# An Adversarial Approach to Structural Estimation

Tetsuya Kaji[1], Elena Manresa[2], and Guillaume Pouliot[1]

[1]University of Chicago, [2]New York University

Presented by Andrew Capron

April 12, 2023

THE UNIVERSITY
*of* NORTH CAROLINA
*at* CHAPEL HILL

# Roadmap

# Why Do Economists Use Structural Estimation?

- Structural estimation allows economists to recover the "deep" parameters of economic models in order to:
  1. Conduct policy-invariant counterfactual experiments,
  2. Compare the importance of various mechanisms in the economic model.
- With fully-specificed parametric models, we often use maximum likelihood estimation (MLE).
- However, the likelihood function may not exist in closed-form, so simulation methods (e.g. SMM) are the popular alternative.

# Simulation

Simulation methods may suffer from:

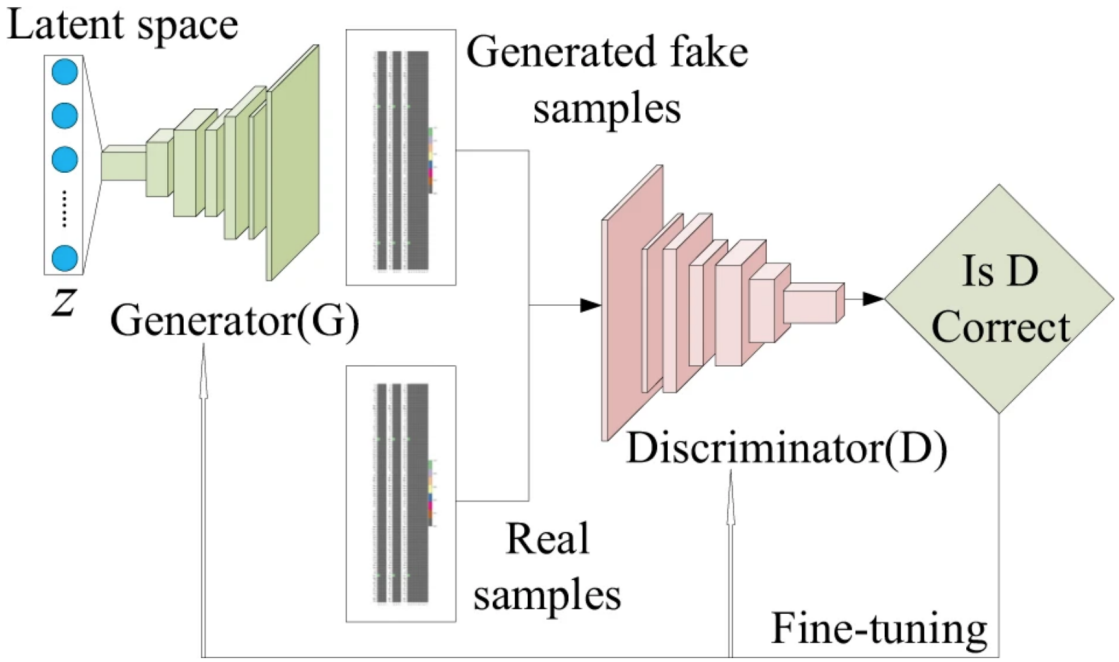
1. Finite sample bias,
2. The curse of dimensionality when incorporating rich heterogeneity (i.e. data must grow exponentially as the dimensionality of the state space grows to obtain reliable estimates),
3. Ad-hoc decision-making.

Adversarial estimation is a flexible alternative that, under certain conditions, captures the attractive features of both MLE and SMM.

# Roadmap

1. Motivation
2. Adversarial Estimator
3. Application

# Inspiration: Generative Adversarial Networks




**Minimax game:**

$$\min_{\{generator\}} \max_{\{discriminator\}} classification\ accuracy$$

# General Setup

- We will consider a parametric model with an untractable likelihood.
- Data: outcomes $y_i \in \mathbb{R}^L$ and covariates $z_i \in \mathbb{R}^M$, drawn from *unknown* conditional distribution, $y_i | z_i \overset{i.i.d.}{\sim} P_0$.
- Model: $T_\theta$ that maps $(\theta, z_i, \epsilon_i) \in \Theta \times \mathbb{R}^M \times \mathbb{R}^L \to y_{i,\theta} \in \mathbb{R}^L$.
  - $\theta$ must be finite-dimensional, ruling out FE and random coefficients.
- Let $X_i = h(y_i, z_i)$ and $X_{i,\theta} = h(y_{i,\theta}, z_i)$ be aspects of the data used in estimation.

