Macroeconomic Effects of Production Networks

A Deep Learning Approach

Jesús Fernández-Villaverde,[1] Christian Matthes,[2] Hikaru Saijo,[3] Felipe Schwartzman[4]

July 16, 2026

[1]University of Pennsylvania

[2]University of Notre Dame

[3]University of California Santa Cruz

[4]Federal Reserve Bank of Richmond

The Great Recession: a puzzle in the recovery

- The Great Recession was the deepest downturn since the 1930s.
- Its depth is not the puzzle, but the slow recovery.
- Output and investment crashed in 2008–2009, then stayed below trend for nearly a decade.
- Textbook logic says they should have rebounded: once the shocks fade, firms rebuild depleted capital.
- So why didn't they?
- Our answer: global nonlinearities in production and investment networks that standard methods approximate away.

Production networks and nonlinearities

- Networks turn sector-level shocks into aggregate fluctuations:
  - Long and Plosser (1983), Horvath (2000), Foerster et al. (2011), Acemoglu et al. (2012), Baqaee and Farhi (2019).
- Recent disruptions were large:
  1. Big shocks (e.g., COVID) pushed the economy quite far away from the steady state.
  2. Sectoral variables (e.g., trade wars) can swing hard even when aggregates look tame.
- Yet most network macro still use local approximations around the steady state.

Where do the nonlinearity comes from?

- The exact function for log-capital next period is log-concave:

$$\widehat{k}_{t+1} = f(\widehat{k}_t, z_t)$$

- The first-order approximation is linear:

$$\widehat{k}_{t+1} = \theta_1 \widehat{k}_t + \theta_2 z_t$$

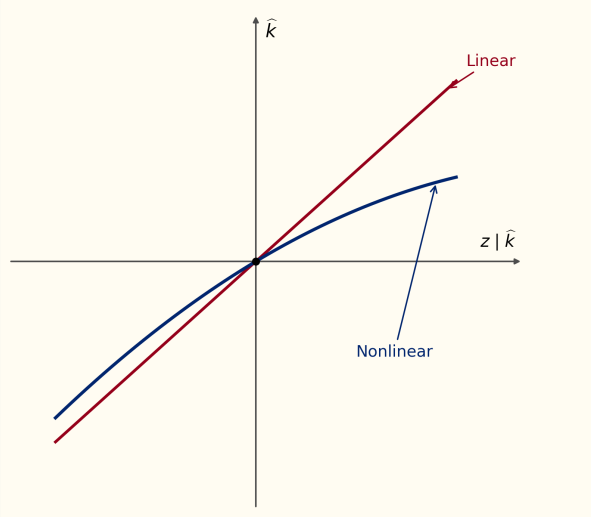
- Thus, a first-order approximation artificially amplifies ordinary fluctuations and implies too fast a recovery from large shocks.

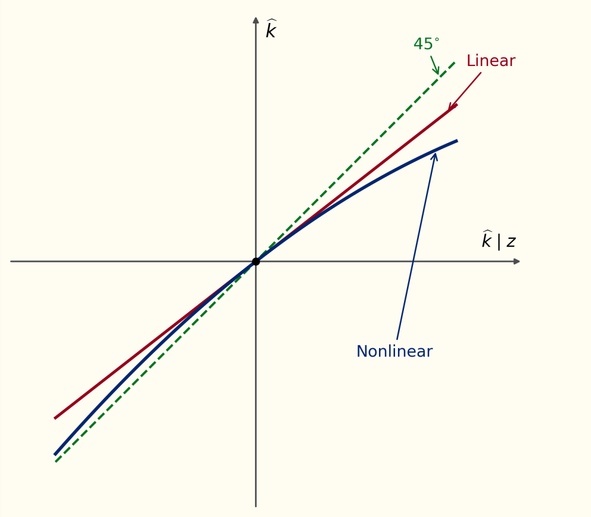
- Higher-order perturbations reduce this problem, but not enough: singularity.

- This mechanism also exists in the model with one sector.

- But the network structure of the economy makes it much stronger. Why?

Exact vs. first-order policy functions





Investment hubs

- A few sectors (construction, machinery, motor vehicles) produce most of the economy's investment goods (vom Lehn and Winberry, 2022).
- With professional & technical services, these investment hubs account for nearly 70% of all investment.
- Hit the hubs with a large shock, and nonlinear resource constraints create bottlenecks: resources cannot flow quickly into investment goods.
- In particular, existing capital is predetermined: the marginal productivity of additional resources in investment hubs will be low.

But how can we do this?

- The curse of dimensionality is a serious roadblock: we calibrate to 37 sectors.
- We build a scalable deep learning framework that solves these high-dimensional network economies globally.
- Core idea: exploit the symmetry (exchangeability) of the environment to beat the curse of dimensionality.
- We benchmark against first-order and third-order perturbation.

The main results

- An (apparent) paradox: the same nonlinearities make ordinary fluctuations smaller, yet make the recovery from a large shock slower.
- They reshuffle the sectoral origins of fluctuations: construction, machinery, and motor vehicles matter *less* than perturbation implies; professional & technical services becomes the main driver.
- We trace perturbation failure to singularities in the capital Euler equation, and build a network statistic that predicts it.
- All under standard Cobb-Douglas technology and log preferences.

