Exploiting Symmetry in High-Dimensional Dynamic Programming

Kahou at al, presented by Yao Luo

Boston College

April 6, 2024

Introduction

Introduction

Model: Investment under uncertainty

Introduction

Model solution: How to solve the curse of dimensionality problem?

Permutation-Invariant Dynamic Programming

- Dimensionality reduction
- Definition 1+2+intuition
- Representation theorem

Permutation-Invariant Dynamic Programming

- Specific functional representation in the investment model

Concentration of Measure

- Why?
- definition 4 + proposition 3 + intuition

Concentration of Measure

- When it holds?
- section 5

Deep Learning Implementation

- networks architecture set up:3
- baseline case
- Training: Euler residuals + minimization

Deep Learning Implementation

- Algorithm

Deep Learning Implementation

Case I: $\nu = 1$

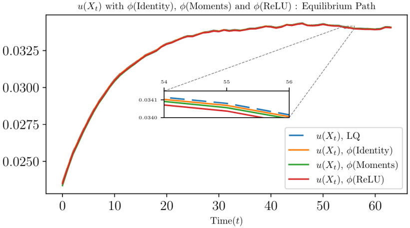

Figure 3: Comparison between the LQ-regulator solution and our three deep learning archictures for the case with $\nu = 1$ and $N = 128$.

Deep Learning Implementation

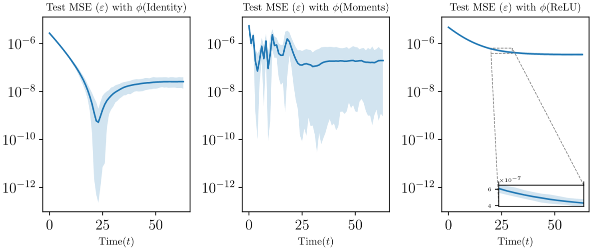

Figure 4: The Euler residuals for $\nu = 1$ and $N = 128$ for ϕ(Identity), ϕ(Moments), and ϕ(ReLU). The dark blue curve shows the average residuals along equilibrium paths for 256 different trajectories. The shaded areas depict the 2.5th and 97.5th percentiles.

Deep Learning Implementation

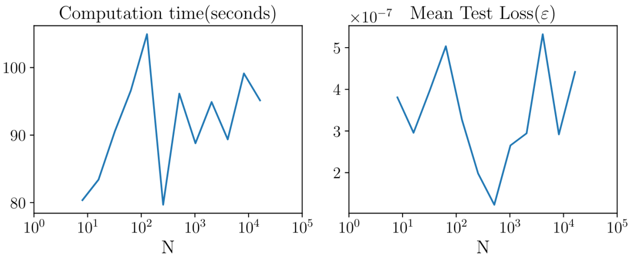

Figure 5: Performance of the ϕ(ReLU) for different N.

Deep Learning Implementation

Table 1: Performance of different networks in solving case I: $\nu = 1$

group	description	Time (s)	Params (K)	Train MSE (ε)	Test MSE (ε)	Val MSE (ε)	Policy Error ($\lvert u - u_{\text{ref}} \rvert$)	Policy Error $\left(\frac{\lvert u - u_{\text{ref}} \rvert}{u_{\text{ref}}}\right)$
ϕ(Identity)	Baseline	42	49.4	4.1e-06	3.3e-07	3.3e-07	2.9e-05	0.10%
	Thin (64 nodes)	33	12.4	3.7e-06	2.7e-07	2.7e-07	3.4e-05	0.10%
ϕ(Moments)	Baseline	55	49.8	1.4e-06	7.6e-07	7.6e-07	2.8e-05	0.09%
	Moments (1,2)	211	49.5	2.4e-06	1.1e-06	2.3e-06	4.4e-05	0.14%
	Very Shallow(1 layer)	241	0.6	1.1e-05	8.4e-06	7.9e-06	1.1e-02	34.00%
	Thin (64 nodes)	82	12.6	1.6e-06	9.1e-07	9.2e-07	3.8e-05	0.12%
ϕ(ReLU)	Baseline	107	66.8	3.7e-06	3.3e-07	3.3e-07	2.7e-05	0.09%
	L = 2	86	66.3	1.3e-05	2.1e-07	2.2e-07	2.6e-05	0.08%
	L = 16	91	69.9	5.5e-06	1.5e-07	1.5e-07	2.1e-05	0.07%
	Shallow(ϕ : 1 layer, ρ : 2 layers)	79	17.7	2.0e-06	5.5e-07	5.5e-07	3.2e-05	0.11%
	Deep(ϕ : 4 layers, ρ : 8 layers)	242	165.1	2.1e-03	2.2e-03	2.1e-03	2.7e-03	8.50%
	Thin(ϕ, ρ : 64 nodes)	87	17.0	1.1e-05	4.5e-07	4.5e-07	3.0e-05	0.10%

Deep Learning Implementation

Case II:

Extensions

The tools are useful for solving any high-dimensional functional equations with some degree of symmetry, especially when these equations contain high-dimensional expectations.

- Decreasing returns to scale
- Multiple productivity types
- Complex idiosyncratic states
- Global solutions with transitions and aggregate shocks

Discussion

- Solve high-dimensional dynamic programming problems in minutes.
- Double-descent
- Model selection since results are sensitive to different network architectures.