# General Setup

- We will consider a parametric model with an untractable likelihood.
- Data: outcomes $y_i \in \mathbb{R}^L$ and covariates $z_i \in \mathbb{R}^M$, drawn from *unknown* conditional distribution, $y_i | z_i \overset{i.i.d.}{\sim} P_0$.
- Model: $T_\theta$ that maps $(\theta, z_i, \epsilon_i) \in \Theta \times \mathbb{R}^M \times \mathbb{R}^L \to y_{i,\theta} \in \mathbb{R}^L$.
  - $\theta$ must be finite-dimensional, ruling out FE and random coefficients.
- Let $X_i = h(y_i, z_i)$ and $X_{i,\theta} = h(y_{i,\theta}, z_i)$ be aspects of the data used in estimation.

As an example, consider a normal location model, with conditional distribution $y_{i,\theta} | z_i \sim P_\theta = N(\theta Z, 1)$. Then, $T_\theta$ converts $\tilde{y}_i \sim \tilde{P}_0 = N(0, 1)$ to $P_\theta$ using $y_{i,\theta} = T_\theta(\tilde{y}_i) = \tilde{y}_i + \theta z_i$.

In this case, $X_i$ and $X_{i,\theta}$ could be moments chosen from the real and simulated data to be used in estimation.

# Adversarial Estimator

- Generator: structural model that generates data.
- Discriminator: classification algorithm that attempts to differentiate between real and simulated data.
  - $D(x)$ is the probability that $x$ was drawn from the real data,
  - $1 - D(x)$ is the probability that $x$ was drawn from the economic model.
  - Note: i.i.d. cross-sectional or panel data only, not time series (possible, but requires changes to the structure of the discriminator).

Then, the estimator is defined as:

$$\hat{\theta} = \underset{\theta \in \Theta}{\operatorname{argmin}} \max_{D \in \mathcal{D}_n} \underbrace{\frac{1}{n} \sum_{i=1}^{n} log D(X_i)}_{\textit{Real Data}} + \underbrace{\frac{1}{m} \sum_{i=1}^{m} log(1 - D(X_{i,\theta}))}_{\textit{Simulated Data}}$$

The inner maximization is essentially maximum likelihood estimation, while the outer minimization searches for $\theta$ that minimizes the classification accuracy with respect to a class of discriminators.

# Oracle Discriminator

With $n, m \to \infty$, the population counterpart becomes:

$$\theta = \underset{\theta \in \Theta}{\operatorname{argmin}} \max_{D \in \mathcal{D}_n} \mathbb{E}_{X_i \sim P_0}[log D(X_i)] + \mathbb{E}_{X_{i,\theta} \sim P_\theta}[log(1 - D(X_{i,\theta}))]$$

- For both the sample and population problems, the inner maximization will always yield a number $\mathbb{M}_\theta \in [2log(1/2), 0]$.
- With no restrictions on the class of discriminators ($\mathcal{D}_n$), the "oracle" discriminator is defined as the best classifier for the population inner maximization:

$$D_\theta(x) \equiv \frac{p_0(x)}{p_0(x) + p_\theta(x)},$$

where $p_\bullet(x)$ represents the conditional pdf for both the real and simulated datasets.

# Example: Logistic Distribution Discriminator

$$\hat{\theta} = \underset{\theta \in \Theta}{\operatorname{argmin}} \max_{\beta} \ \frac{1}{n} \sum_{i=1}^{n} log(\Lambda(X_i^T \beta)) + \frac{1}{m} \sum_{i=1}^{m} log(1 - \Lambda(X_{i,\theta}^T \beta)),$$

where $\Lambda(x) = \frac{1}{1+e^{-x}}$ (i.e. the standard logistic CDF).