Model

Preferences and aggregation

- Stand-in household with utility function:

$$U = \mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t \left[\frac{C_t^{1-\sigma}}{1-\sigma} - \frac{1}{1+\eta} \left(\sum_i h_{i,t} \right)^{1+\eta} \right]$$

  (σ risk aversion, η inverse Frisch elasticity, β discount factor).

- Consumption aggregation:

$$C_t = \prod_{i=1}^{n} c_{i,t}^{\omega_i}, \qquad \sum_{i=1}^{n} \omega_i = 1$$

  where $c_{i,t}$ is consumption of the sector-i good and ω_i are preference shares.

- Common labor market; labor earns a single real wage w_t.

Network structure

- Production function:

$$y_{i,t} = e^{z_{i,t}} k_{i,t}^{\alpha_i^k} h_{i,t}^{\alpha_i^h} \prod_{j=1}^{n} m_{ij,t}^{a_{ij}}$$

  with constant returns: $\alpha_i^k + \alpha_i^h + \sum_{j=1}^{n} a_{ij} = 1$.

- $z_{i,t}$ are sectoral TFP shocks that follow an AR(1) process,

$$z_{i,t} = \rho_i^z z_{i,t-1} + \varepsilon_{i,t}^z$$

  with $\varepsilon_t^z \sim \mathcal{N}(0, \Sigma^z)$.

- a_{ij} are intermediate-input shares (input-output matrix A).

Capital, investment, and resource constraints

- Law of motion for capital:

$$k_{i,t+1} = (1 - \delta_i)\, k_{i,t} + x_{i,t}$$

- Investment produced by combining sectoral goods (investment network B):

$$x_{i,t} = \prod_{j=1}^{n} q_{ij,t}^{b_{ij}}, \qquad \sum_{j=1}^{n} b_{ij} = 1$$

- Resource constraint:

$$y_{i,t} = c_{i,t} + \sum_{j=1}^{n} m_{ji,t} + \sum_{j=1}^{n} q_{ji,t}$$

The price of capital

- Since new capital in sector i is a Cobb-Douglas bundle of sectoral goods, cost minimization gives its price:

$$p^I_{i,t} = \prod_{j=1}^{n} \left(\frac{p_{j,t}}{b_{ij}} \right)^{b_{ij}}$$

- Optimal capital choice equates this price to its discounted return (Euler equation):

$$p^I_{i,t} = \beta \, \mathbb{E}_t \left[\left(\frac{C_{t+1}}{C_t} \right)^{-\sigma} \left(\underbrace{p_{i,t+1} \, \alpha^k_i \, \frac{y_{i,t+1}}{k_{i,t+1}}}_{\text{marginal product}} + \underbrace{(1 - \delta_i) \, p^I_{i,t+1}}_{\text{resale value}} \right) \right]$$

Solution

Why solving this model is hard

- Individual sector state: $s_i = [\, k_i,\ z_i \,]$ (capital, productivity).

- Aggregate state stacks all sectors into an ordered $n \times 2$ matrix:

$$S = [\, s_1,\ s_2,\ \ldots,\ s_n \,]'$$

- Firm i's decisions depend on the *entire* price vector $\boldsymbol{p}$ and the discount factor $C^{-\sigma}$; the networks connect all sectors, so S is the minimal sufficient statistic.

- Curse of dimensionality: grids are infeasible as n grows, and an ordered S forces tracking how i responds to j vs. k (pairwise interactions explode).

- We want to calibrate to $n = 37 \Rightarrow$ a 74-dimensional state.

- Need a representation that scales with n while preserving general-equilibrium discipline.

Exploiting symmetry

- Assume (just for this slide and the next) that all sectors are symmetric.
- What matters is the distribution of the other sectors' states, not their labels.
  - Swap sectors j and k's states $\Rightarrow$ same prices, same aggregates.
  - This is permutation invariance (exchangeability).
- Representation theorem (Zaheer et al., 2017; Ebrahimi Kahou et al., 2021): any permutation-invariant f can be written as

$$f(s_i, S) \approx \rho\Big(s_i,\ \tfrac{1}{n}\textstyle\sum_{j=1}^{n} \phi(s_j)\Big)$$

- Krusell–Smith is the special case $\phi(s_i) = s_i$ (track only the mean); we let the data choose ϕ.

Deep Sets architecture

Three parts of the representation, each a neural network:

$$f(s_i, S) \approx \underbrace{\rho}_{\text{decoder}} \Bigg(s_i,\ \underbrace{\frac{1}{n}\sum_{j=1}^{n} \underbrace{\phi(s_j)}_{\text{encoder}}}_{\text{pooling}} \Bigg)$$

- Encoder ϕ: maps each sector state s_j into a latent vector.
- Pooling: permutation-invariant average, a sufficient statistic for the distribution of states.
- Decoder ρ: combines own state s_i with the pooled vector to produce the output.
- We approximate ϕ and ρ with deep neural nets:
  - # parameters independent of $n \Rightarrow$ scalability.
  - Concentration of measure: the pooled mean concentrates sharply as n grows.

