_t)$
- ϕ_t, μ_t, η_t^2 – market state: external
- α_t – portfolio state: internal
- Systematic sampling of the portfolio-state

⇒ +++

Log-optimal Investment – FX

- Fitted Value Iteration (with generative model):

⇒ +++

- Fitted Policy Iteration (single sample path):

⇒ - - -

- Trick:

- $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$
- ϕ_t, ρ_t, η_t^2 – market state: **external**
- α_t – portfolio state: **internal**
- Systematic sampling of the portfolio-state

⇒ +++

Log-optimal Investment – FX

- Fitted Value Iteration (with generative model):

⇒ +++

- Fitted Policy Iteration (single sample path):

⇒ - - -

- Trick:

- $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$

- ϕ_t, ρ_t, η_t^2 – market state: **external**

- α_t – portfolio state: **internal**

- Systematic sampling of the portfolio-state

⇒ +++

Log-optimal Investment – FX

- Fitted Value Iteration (with generative model):

⇒ +++

- Fitted Policy Iteration (single sample path):

⇒ - - -

- Trick:

- $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$
- ϕ_t, ρ_t, η_t^2 – market state: **external**
- α_t – portfolio state: **internal**
- Systematic sampling of the portfolio-state

⇒ +++

Log-optimal Investment – FX

- Fitted Value Iteration (with generative model):

⇒ +++

- Fitted Policy Iteration (single sample path):

⇒ - - -

- Trick:

- $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$
- ϕ_t, ρ_t, η_t^2 – market state: **external**
- α_t – portfolio state: **internal**
- Systematic sampling of the portfolio-state

⇒ +++

Log-optimal Investment – FX

- Fitted Value Iteration (with generative model):

⇒ +++

- Fitted Policy Iteration (single sample path):

⇒ - - -

- Trick:

- $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$
- ϕ_t, ρ_t, η_t^2 – market state: **external**
- α_t – portfolio state: **internal**
- Systematic sampling of the portfolio-state

⇒ +++

Log-optimal Investment – FX

- Fitted Value Iteration (with generative model):

⇒ +++

- Fitted Policy Iteration (single sample path):

⇒ - - -

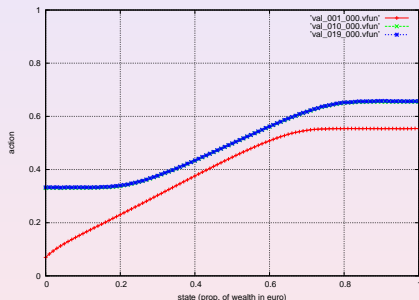
- Trick:

- $X_t = (\phi_t, \rho_t, \eta_t^2, \alpha_t)$
- ϕ_t, ρ_t, η_t^2 – market state: **external**
- α_t – portfolio state: **internal**
- Systematic sampling of the portfolio-state

⇒ +++

Results

Kernel-regression, $\phi_t = \emptyset$, $N = 100$ samples



- Final yield: 0.0014
- Yield of CBAL(0.5): 0.00076

Conclusions

- **MDPs – not only in finite spaces**
- Fitted Value/Policy Iteration
- Generative Model: OK
- Single-sample Path: Requires care
- Good: No “state”, just good enough features
- Alternatives: Gradient Methods³

³Gerencsér et al.: Log-optimal Currency Portfolios and Control Lyapunov Exponent

Conclusions

- MDPs – not only in finite spaces
- **Fitted Value/Policy Iteration**
- Generative Model: OK
- Single-sample Path: Requires care
- Good: No “state”, just good enough features
- Alternatives: Gradient Methods³

³Gerencsér et al.: Log-optimal Currency Portfolios and Control Lyapunov Exponent

Conclusions

- MDPs – not only in finite spaces
- Fitted Value/Policy Iteration
- **Generative Model: OK**
- Single-sample Path: Requires care
- Good: No “state”, just good enough features
- Alternatives: Gradient Methods³

³Gerencsér et al.: Log-optimal Currency Portfolios and Control Lyapunov Exponent

Conclusions

- MDPs – not only in finite spaces
- Fitted Value/Policy Iteration
- Generative Model: OK
- **Single-sample Path: Requires care**
- Good: No “state”, just good enough features
- Alternatives: Gradient Methods³

³Gerencsér et al.: Log-optimal Currency Portfolios and Control Lyapunov Exponent

Conclusions

- MDPs – not only in finite spaces
- Fitted Value/Policy Iteration
- Generative Model: OK
- Single-sample Path: Requires care
- **Good: No “state”, just good enough features**
- Alternatives: Gradient Methods³

³Gerencsér et al.: Log-optimal Currency Portfolios and Control Lyapunov Exponent

Conclusions

- MDPs – not only in finite spaces
- Fitted Value/Policy Iteration
- Generative Model: OK
- Single-sample Path: Requires care
- Good: No “state”, just good enough features
- **Alternatives: Gradient Methods³**

³Gerencsér et al.: Log-optimal Currency Portfolios and Control Lyapunov Exponent

Questions?

???