CHICAGO BOOTH
The University of Chicago Booth School of Business

Adversarial Bayesian Simulation

Yuexi Wang

joint work with Veronika Ročková

Booth School of Business
University of Chicago

November 24, 2022

Simulation-Based Inference

The basic Bayesian ingredients

- Data $\boldsymbol{X} = \{X_i\}_{i=1}^n$ realized from $P_{\theta_0}^{(n)}$ indexed by $\theta_0 \in \Theta$
- Prior $\Pi(\cdot)$
- Model $P_\theta^{(n)}$, for each $\theta \in \Theta$, admits a density $p_\theta^{(n)}$

We focus on posterior

$$\pi(\theta \mid \mathbf{X}) \propto \underbrace{p_\theta^{(n)}(\mathbf{X})} \times \pi(\theta)$$

too costly to evaluate ☹
but can be readily sampled from ☺

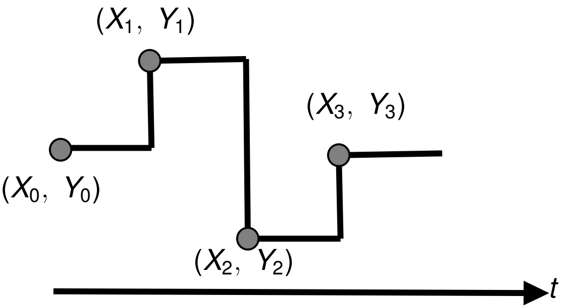
- ⇝ Heston model: Stochastic volatility dynamics in finance
- ⇝ Lotka-Volterra model: Predator-prey population dynamics in ecology.
- ⇝ SIR model: Disease spreading dynamics in epidemiology.

Lotka-Volterra (LV) model with $\theta = (\theta_1, \theta_2, \theta_3, \theta_4)'$

$$\frac{\mathrm{d}\, X_t}{\mathrm{d}\, t} = \underbrace{\theta_1 X_t Y_t}_{\text{predator being born}} - \underbrace{\theta_2 X_t}_{\text{predator dying}}$$

$$\frac{\mathrm{d}\, Y_t}{\mathrm{d}\, t} = \underbrace{\theta_3 Y_t}_{\text{prey being born}} - \underbrace{\theta_4 X_t Y_t}_{\text{prey dying}}$$

Stochastic Markov Jump Process



Despite easy to sample from (using the Gillespie algorithm), the likelihood for this model is unavailable.

Approximate Bayesian Computation (ABC)

1. Simulate pair $(\theta_j, \widetilde{\boldsymbol{X}}^{\theta_j})$ from the joint distribution $p(\theta, \widetilde{\boldsymbol{X}}^{\theta}) = \pi(\theta) \times p_{\theta}^{(n)}(\widetilde{\boldsymbol{X}}^{\theta})$
2. Picks θ_j if $\widetilde{\boldsymbol{X}}_j^{\theta}$ looks “similar" to $\boldsymbol{X}$

   One solution: Accept θ_j if $\left\| S(\widetilde{\boldsymbol{X}}^{\theta_j}) - S(\boldsymbol{X}) \right\| \leq \epsilon$.

☹ *Reliance on summary statistics*

☹ *Posterior shapes vary with how the ABC draws are weighted*

☺ *Parallel computation feasible*

☺ *Not sensitive to initialization*

Adversarial Learning

A good classifier can tell us how “similar” the two datasets are.

Our Approaches

- **IID data**
  - In the LV model, each observation is a time series and we observe n iid copies
  - $p_\theta^{(n)}(\boldsymbol{X}) = \prod_{i=1}^n p_\theta(X_i) = p_{\theta_0}^{(n)}(\boldsymbol{X}) \times \boxed{\prod_{i=1}^n \frac{p_\theta(X_i)}{p_{\theta_0}(X_i)}}$
  - *ABC via Classification!*
    Wang, Kaji, and Ročková [2022]. JMLR.
- **Dependent data**
  - For the LV model, we only observe one time series
  - $p_\theta^{(n)}(\boldsymbol{X})$ has no product form
  - *Bayesian Generative Adversarial Networks (B-GANs)!*
    Wang and Ročková [2022]. arXiv:2208.12113.

ABC via Classification[1]

[1] Wang, Kaji, and Ročková [2022]. JMLR

The Classification Trick

Now we have n iid 'real' observations, and we **simulate** m 'fake' observations from P_θ

We consider a classification problem as

$$\max_{D \in \mathcal{D}} \left[\frac{1}{n} \sum_{i=1}^{n} \log D(X_i) + \frac{1}{m} \sum_{i=1}^{m} \log(1 - D(\widetilde{X}_i^\theta)) \right], \tag{1}$$

where $D : \mathcal{X} \to (0, 1)$ (1 for 'real' and 0 for 'fake' data).

- When θ is **close** to θ_0, P_θ and P_{θ_0} are very similar

  $\Rightarrow$ Hard to distinguish $\widetilde{\boldsymbol{X}}^\theta$ from $\boldsymbol{X}$.
  $\Rightarrow$ $\hat{D}(\widetilde{X}_j^\theta)$ **close to 0.5**

- When θ is **far** from θ_0, P_θ and P_{θ_0} are very different

  $\Rightarrow$ Easy to distinguish $\widetilde{\boldsymbol{X}}^\theta$ from $\boldsymbol{X}$.
  $\Rightarrow$ $\hat{D}(\widetilde{X}_j^\theta)$ **close to 0**

Estimating KL via Classification

We adopt KL divergence $K(p_{\theta_0}, p_\theta)$ inside ABC, first proposed by Jiang et al. [2018].