Symmetry with heterogeneous sectors

- But sectors are *not* symmetric:
  - Factor shares differ.
  - Position in the input-output and investment networks differs.
- But symmetry only requires that two sectors with the same state vector behave identically.
- So we fold all sector-specific parameters into the state s_i:

$$s_i \equiv \Big[\, k_i,\ z_i,\ \underbrace{\omega_i, \alpha_i^k, \alpha_i^h, \delta_i, \rho_i^z, \sigma_i^z}_{\text{sector parameters}},\ \underbrace{\zeta(a_{i:}, a_{:i}, b_{i:}, b_{:i})}_{\text{network position}} \,\Big]'$$

- Feature extractor $\zeta : \mathbb{R}^{4n} \to \mathbb{R}^{L_\zeta}$ compresses sector i's $4n$ links into $L_\zeta < 4n$:
  - Inputs: the i-th row and column of the input-output matrix A and investment matrix B.
  - Keeps $\dim(s_i)$ independent of n $\Rightarrow$ preserves scalability.

Deep learning implementation, I

- Three networks to train: encoder ϕ, decoder ρ, feature extractor ζ.
- Approximate sectoral consumption ($\widehat{c}$), investment ($\widehat{x}$), and value functions ($\widehat{V}$)

$$\begin{bmatrix} \hat{c}\left(s_i, S\right) \\ \hat{x}\left(s_i, S\right) \\ \hat{V}\left(s_i, S\right) \end{bmatrix} \approx \rho\left(s_i, \frac{1}{n}\sum_{j=1}^{n}\phi(s_j); \theta\right),$$

  where hat denotes log-deviations (deviations for V) from the non-stochastic steady state.

- Remaining allocations (p_i, h_i, m_{ij}, q_{ij}, ...) recovered *exactly* from equilibrium conditions.
- All sectoral information is encoded as arguments in functions, so we do not need a separate function for each sector! (no i in ρ or ϕ)

Deep learning implementation, II

- Train the networks by enforcing the equilibrium conditions on simulated state draws (residual minimization).
- Loss function over M simulated draws, averaged across sectors n:

$$\mathcal{L}(\theta) = \frac{1}{Mn} \sum_{m=1}^{M} \left[\sum_{i=1}^{n} \nu \varepsilon_{i,m}^{rewards} + \sum_{i=1}^{n} \left(\varepsilon_{i,m}^{value} \right)^2 + \sum_{i=1}^{n} \left(\varepsilon_{i,m}^{resource} \right)^2 + \sum_{i=1}^{n} \left(\varepsilon_{i,m}^{Euler} \right)^2 \right]$$

- Minimizing $\varepsilon_{i,m}^{rewards}$ enforces firm value maximization; the other residuals enforce the value definition, resource constraints, and the Euler equation.

Errors to be minimized

Residuals entering $\mathcal{L}(\theta)$ (tildes = network approximations; $\widetilde{\mu}_i$ is the approximated multiplier):

$$\varepsilon_i^{rewards} = -\left(\widetilde{C}^{-\sigma}\left[\widetilde{p}_i\widetilde{y}_i - \sum_{j=1}^{n}\widetilde{p}_i\widetilde{m}_{ij} - \sum_{j=1}^{n}\widetilde{p}_j\widetilde{q}_{ij} - \widetilde{w}\widetilde{h}_i\right] + \beta\mathbb{E}\left[\widetilde{V}\left(s_i', S'\right)\right]\right)$$

$$\varepsilon_i^{value} = 1 - \frac{\widetilde{C}^{-\sigma}\left[\widetilde{p}_i\widetilde{y}_i - \sum_{j=1}^{n}\widetilde{p}_i\widetilde{m}_{ij} - \sum_{j=1}^{n}\widetilde{p}_j\widetilde{q}_{ij} - \widetilde{w}\widetilde{h}_i\right] + \beta\mathbb{E}\left[\widetilde{V}\left(s_i', S'\right)\right]}{\widetilde{V}\left(s_i, S\right)}$$

$$\varepsilon_i^{resource} = 1 - \frac{\widetilde{c}_i + \sum_{j=1}^{n}\widetilde{m}_{ji} + \sum_{j=1}^{n}\widetilde{q}_{ji}}{\widetilde{y}_i}$$

$$\varepsilon_i^{Euler} = 1 - \frac{\beta\mathbb{E}\left\{\widetilde{C}'^{-\sigma}\widetilde{p}_i'\alpha_i^k\frac{\widetilde{y}_i'}{\widetilde{k}_i} + \widetilde{\mu}_i'(1-\delta_i)\right\}}{\widetilde{\mu}_i}$$

The training loop at a glance

1. Pre-train: anchor ϕ, ρ, ζ to the first-order solution.
2. Simulate: draw trajectories from the current policies (stay on the ergodic set).
3. Minimize: take AdamW steps on $\mathcal{L}(\theta)$ over mini-batches.
4. Repeat 2–3, regenerating data every 20 epochs, until convergence.