For a given candidate $\theta$, the FOC of the inner max problem yields:

$$\frac{1}{n} \sum_{i=1}^{n} (1 - \hat{D}(X_i^T \hat{\beta}(\theta))) X_i - \frac{1}{m} \sum_{i=1}^{m} \hat{D}(X_{i,\theta}^T \hat{\beta}(\theta)) X_{i,\theta} = 0,$$

where $\hat{D}(\bullet)$ represent the fitted values from the logistic regression.

When $\theta \neq \theta_0$, the FOC holds trivially. However, when $\theta = \theta_0$, $\hat{\beta} = o_p(1) \implies$

$$\frac{1}{n} \sum_{i=1}^{n} X_i - \frac{1}{m} \sum_{i=1}^{m} X_{i,\theta} = o_p(1).$$

# Computational Algorithm

1. Inputs (choices):
    1. Actual (transformed) data $X_i$,
    2. Structural model $T_\theta$,
    3. Known distribution $\tilde{P}_0$,
    4. Simulation sample size m,
    5. Discriminator class $\mathcal{D}_n$.
2. Draw $\tilde{X}_i$ from $\tilde{P}_0$ (only once),
3. Set initial guess $\hat{\theta}$,
4. Compute $X_{i,\theta} = T_\theta(\tilde{X}_i)$,
5. Solve inner maximization to get $\mathbb{M}_{\hat{\theta}}$ (e.g. logistic regression by MLE),
6. Obtain new candidate $\tilde{\theta}$ (e.g. gradient descent of objective function),
7. Repeat Steps $4-6$ until $\hat{\theta}$ converges.

# Example Using Logistic Location Model: Setup

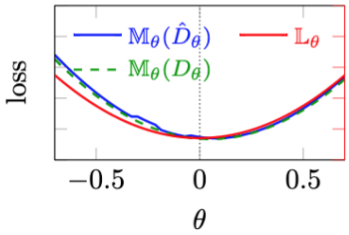
- Let $\mathbb{L}_\theta$ denote the log-likelihood and $\mathbb{M}_\theta(D)$ be the objective function.
- Given n i.i.d. draws from the standard logistic distribution with pdf $p_0(x) = \Lambda(x)(1 - \Lambda(x))$, we can write the structural model as $p_\theta(x) = \Lambda(x - \theta)(1 - \Lambda(x - \theta))$.
- Simulated data is generated as $X_{i,\theta} = T_\theta(\tilde{X}_i) = \tilde{X}_i + \theta$, where $\tilde{X}_i \sim Logistic(0, 1)$.
- The oracle discriminator and our best representation are respectively:

$$D_\theta(x) = \Lambda(-\theta - 2log(1 + e^{-x}) + 2log(1 + e^{-(x-\theta)})),$$
$$D_\lambda(x) = \Lambda(\lambda_0 - 2log(1 + e^{-x}) + 2log(1 + e^{-(x-\lambda_1)})),$$

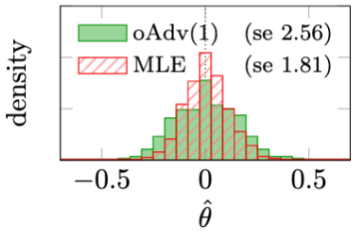
where $D_\lambda(x)$ is "correctly specified" in the sense that the oracle is given by $\lambda_\theta = (-\theta, \theta)^T$.

# Comparison With MLE: Efficiency

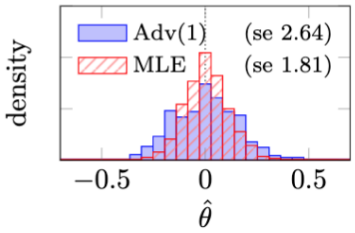
## Parametric specification:




(a) Curvature of cross-entropy loss and log likelihood.

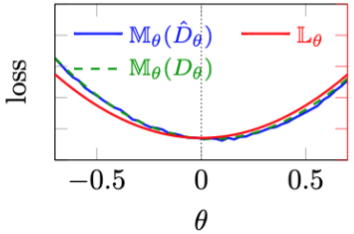



(b) Oracle adversarial estimator and MLE.