Oracle discriminator to the log loss in Eq. (1) is

$$D_\theta^O(X) := \frac{p_{\theta_0}(X)}{p_{\theta_0}(X)+p_\theta(X)} \Rightarrow \frac{p_0}{p_\theta}(X) = \frac{D_\theta^O(X)}{1-D_\theta^O(X)}$$

replace D_θ^O with $\hat{D}_{n,m}$

KL estimator

$$\hat{K}\left(\boldsymbol{X}, \tilde{\boldsymbol{X}}^\theta\right) = \frac{1}{n}\sum_{i=1}^{n} \log \frac{\hat{D}_{n,m}(X_i)}{1-\hat{D}_{n,m}(X_i)}$$

$\Rightarrow$Accept-Reject ABC

Likelihood Ratio estimator

$$\prod_{i=1}^{n} \frac{\hat{D}_{n,m}(X_i)}{1-\hat{D}_{n,m}(X_i)} = \exp\left(-n\hat{K}(\boldsymbol{X}, \tilde{\boldsymbol{X}}^\theta)\right)$$

$\Rightarrow$Exponential-Weighted ABC

Accept and Reject ABC (AR-ABC)

For a pre-determined tolerance level $\epsilon_n > 0$, repeat for $j = 1, \ldots, N$:

1. Simulate θ_j from $\pi(\theta)$.
2. Simulate $\widetilde{\boldsymbol{X}}^{\theta_j} = (\widetilde{X}_1^{\theta_j}, \ldots, \widetilde{X}_m^{\theta_j})'$ through i.i.d sampling from the model P_{θ_j}.
3. Construct $\hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^{\theta_j})$ by training a classifier distinguishing $\boldsymbol{X}$ and $\widetilde{\boldsymbol{X}}^{\theta_j}$.
4. Accept θ_j when $\hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^{\theta_j}) \leq \epsilon_n$.

$$\hat{\pi}^{AR}(\theta \mid \boldsymbol{X}) = \frac{\int \pi(\theta) p_\theta^{(n)}(\widetilde{\boldsymbol{X}}^\theta) \mathbb{I}\left(\hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^\theta) \leq \epsilon_n\right) \mathrm{d}\widetilde{\boldsymbol{X}}^\theta}{\int\int \pi(\theta) p_\theta^{(n)}(\widetilde{\boldsymbol{X}}^\theta) \mathbb{I}\left(\hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^\theta) \leq \epsilon_n\right) \mathrm{d}\widetilde{\boldsymbol{X}}^\theta \, \mathrm{d}\theta}$$

- ☺ Posterior concentration rate depends on estimation error δ_n and threshold ϵ_n. Consistency is guaranteed as long as $\epsilon_n \to 0$
- ☹ The proper choice of ϵ_n is still unclear for complex models

Exponential-Weighted ABC

Motivated by the connection between KL and the likelihood ratio, we propose a scaled exponential kernel that requires no ad hoc scaling

$$\hat{\pi}^{EK}(\theta \mid \boldsymbol{X}) = \frac{\int \pi(\theta) p_\theta^{(n)}(\widetilde{\boldsymbol{X}}^\theta) \exp\left(-n\hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^\theta)\right) \mathrm{d}\widetilde{\boldsymbol{X}}^\theta}{\int\int \pi(\theta) p_\theta^{(n)}(\widetilde{\boldsymbol{X}}^\theta) \exp(-n\hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^\theta)) \mathrm{d}\widetilde{\boldsymbol{X}}^\theta \mathrm{d}\theta}$$

We can rewrite the approximated posterior as

$$\hat{\pi}^{EK}(\theta \mid X^{(n)}) \propto \underbrace{p_\theta^{(n)}(\boldsymbol{X}) e^{\hat{u}_\theta(\boldsymbol{X})}}_{\text{Model mis-specification}} \pi(\theta), \tag{2}$$

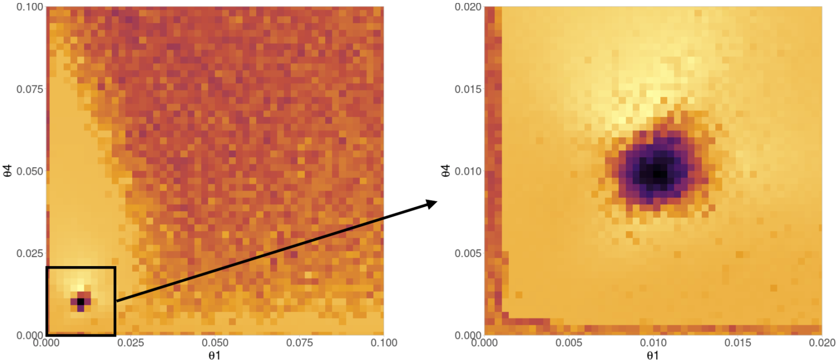
where $\hat{u}_\theta(\boldsymbol{X}) = \log \int e^{-n\times\left(\hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^\theta) - \frac{1}{n}\sum_{i=1}^n \log \frac{p_{\theta_0}}{p_\theta}(X_i)\right)} \mathrm{d}\widetilde{\boldsymbol{X}}^\theta$.