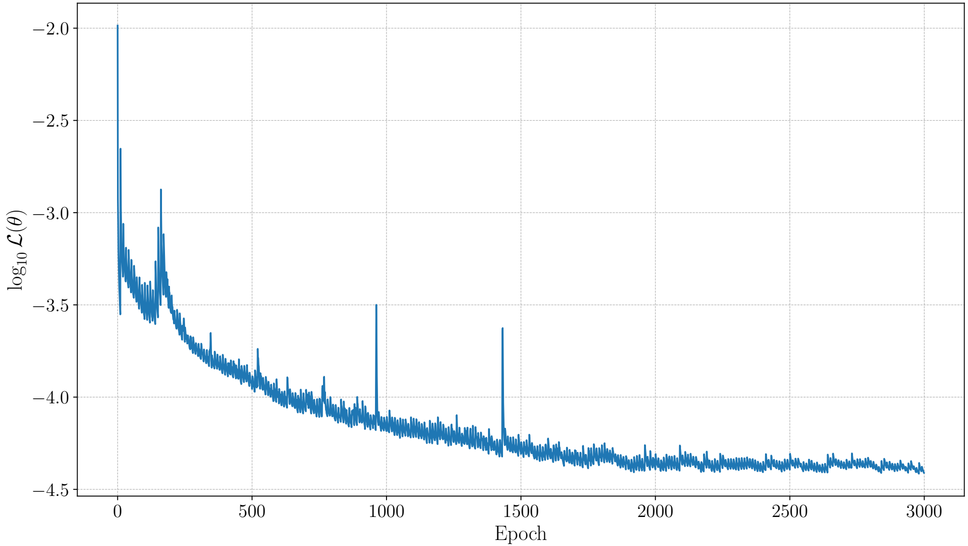
Output: functions ϕ, ρ, ζ that solve the model for *any* sector, with a parameter count independent of n.

Quantitative results

Calibration

- Model calibrated to the U.S. economy at annual frequency.
- $n = 37$ sectors.
- $\beta = 0.96$; $\sigma = 1$ (log utility); $\eta = 0$ (indivisible labor, Hansen, 1985).
- Networks A, B, preference/factor shares, depreciation rates $\{\delta_i\}$, and TFP process $(\{\rho_i^z\}, \Sigma^z)$: we follow as much as possible vom Lehn and Winberry (2022).

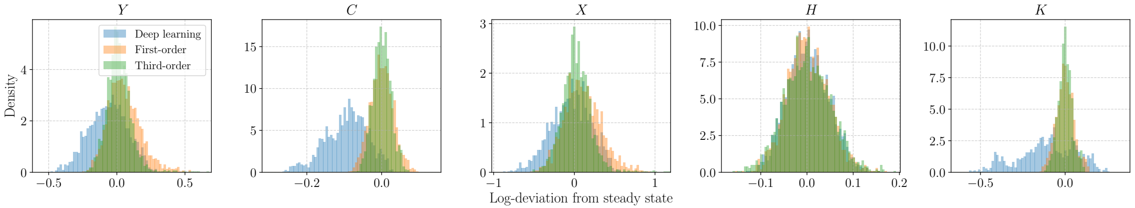
Training diagnostics



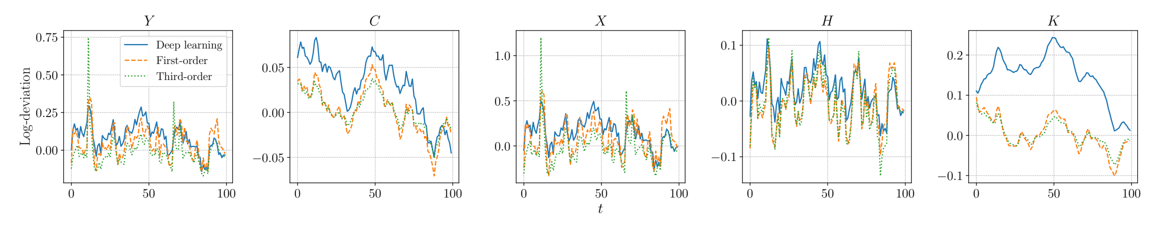
Test errors: deep learning vs. perturbation

	MAE	MSE	Max	P10	P50	P90
Panel A: Deep learning solution						
Euler equation	9.7e-3	2.0e-4	7.2e-2	1.0e-3	6.1e-3	2.4e-2
Resource constraint	9.0e-3	1.8e-4	1.1e-1	1.1e-3	6.0e-3	2.0e-2
Panel B: First-order solution						
Euler equation	1.4e-2	7.8e-4	5.3e-1	1.3e-3	7.1e-3	3.2e-2
Resource constraint	5.9e+2	1.5e+8	8.7e+5	2.2e-4	1.1e-2	3.3e+1
Panel C: Third-order solution						
Euler equation	1.2e-2	1.4e-3	1.1e+0	5.2e-4	3.9e-3	2.2e-2
Resource constraint	inf	inf	inf	9.4e-3	1.2e-1	inf