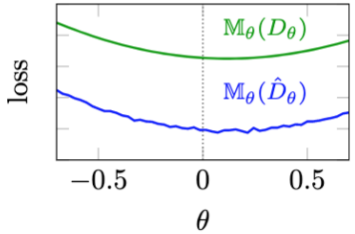



(c) Adversarial estimator and MLE.

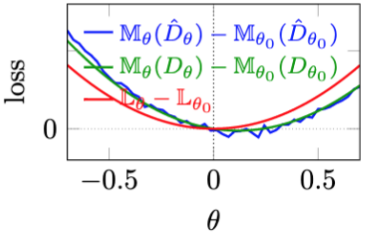
## Nonparametric specification (shallow neural network):




(a) $m = n$. The curve of $\mathbb{M}_\theta(\hat{D}_\theta)$ matches the oracle.



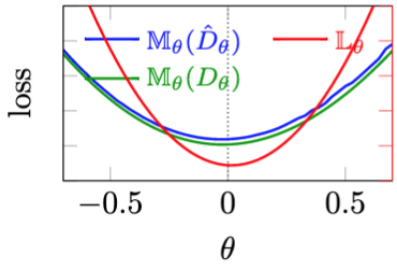

(b) $m = 2n$. The level is off, but the curvature is right.



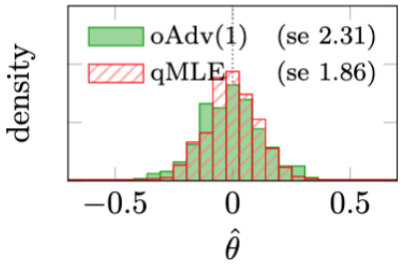

(c) $m = 2n$. Demeaned (b) to highlight the curvature.

▸ Neural Network Example

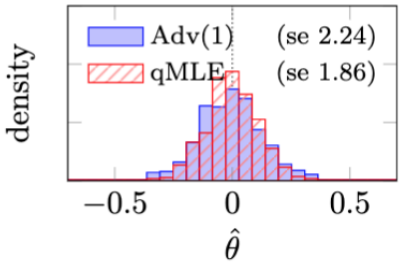
# Normality Under Misspecification




(a) Loss and quasi-log likelihood.




(b) Oracle adversarial estimator and quasi-MLE.




(c) Adversarial estimator and quasi-MLE.

With $p_\theta(x)$ misspecificed as the normal instead of the (correct) logistic distribution, the estimator is still comparable to QMLE on efficiency.

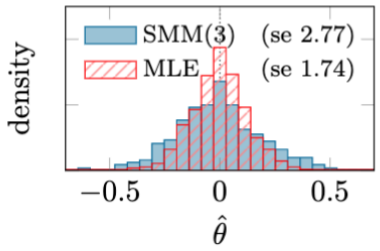
# Comparison With SMM: The Real Upside

While asymptotically identical to SMM with the logistic discriminator, the adversarial estimator can have large finite-sample efficiency gains.

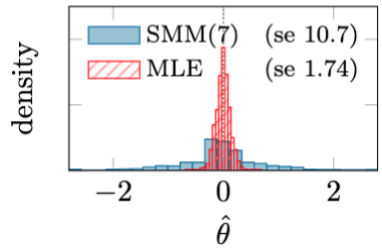
In the logistic case, the collection of order statistics is the minimal sufficient statistic for identifying the mean of the distribution.

We compare the two methods using the first 3, 7, and 11 moments in SMM with the discriminator $D_\lambda(x) = \Lambda(\lambda_0 + \lambda_1 x + ... + \lambda_d x^d)$ for d $= 3$, 7, and 11 in the adversarial estimator.

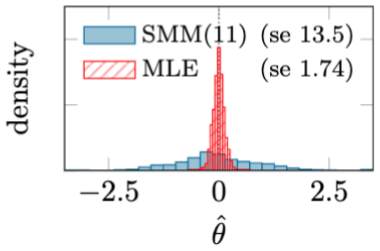
# Comparison With SMM: The Real Upside



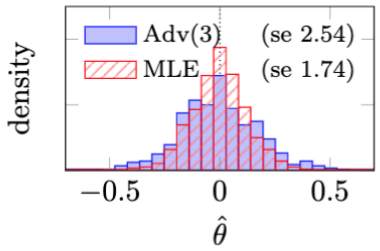

(a) SMM with 3 moments.