Eq.(2) can be characterized as a posterior under a mis-specified likelihood $\tilde{p}_\theta$.

$\rightsquigarrow$ Concentration around θ^* (KL projection point)

Lotka-Volterra: Likelihood is Spiky!

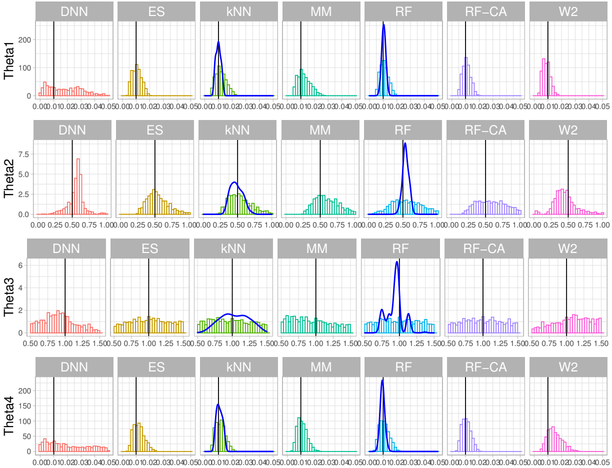
Simulation starts at $X_0 = 50$ and $Y_0 = 100$ and is over 20 time units with intervals of 0.1, resulting in a series of $T = 201$ observations each. Fix $\theta_2 = 0.5, \theta_3 = 1$ and only change θ_1 and θ_4.



- Narrow range of likely parameter values
  - Interaction patterns are very sensitive to parameter changes
- MCMC convergence speed is very sensitive to the initialization

ABC Results

True values $\theta_0 = (0.01, 0.5, 1, 0.01)$. Uniform Prior on $[0, 0.1] \times [0, 1] \times [0, 2] \times [0, 0.1]$



▸ Other ABC methods

Conclusions: Part I

- ☺ We have developed an ABC approach which obviates the need for summary statistics.
- ☺ We adopt two versions of ABC
  - Accept-Reject ABC
  - **New!** Exponential-Weighted ABC that requires no ad hoc thresholding

Yet limitations?

- How to construct a reasonable classifier for dependent data is unclear
- The computation costs for training a new classifier after each ABC draw is daunting.

Even Better!

Adversarial Bayesian Simulation[2]

[2] Wang and Ročková [2022]. arXiv:2208.12113.

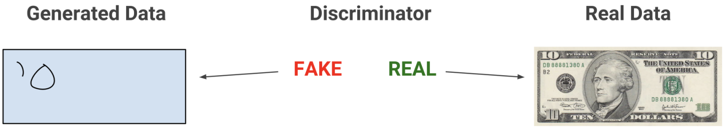
Generative Adversarial Networks (GANs)

Generator against Discriminator

https://this-person-does-not-exist.com/

Generator against Discriminator

When training begins, the generator produces obviously fake data, and the discriminator quickly learns to tell that it's fake:



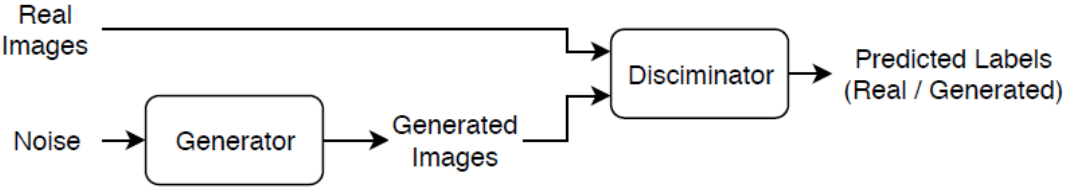
As training progresses, the generator gets closer to producing output that can fool the discriminator:

Finally, if generator training goes well, the discriminator gets worse at telling the difference between real and fake. It starts to classify fake data as real, and its accuracy decreases.

[3] credit to `https://developers.google.com/machine-learning/gan/gan_structure`

GANs

“generate a fake human face image”



The two-player min-max game with Generator g and Discriminator d

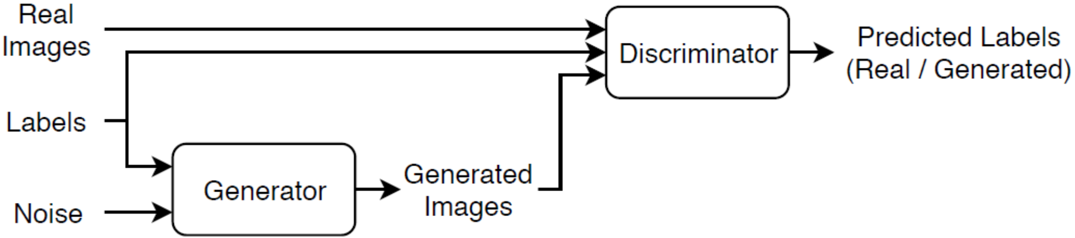
$$\min_{g \in \mathcal{G}} \max_{d \in \mathcal{D}} P_{X^{(n)} \sim \pi(X^{(n)})} \log d(X^{(n)}) + P_{Z \sim \pi_Z(Z)} \log(1 - d(g(Z)))$$

The generator g learns to approximate the **marginal** distribution $\pi(X^{(n)})$.