Ergodic distributions



Dynamics

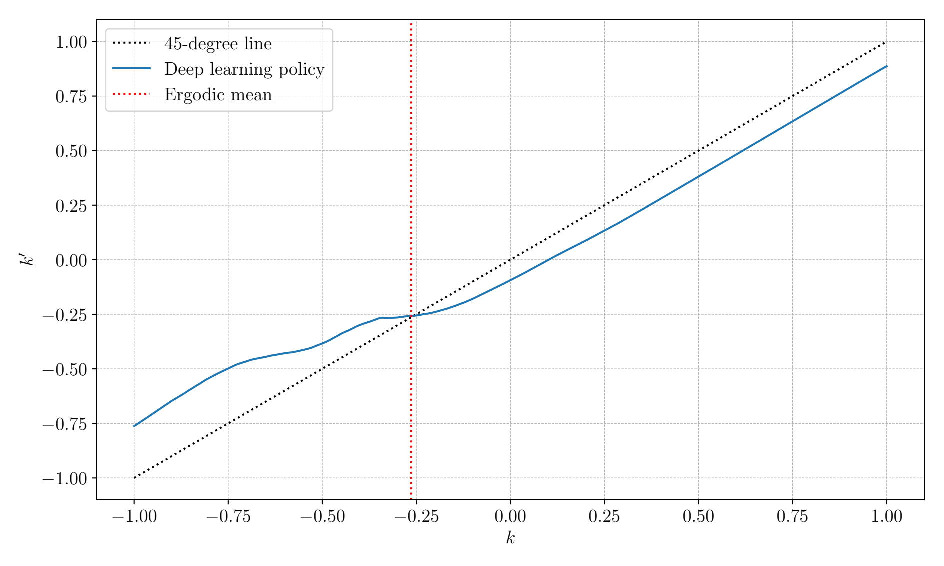


Y
C
X
H
K
Log-deviation
Deep learning
First-order
Third-order
0.75
0.50
0.25
0.00
0.05
0.00
−0.05
1.0
0.5
0.0
0.1
0.0
−0.1
0.2
0.1
0.0
−0.1
0
50
100
t

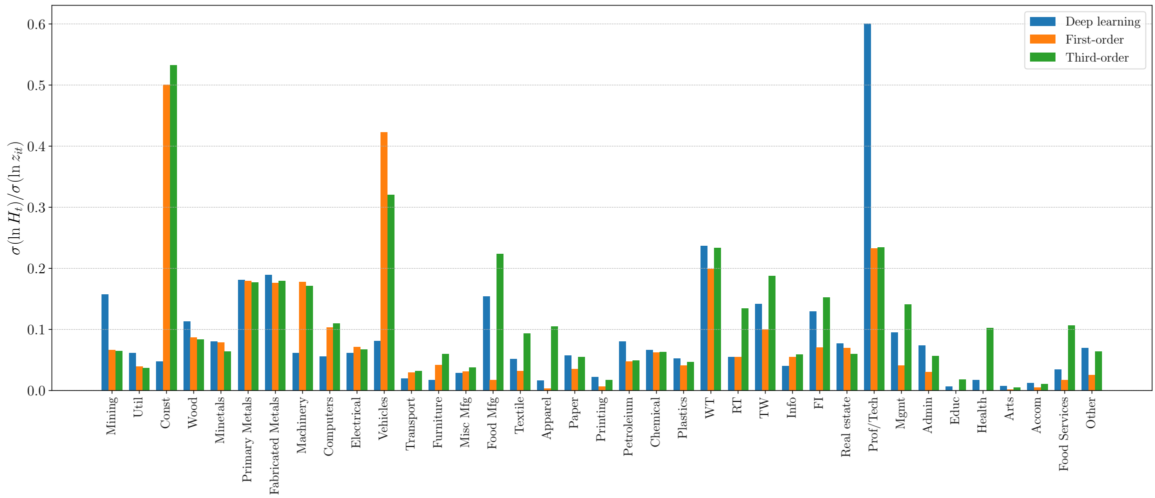
Ergodic moments of aggregate variables

	Y	C	X	H	K
Panel A: Ergodic means					
Deep learning	-0.06	-0.09	-0.04	0.00	-0.11
First-order	0	0	0	0	0
Third-order	0.02	0.00	0.04	0.00	0.00
Panel B: Standard deviations					
Deep learning	5.6	0.8	11.2	2.7	1.0
First-order	9.5	0.9	18.4	3.3	1.1
Third-order	23.7	0.6	31.6	3.8	0.8
Data	2.2	1.8	9.9	2.7	0.8