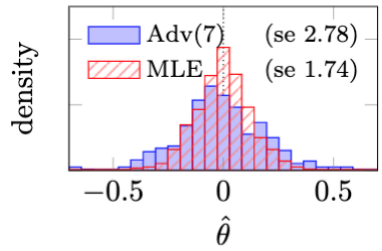

(b) SMM with 7 moments.



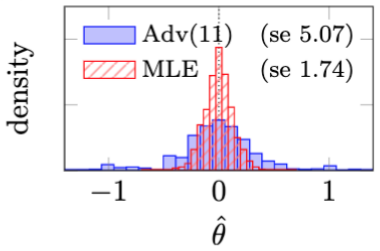

(c) SMM with 11 moments.




(d) Adversarial estimator with 3 moments.




(e) Adversarial estimator with 7 moments.




(f) Adversarial estimator with 11 moments.

Adversarial estimator far less sensitive to the number of moments compared with SMM. Very advantageous when making ad-hoc decisions about which moments to include.

# Roadmap

# Why Do The Elderly Save?

Compare the adversarial estimator against SMM in the application explored by De Nardi, French, Jones (JPE, 2010).

Motives for precautionary saving?

1. Survival uncertainty,
2. Medical expenses,
3. Bequests.

The authors show that the adversarial estimator performs much better when incorporating (important) rich heterogeneity, due to the curse of dimensionality when using SMM.

# High-Level Overview of Model

- Heterogenous retirees outside the labor force obtain utility from two actions:

  Consumption: $u(c) = \frac{c^{1-\nu}}{1-\nu}$, and

  Bequests: $\phi(e) = \frac{\theta(e+k)^{1-\nu}}{1-\nu}$.

- Agents maximize the sum of discounted utilities, taking into account uncertainty shocks that are conditional on gender, age, health status, and permanent income.
- Authors consider two sets of moments using a neural network discriminator:
  1. $X_1 = \log(age_{1996})$, $\bar{I}$, asset profile, survival profile.
  2. $X_2 = X_1$, gender, health status profile.
- SMM is unable to incorporate gender or health status due to the number of moments.

# Bequests Are An Important Motive

| | $\beta$ | $\underline{c}$ [\$] | $\nu$ | $\vartheta$ | $k$ [k\$] | MPC | $\underline{a}$ [\$] | Loss |
|---|---|---|---|---|---|---|---|---|
| DFJ, Table 3 | 0.97 | 2,665 | 3.84 | 2,360 | 273 | 0.12 | 36,215 | −0.67 |
| | (0.05) | (353) | (0.55) | (8,122) | (446) | | | |
| Adversarial $X_1$ | 0.97 | 4,500 | 6.14 | 4,865 | 16.89 | 0.20 | 4,243 | −0.67 |
| | | | (.009) | (9.002) | (.030) | (.017) | (19.73) | |
| Adversarial $X_2$ | 0.97 | 4,500 | 5.99 | 192,676 | 10.02 | 0.12 | 1,320 | −0.78 |
| | | | (.005) | (8,112) | (.015) | (.014) | (3.66) | |

DFJ has difficulty explaining why bequests motivate savings. The adversarial estimator shows that bequests matter across the entire income distribution. SMM fails to explain a key feature of the model.

# Shallow Neural Network Discriminator

- Single-layer simple neural network with activation function g(•):
$$D(X, \beta) = g(\alpha_0 + \sum_{k=1}^{d_1} \gamma_k g(\alpha_{0,k} + \lambda_k X)),$$
where $\beta = (\alpha_0, \alpha_{0,k}, \gamma_k, \lambda_k)$.
- Neural networks use training examples $X_1, X_2, \ldots, X_m$ with known classifications $y_1, y_2, \ldots, y_m$ to minimize the cost function (using a weighted difference between predicted and true outcomes) with respect to the model parameters.
- Neural networks are attractive for their ability to handle rich heterogeneity through dimension reduction and low-dimensional prediction (i.e. $X_i$ much higher dimensional input than $y_i$).

Return