Draws from the implicit distribution $\pi(X^{(n)})$ are obtained by passing a random noise vector $Z \in \mathcal{Z} \in \mathbb{R}^{d_Z}$ through a non-stochastic mapping $g : \mathcal{Z} \to \mathcal{X}$.

Conditional GANs (cGANs)

What if we want to generate distribution conditioned on some extra information, like labels for images ('cat', 'dog' etc.)?



- The generator generates fake images given the labels.
  "generate a fake cat image"
- The discriminator distinguishes pairs of (real image, label) and (fake image, label).
  "how is (real cat image, cat) different from (fake cat image, cat)"

Generate 'fake' θ given $X^{(n)}$

Input		$\{\text{image}_i, \text{label}_i\}_{i=1}^T$	$\{\theta_i, X_{\theta_i}^{(n)}\}_{i=1}^T$
Generator		fake_image$_i$ given label$_i$	$\widehat{\theta}_i = g\left(Z_i, X_{\theta_i}^{(n)}\right)$
Discriminator	"1"	$(\text{image}_i, \text{label}_i)$ 	$\left(\theta_i, X_{\theta_i}^{(n)}\right)$ 
	"0"	(fake_image$_i$, label$_i$) 	$\left(\widehat{\theta}_i, X_{\theta_i}^{(n)}\right)$ 

cGANs

Consider the two-player min-max game

$$\min_{g \in \mathcal{G}} \max_{d \in \mathcal{D}} P_{X^{(n)},\theta \sim \pi(X^{(n)},\theta)} \log d\left(X^{(n)}, \theta\right) + P_{X^{(n)} \sim \pi(X^{(n)}),Z \sim \pi_Z(Z)} \log\left(1 - d\left(X^{(n)}, g(Z, X^{(n)})\right)\right) \tag{3}$$

Now $X^{(n)}$ enters both discriminator and generator.

- Fix the marginal distribution $\pi(X^{(n)})$,

  matching the **joint** distribution $\Leftrightarrow$ matching the **conditional** distribution

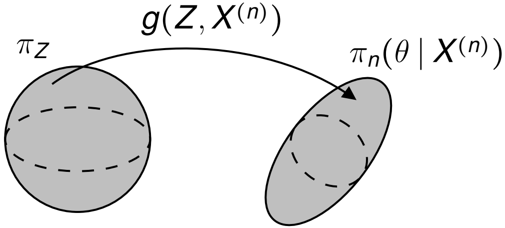
- With flexible enough $\mathcal{D}$ and $\mathcal{G}$, the solution (g^*, d^*) to the minimax game satisfies

$$\pi_{g^*}(\theta \mid X^{(n)}) = \frac{\pi(X^{(n)}, \theta)}{\pi(X^{(n)})} = \pi(\theta \mid X^{(n)}) \quad \text{for any } X^{(n)} \in \mathcal{X}$$

$$d_g^*(X^{(n)}, \theta) = \frac{\pi(X^{(n)}, \theta)}{\pi(X^{(n)}, \theta) + \pi_g(X^{(n)}, \theta)}$$

$g(\cdot, X^{(n)})$ is a pushforward mapping from π_Z to $\pi(\theta \mid X^{(n)})$

The observed data $X_0^{(n)}$ is **not used** in the training process.



Normal location?

$P_\theta = N(1_n\theta,\ I_n)$

$\pi(\theta) = N(0, 1)$

$Z \sim N(0, 1)$ → $\theta \mid X^{(n)} \sim N\left(\frac{\sum_{i=1}^n X_i}{n+1}, \frac{1}{n+1}\right)$

$$\theta = \frac{1}{\sqrt{n+1}} Z + \frac{\sum_{i=1}^n X_i}{n+1}$$

$\Rightarrow$ The approximate posterior is then obtained as

$$\theta \mid X_0^{(n)} \sim g(Z, X_0^{(n)})$$

Wasserstein GANs

However, conditional GANs suffer from training issues.

- The gradients of the generator vanish when discriminator is too strong [Arjovsky et al., 2017].
- cGAN does not work well with continuous conditions [Zhou et al., 2022].

Consider the *Wasserstein variant*

$$\min_{g \in \mathcal{G}} \max_{f \in \mathcal{F}} P_{X^{(n)} \sim \pi(X^{(n)}), Z \sim \pi_Z} f\left(g(Z, X^{(n)}), X^{(n)}\right) - P_{(\theta, X^{(n)}) \sim \pi(\theta, X^{(n)})} f(\theta, X^{(n)})$$

where $\mathcal{F}$ is the class of functions that are 1-Lipschitz with respect to θ.

Bayesian Simulation via WGANs (B-GANs)

The generator class $\mathcal{G}$ is parametrized with β and the critic class $\mathcal{F}$ is parametrized with ω.

Initialize networks f_ω and g_β.