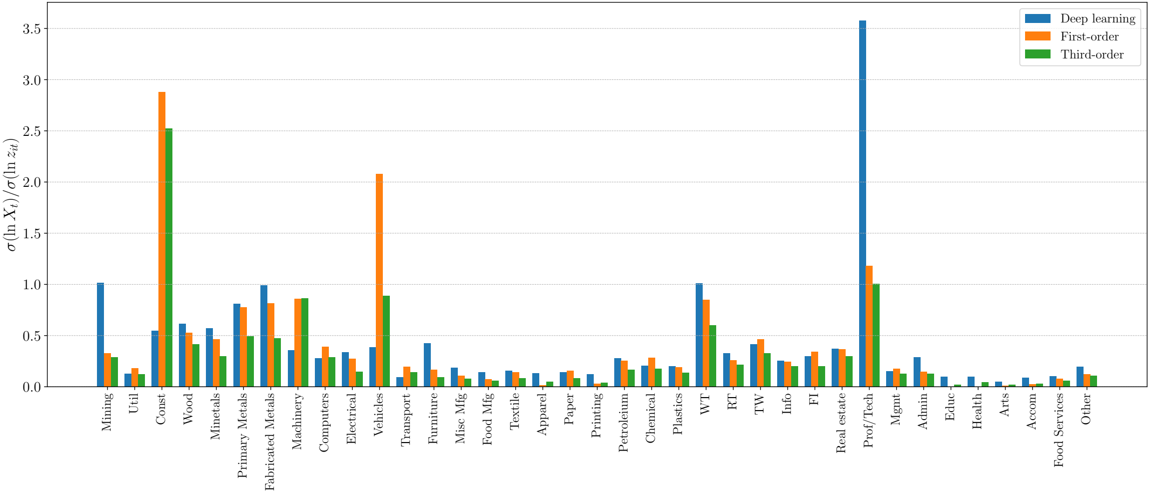
Policy function for the construction sector



Elasticity of aggregate labor to sectoral TFP shocks



Elasticity of aggregate investment to sectoral TFP shocks



Reallocation of sectoral importance

- Under both perturbation methods, construction and vehicles dominate aggregate labor and investment (as in vom Lehn and Winberry, 2022).
- The deep learning solution drastically dampens the elasticities of these investment hubs.
- Instead, professional & technical services emerges as the most important sector: investment elasticity roughly $3\times$ that of the perturbation solutions.
- The biggest gap between solutions lies precisely in the investment hubs (nearly 70% of total investment).

The Great Recession

The Great Recession and the slow recovery

- Solution methods disagree most on investment hubs $\Rightarrow$ predictions diverge during episodes with large investment declines.
- Experiment: start in 2007 (DL and third-order at ergodic means, first-order at steady state).
- Feed the estimated 2008–2018 sectoral TFP residuals of vom Lehn and Winberry (2022) into the model.
- Compare model paths to detrended data. Aggregate TFP drops sharply in 2008–2009 and never fully recovers.

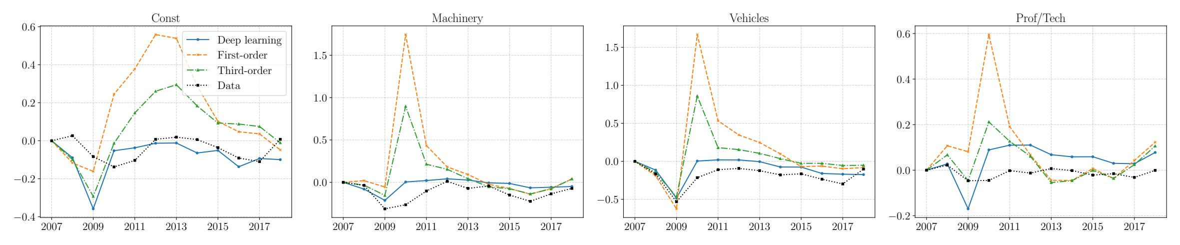
The Great Recession: model vs. data

Model vs. data: mean squared error

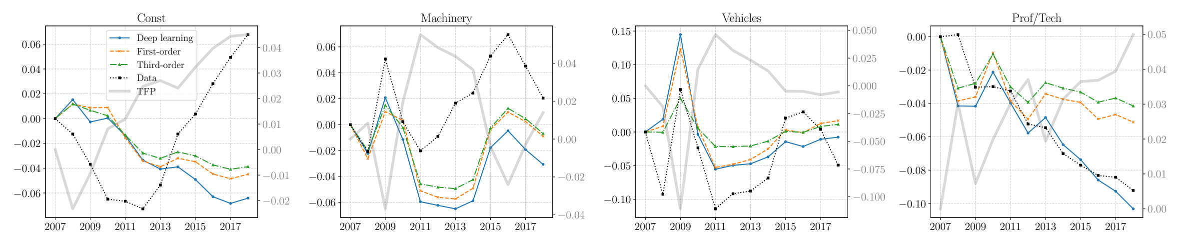
	Y	C	X	H	K	Average
Deep learning	6.3e-4	2.1e-4	5.1e-3	1.6e-3	4.5e-4	1.6e-3
First-order	2.5e-2	1.6e-4	1.4e-1	3.9e-3	1.7e-4	3.3e-2
Third-order	5.1e-3	1.8e-4	4.5e-2	3.3e-3	1.2e-4	1.1e-2

- Deep learning fit is an order of magnitude better on average, especially for output and investment, where perturbation predicts a boom.
- First-order predicts output $>$30% above its 2007 level by 2010, with investment surging $>$60%.