1. **Generate** the ABC reference table.
   Simulate $\{X_j^{(n)}, \theta_j\}_{j=1}^T$ where $\theta_j \sim \pi(\theta)$ and $X_j^{(n)} \sim P_{\theta_j}^{(n)}$, and $\{Z_j\}_{j=1}^T \overset{\text{iid}}{\sim} \pi_Z(\cdot)$
2. **Train** the empirical version of Wasserstein loss.

$$\hat{\beta}_T = \arg\min_{\beta: g_\beta \in \mathcal{G}} \left[\max_{\omega: f_\omega \in \mathcal{F}} \left| \sum_{j=1}^T f_\omega\big(X_j^{(n)}, g_\beta(Z_j, X_j^{(n)})\big) - \sum_{j=1}^T f_\omega(X_j^{(n)}, \theta_j) \right| \right]$$

3. **Simulate** posterior.
   Generate $\{Z_i\}_{i=1}^M \overset{\text{iid}}{\sim} \pi_Z(Z)$, Predict $\tilde{\theta}_i = g_{\widehat{\beta}_T}(Z_i, X_0^{(n)})$.

We obtain approximated posterior draws $\{\tilde{\theta}_1, \ldots, \tilde{\theta}_M\}$.

Convergence in TV: Three Terms

Our result is built on oracle inequalities established in Liang [2021].

Theorem. Denote the solution with $\widehat{\beta}_T$ where $\mathcal{F} = \{f : \|f\|_\infty \le B\}$ for some $B > 0$. Assume

$$\Pi[B_n(\theta_0; \epsilon)] \ge e^{-C_2 n\epsilon^2} \text{ for some } C_2 > 2 \text{ and } \epsilon > 0.$$

For $T \ge P_{\max}$ we have for any $C > 0$

$$P_{\theta_0}^{(n)} \mathbb{P}_{X_0^{(n)}} d_{TV}^2\left(\pi(\theta \mid X_0^{(n)}), \pi_{\widehat{\beta}_T}(\theta \mid X_0^{(n)})\right) \le C_n^T(\widehat{\beta}_T, \epsilon, C),$$

where for some $\tilde{C} > 0$

$$C_n^T(\widehat{\beta}_T, \epsilon, C) = \frac{1}{C^2 n\epsilon^2} + \frac{e^{(1+C+C_2)n\epsilon^2}}{4}\left[2\mathcal{A}_1(\mathcal{F}, \widehat{\beta}_T) + \frac{B\mathcal{A}_2(\mathcal{G})}{\sqrt{2}} + 4\tilde{C}B\sqrt{\frac{\log T \times P_{\max}}{T}}\right].$$

The prior concentration condition ensures we have enough mass around the truth.

Convergence in TV: Three Terms

Our result is built on oracle inequalities established in Liang [2021].

Theorem. Denote the solution with $\widehat{\beta}_T$ where $\mathcal{F} = \{f : \|f\|_\infty \le B\}$ for some $B > 0$. Assume

$$\Pi[B_n(\theta_0;\epsilon)] \ge e^{-C_2 n\epsilon^2} \text{ for some } C_2 > 2 \text{ and } \epsilon > 0.$$

For $T \ge P_{\max}$ we have for any $C > 0$

$$P_{\theta_0}^{(n)} \mathbb{P}_{X_0^{(n)}} d_{\mathrm{TV}}^2\left(\pi(\theta \mid X_0^{(n)}), \pi_{\widehat{\beta}_T}(\theta \mid X_0^{(n)})\right) \le C_n^T(\widehat{\beta}_T, \epsilon, C),$$

where for some $\tilde{C} > 0$

$$C_n^T(\widehat{\beta}_T, \epsilon, C) = \frac{1}{C^2 n\epsilon^2} + \frac{e^{(1+C+C_2)n\epsilon^2}}{4}\left[2\boxed{\mathcal{A}_1(\mathcal{F}, \widehat{\beta}_T)} + \frac{B\mathcal{A}_2(\mathcal{G})}{\sqrt{2}} + 4\tilde{C}B\sqrt{\frac{\log T \times P_{\max}}{T}}\right].$$

The ability of the critic to express the class of density ratios

$$\mathcal{A}_1(\mathcal{F}, \widehat{\beta}_T) = \inf_{\omega : f_\omega \in \mathcal{F}} \left\| \log \frac{\pi(\theta \mid X^{(n)})}{\pi_{g_{\widehat{\beta}_T}}(\theta \mid X^{(n)})} - f_\omega(\theta, X^{(n)}) \right\|_\infty$$

Convergence in TV: Three Terms

Our result is built on oracle inequalities established in Liang [2021].