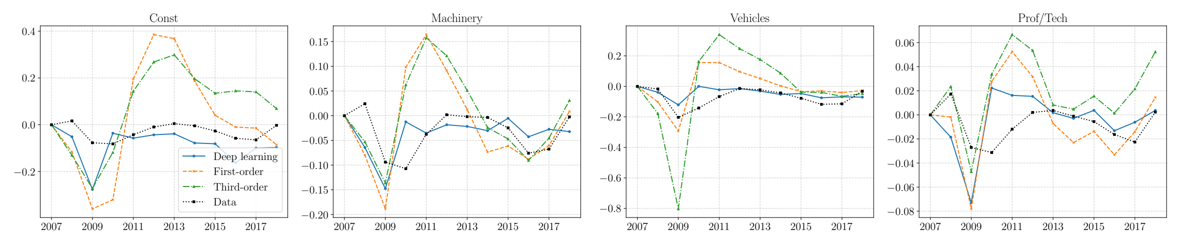
Investment production in investment hubs



Sectoral prices in investment hubs



Sectoral labor in investment hubs



The bottleneck mechanism

- As TFP recovers, prices in investment hubs decline.
- Perturbation: low prices trigger a massive reallocation toward investment goods $\Rightarrow$ counterfactual boom.
- Deep learning: the same price declines fail to trigger a boom because the nonlinear resource constraints bind.
- Feedback loop: depleted capital $\downarrow$ MPL $\Rightarrow \downarrow$ labor demand $\Rightarrow \downarrow$ investment production $\Rightarrow$ capital cannot recover.
- This is the slow recovery.

Why do perturbations fail?

The source of the nonlinearity

- With Cobb-Douglas production and log utility, the static model is log-linear: a first-order approximation is exact.
- Not in the dynamic model: capital accumulation is not log-linear once $\delta_i < 1$.
- Even higher-order Taylor approximations fail, because of singularities: points where capital demand blows up.
- With $\delta_i = 1$ the singularity is gone and the model has an exact global solution (Long and Plosser, 1983).

The crash: two speeds

- A negative shock hits sector i: $z_i \downarrow$, persistent (AR(1)).
- Price $=$ marginal cost $\propto e^{-z_i} \Rightarrow p_i \uparrow$, output $y_i \downarrow$ (the crash).
- Capital is slow: predetermined, moves only via $k_{i,t+1} = (1-\delta_i)k_{i,t} + x_{i,t}$.
- TFP is fast: AR(1) mean-reverts right away.
- Speed mismatch: productivity snaps back while capital only inches $\Rightarrow$ capital enters the recovery a bit depleted.

The recovery: prices fall, firms want to rebuild

- As z_i recovers, p_i falls back; the investment price $p_j^I = \prod_k (p_k / b_{jk})^{b_{jk}}$ falls with it.
- Firms want to rebuild: productivity is back and capital is scarce (high MPK), and investment goods are cheaper.
- The crash pushed investment prices up; the recovery pushes them down, toward the region where linear rules break.
- All solution methods agree on the crash; they split in the recovery.

The bottleneck: why the rebuild is capped

- Rebuilding capital needs more investment goods $x_j = \prod_k q_{jk}^{b_{jk}}$ from the hubs.
- But hub output is capital-constrained, and hub capital is depleted:
  - labor is elastic, but with low capital its marginal product is low $\Rightarrow$ diminishing returns.
- Resource constraint $y_i = c_i + \sum_j m_{ji} + \sum_j q_{ji}$: more investment carries a rising opportunity cost (forgone consumption, materials).
- Feedback loop: depleted capital $\rightarrow$ low output $\rightarrow$ low MPL $\rightarrow$ little investment production $\rightarrow$ capital stays low. This is the slow recovery.
- Perturbation *linearizes the constraint away* (constant shares) $\Rightarrow$ permits the reallocation $\Rightarrow$ counterfactual boom.

Resource constraints: the biggest error

- The largest linearization error is in the resource constraints:

$$\widehat{y}_{i,t} = \frac{c_i}{y_i}\widehat{c}_{i,t} + \sum_j \frac{m_{ji}}{y_i}\widehat{m}_{ji,t} + \sum_j \frac{q_{ji}}{y_i}\widehat{q}_{ji,t}$$

- With full depreciation ($\delta_i = 1$) the expenditure shares are constant, so this holds exactly.

- With $\delta_i < 1$ the shares move, and the error grows exponentially in $\widehat{q}_{ji,t} - \widehat{y}_{i,t}$.

- A large attempted reallocation, exactly what the recovery calls for, blows the linearized constraint up.

Capital dynamics

- Around steady state, investment moves a lot to deliver a given change in $k_{i,t+1}$:

$$\widehat{k}_{i,t+1} = (1-\delta_i)\widehat{k}_{i,t} + \delta_i \widehat{x}_{i,t}$$

- Linearize instead around some other $k_{i,t}$, $x_{i,t}$:

$$\bar{k}_{i,t+1} = \frac{(1-\delta_i)k_{i,t}}{k_{i,t+1}}\bar{k}_{i,t} + \frac{x_{i,t}}{k_{i,t+1}}\bar{x}_{i,t}$$

  now $\bar{x}_{i,t}$ responds inversely to $x_{i,t}/k_{i,t+1}$.

- So the larger the initial investment, the less elastic investment is to $\bar{k}_{i,t+1}$.
- Investment is log-concave in desired capital, which dampens its fluctuations, the smaller side of the paradox.

Euler equation and the singularity

- Recall the optimal capital choice in sector i:

$$p^I_{i,t} = \beta\,\mathbb{E}_t\left[\left(\frac{C_{t+1}}{C_t}\right)^{-\sigma}\left(\underbrace{p_{i,t+1}\,\alpha^k_i\,\frac{y_{i,t+1}}{k_{i,t+1}}}_{\text{marginal product}} + \underbrace{(1-\delta_i)\,p^I_{i,t+1}}_{\text{resale value}}\right)\right]$$

- Rearranging gives capital demand $k_{i,t+1} = \dfrac{\beta\,\mathbb{E}_t\left[\left(\frac{C_{t+1}}{C_t}\right)^{-\sigma} p_{i,t+1}\,\alpha^k_i\,y_{i,t+1}\right]}{u_{i,t}}$, with user cost:

$$u_{i,t} = p^I_{i,t} - (1-\delta_i)\,\beta\,\mathbb{E}_t\left[\left(\frac{C_{t+1}}{C_t}\right)^{-\sigma} p^I_{i,t+1}\right]$$

- As $u_{i,t} \to 0$, capital becomes a near-free asset and $k_{i,t+1} \to \infty$: a singularity.

Radius of convergence

- Taylor series have a finite radius of convergence R, set by the nearest singularity p_s:

$$R = |p_s - p_0|$$

- Converges to the true function for $|p - p_0| < R$; diverges for $|p - p_0| > R$.
- Adding more terms (higher-order perturbation) improves accuracy within R but cannot extend R.
- This is why the third-order solution can be *worse* in the tails.

Distance to the singularity

- Perturb the current investment price, holding SDF, expected MPK, and future prices at steady state:

$$\frac{k_{i,t+1}}{k_i} = \frac{u_i}{u_i + p_i^I(e^{\varepsilon_t} - 1)}$$

- Singularity when the denominator hits zero:

$$\varepsilon_t^* = \log\big(\beta(1 - \delta_i)\big)$$

- Typical calibration ($\beta = 0.96$, $\delta = 0.10$): $\varepsilon_t^* = \log(0.864) \approx -0.146$.
- A 14.6% drop in investment prices hits the singularity.
    - In Greenwood et al. (2000) investment-price deviations stay $<10\%$ $\Rightarrow$ bounded for one-sector models.

From one sector to the network

- In a multi-sector model a TFP shock does not stay confined: it propagates and moves investment prices economy-wide.
- Three factors decide whether a shock to sector i pushes the economy near a singularity:
  1. *Network propagation.* A shock to i moves sector j's log investment price by ψ_{ji}, where $\Psi = BL$, $L = (I - A)^{-1}$.
  2. *Distance to singularity.* Sector j's singularity is at $|\varepsilon_j^*| = |\log(\beta(1 - \delta_j))|$.
  3. *Aggregation.* Weight by Domar weights w_j, scale by $\sigma(\ln z_i)$.

The singularity exposure index

- Combining the three factors, define for each sector i:

$$SE_i = \sigma(\ln z_i) \sum_{j=1}^{n} w_j \frac{\psi_{ji}}{|\log(\beta(1-\delta_j))|}$$

- Computable directly from calibration parameters, no need to solve the model.
- SE_i is high when sector i has volatile TFP ($\sigma(\ln z_i)$ large), strongly influences investment prices (ψ_{ji} large) for sectors near their singularity ($|\log(\beta(1-\delta_j))|$ small) that are large (w_j large).
- Prediction: high-SE_i sectors show the largest deep learning vs. perturbation gaps.

SE_i predicts the volatility gap

	DL vs. First-order		DL vs. Third-order	
Metrics	Labor	Invest.	Labor	Invest.
w_i	0.301	0.253	0.303	0.227
δ_i	0.533	0.494	0.465	0.348
α_i^k	-0.076	-0.013	-0.079	0.035
α_i^h	-0.026	-0.037	0.016	0.006
$\sigma(\ln z_i)$	0.383	0.430	0.381	0.481
SE_i	0.785	0.718	0.780	0.699

SE_i is the only consistent predictor

	DL vs. First-order		DL vs. Third-order	
	Labor	Invest.	Labor	Invest.
w_i	0.040**	0.203	0.037**	0.145
δ_i	0.029*	0.185*	0.018	0.068
α_i^k	−0.004	−0.012	−0.004	−0.005
α_i^h	0.000	0.003	0.001	0.006
$\sigma(\ln z_i)$	0.041	0.345	0.036	0.336
SE_i	0.050***	0.224**	0.052***	0.232**
R^2	0.703	0.601	0.671	0.541

$^{*}\ p < 0.1$, $^{**}\ p < 0.05$, $^{***}\ p < 0.01$

- Controlling for all sectoral characteristics jointly, SE_i is the only consistently significant predictor.

Takeaways

Conclusions

- A scalable deep learning framework, built on symmetry, solves high-dimensional production-network models globally.
- A paradox: the same nonlinearities make ordinary fluctuations smaller, yet make recovery from a large shock slower, reproducing the Great Recession's slow recovery through bottlenecks in investment hubs.
- Perturbation misses this: it predicts a counterfactual boom and misidentifies which sectors drive the cycle.
- It fails at singularities in the capital Euler equation; our singularity exposure index, computable from calibration alone, is the dominant predictor of the gap.
- All from durable-capital accumulation, under Cobb-Douglas technology and log utility.