Theorem. Denote the solution with $\widehat{\beta}_T$ where $\mathcal{F} = \{f : \|f\|_\infty \le B\}$ for some $B > 0$. Assume

$$\Pi[B_n(\theta_0; \epsilon)] \ge e^{-C_2 n\epsilon^2} \text{ for some } C_2 > 2 \text{ and } \epsilon > 0.$$

For $T \ge P_{\max}$ we have for any $C > 0$

$$P_{\theta_0}^{(n)} \mathbb{P}_{X_0^{(n)}} d_{TV}^2\Big(\pi(\theta \mid X_0^{(n)}), \pi_{\widehat{\beta}_T}(\theta \mid X_0^{(n)})\Big) \le C_n^T(\widehat{\beta}_T, \epsilon, C),$$

where for some $\tilde{C} > 0$

$$C_n^T(\widehat{\beta_T}, \epsilon, C) = \frac{1}{C^2 n\epsilon^2} + \frac{e^{(1+C+C_2)n\epsilon^2}}{4}\left[2\mathcal{A}_1(\mathcal{F}, \widehat{\beta}_T) + \frac{B\,\boxed{\mathcal{A}_2(\mathcal{G})}}{\sqrt{2}} + 4\tilde{C}B\sqrt{\frac{\log T \times P_{\max}}{T}}\right].$$

The ability of the generator to approximate the average true posterior

$$\mathcal{A}_2(\mathcal{G}) = \inf_{\beta : g_\beta \in \mathcal{G}} \left[P_{X^{(n)}} \left\| \log \frac{\pi_{g_\beta}(\theta \mid X^{(n)})}{\pi(\theta \mid X^{(n)})} \right\|_\infty\right]^{1/2}$$

Convergence in TV: Three Terms

Our result is built on oracle inequalities established in Liang [2021].

Theorem. Denote the solution with $\widehat{\beta}_T$ where $\mathcal{F} = \{f : \|f\|_\infty \leq B\}$ for some $B > 0$. Assume

$$\Pi[B_n(\theta_0; \epsilon)] \geq e^{-C_2 n\epsilon^2} \text{ for some } C_2 > 2 \text{ and } \epsilon > 0.$$

For $\boxed{T \geq P_{\max}}$ we have for any $C > 0$

$$P_{\theta_0}^{(n)} \mathbb{P}_{X_0^{(n)}} d_{\mathrm{TV}}^2\left(\pi(\theta \mid X_0^{(n)}), \pi_{\widehat{\beta}_T}(\theta \mid X_0^{(n)})\right) \leq C_n^T(\widehat{\beta}_T, \epsilon, C),$$

where for some $\tilde{C} > 0$

$$C_n^T(\widehat{\beta_T}, \epsilon, C) = \frac{1}{C^2 n\epsilon^2} + \frac{e^{(1+C+C_2)n\epsilon^2}}{4}\left[2\mathcal{A}_1(\mathcal{F}, \widehat{\beta}_T) + \frac{B\mathcal{A}_2(\mathcal{G})}{\sqrt{2}} + 4\tilde{C}B\sqrt{\frac{\log T \times \boxed{P_{\max}}}{T}}\right].$$

Complexity in Pseudo Dim [Bartlett et al., 2017]

$$P_{\max} = \mathrm{Pdim}(\mathcal{F}) \vee \mathrm{Pdim}(\mathcal{H})$$

critic class

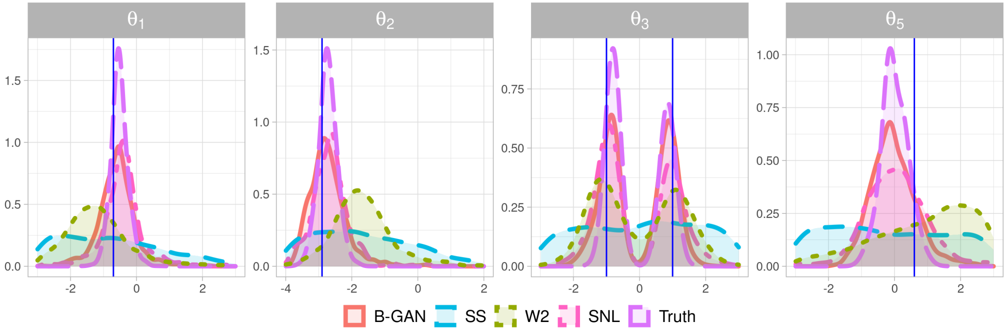
composition of critic and discriminator $\{h_{\omega,\beta}(Z, X) = f_\omega(g_\beta(Z, X), X)\}$

Toy Example

We observe bi-variate Gaussians $X^{(n)} = (X_1, X_2, X_3, X_4)'$ with $X_j \sim \mathcal{N}(\mu_{\boldsymbol{\theta}}, \Sigma_{\boldsymbol{\theta}})$ parametrized by $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5)'$, where

$$\mu_{\boldsymbol{\theta}} = (\theta_1, \theta_2)' \quad \text{and} \quad \Sigma_{\boldsymbol{\theta}} = \begin{pmatrix} s_1^2 & \rho s_1 s_2 \\ \rho s_1 s_2 & s_2^2 \end{pmatrix}$$

with $s_1 = \theta_3^2, s_2 = \theta_4^2$, $\rho = \tanh(\theta_5)$, and $n = 4$.



Local Enhancements

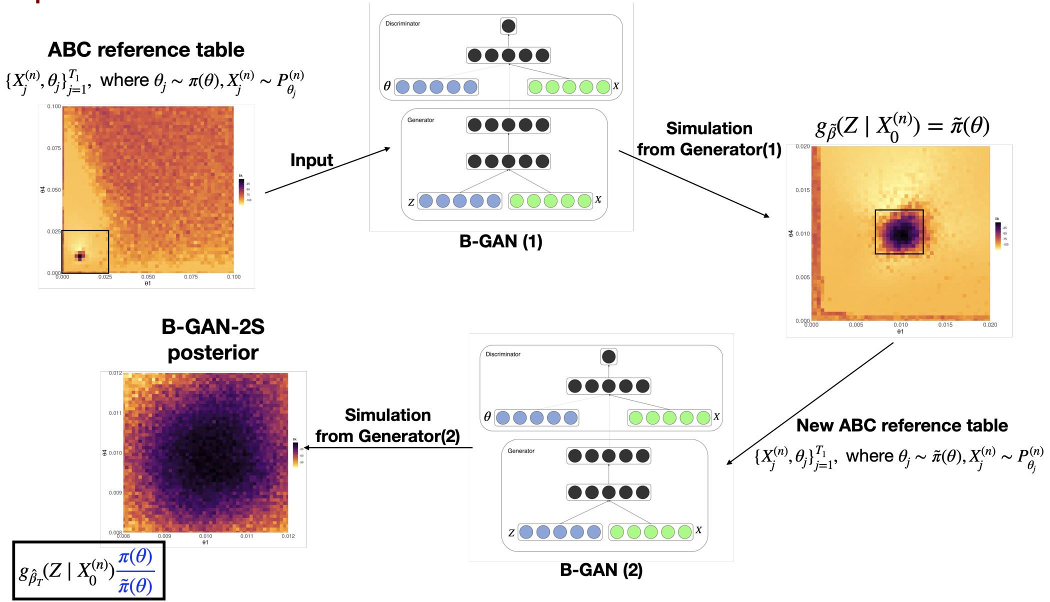
Our goal is to find a high-quality approximation to the conditional $\pi(\theta \mid X_0^{(n)})$, which is not necessarily uniformly over the entire domain $\mathcal{X}$.

The vanilla B-GAN is not trained on the observed data $X_0^{(n)}$.

Can we do better? Yes!

- $X_0^{(n)}$ in proposal $\Rightarrow$ 2-Step Refinement
- $X_0^{(n)}$ in training $\Rightarrow$ Adversarial Variational Bayes

2-Step Refinement



B-GAN-2S

If we **zoom in** the area close to θ_0, the precision of $g(\cdot)$ around $X_0^{(n)}$ can be improved.

- A pilot generator $g_{\hat{\beta}}(Z, X_0^{(n)})$ learned under the original prior $\pi(\theta)$ can be used to guide the "promising" region for the next round $\Rightarrow \tilde{\pi}(\theta)$.
- We adjust the "wrong" prior by reweighting with importance weights

$$r(\theta) = \frac{\pi(\theta)}{\tilde{\pi}(\theta)}.$$

- The new B-GAN returns weighted approximated posterior sample pairs $\left(\tilde{\theta}_1, \hat{r}(\tilde{\theta}_1)\right), \ldots, \left(\tilde{\theta}_M, \hat{r}(\tilde{\theta}_M)\right)$.

Adversarial Variational Approximation

Implicit distributions are explored within the variational framework to obtain finer and tighter posterior approximations.

Find a set of parameter $\boldsymbol{\beta}^*$ that maximizes the Evidence Lower Bound (ELBO) as

$$\beta^* = \arg\max_\beta \; \mathcal{L}(\beta) = \arg\min_\beta \; KL\left(q_\beta\left(\theta \mid X_0^{(n)}\right) \| \pi\left(\theta \mid X_0^{(n)}\right)\right)$$

$$= \arg\min_\beta \; P_{\theta \sim q_\beta\left(\theta | X_0^{(n)}\right)} \log \frac{q_\beta\left(\theta | X_0^{(n)}\right)}{\pi\left(\theta | X_0^{(n)}\right)}$$

simulation from $g_\beta\left(Z, X_0^{(n)}\right)$

implicit $\Rightarrow$ classification trick $\log \frac{d_\beta^*\left(X_0^{(n)}, \theta\right)}{1 - d_\beta^*\left(X_0^{(n)}, \theta\right)}$

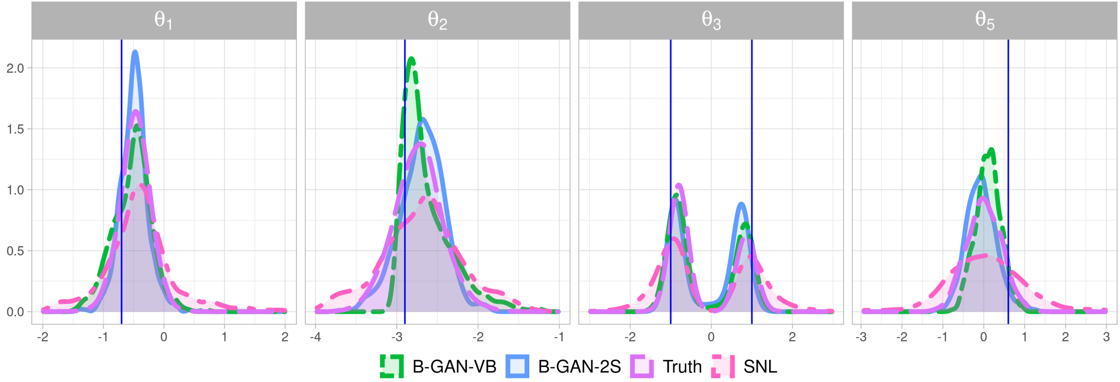
$\Rightarrow$ replace d_β^* with $\widehat{d_\beta}$

$\Rightarrow$ train d_β on $\pi\left(X^{(n)}, \theta\right)$ and $\pi_g\left(X^{(n)}, \theta\right)$

Discriminator learns globally, generator learns locally!

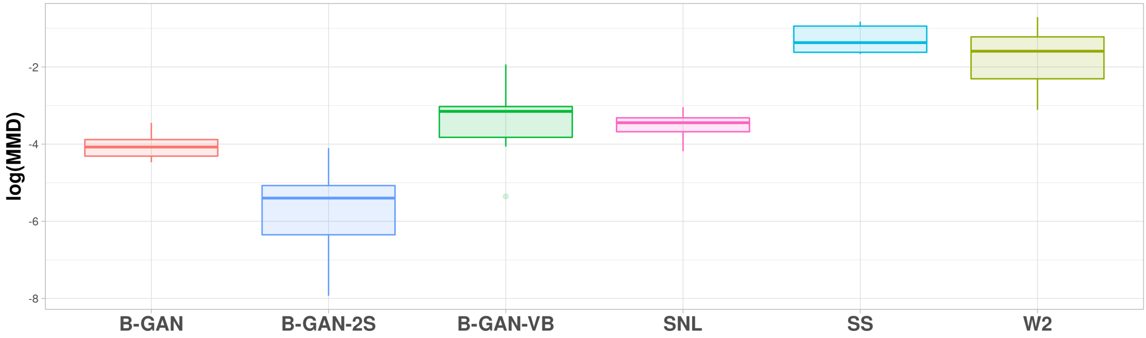
Toy Example (cont'd)

B-GAN-2S and B-GAN-VB have smaller biases and tighter credible regions.



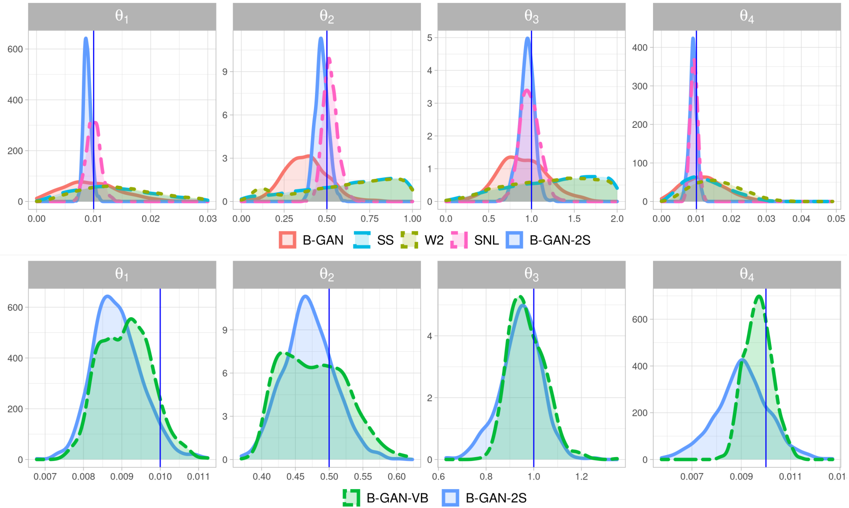
Toy Example (cont'd)

We report the Maximum Mean Discrepancies (MMDs) between the approximated posteriors and the true posterior.



Lotka-Volterra Revisited

We follow the same setup as before, except that now $X_0^{(n)}$ is a single time-series.



Conclusions: Part II

- We propose a Bayesian GAN sampler
  - ☺ can be applied to dependent data and/or very few observations
  - ☺ convergence results in terms in total variation distance is provided
- Two types of local performance enhancements are considered
  - 2-Step Refinement
  - Adversarial Variational Bayes

Thank you!

AR-ABC: Posterior Concentration

Our results can be viewed as a special case of Frazier et al. [2018].

Theorem. Under some mild assumptions, as $n \to \infty$ and $\epsilon_n = o(1)$ and $C_n \delta_n = o(\epsilon_n)$ for some arbitrarily slowly increasing sequences $M_n, C_n > 0$,

$$P_0^{(n)} \Pi[K(p_{\theta_0}, p_\theta) > \lambda_n \mid \hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^\theta) \leq \epsilon_n] = o(1),$$

where

$$\lambda_n = M_n C_n \underbrace{\delta_n}_{|\hat{K} - K_n|} \epsilon_n^{-\kappa} + \sqrt{M_n} n^{-1/2} \epsilon_n^{-\kappa/2} + \epsilon_n \tag{4}$$

Using Kaji et al. [2020] we show

$$\left| \hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^\theta) - \frac{1}{n} \sum_{i=1}^{n} \log \frac{p_{\theta_0}}{p_\theta}(X_i) \right| = O_p(\delta_n).$$

The rate δ_n depends on the choice of the discriminator, smoothness of the model, and the dimension of the data space d.

AR-ABC: Posterior Concentration

Our results can be viewed as a special case of Frazier et al. [2018].

Theorem. Under some mild assumptions, as $n \to \infty$ and $\epsilon_n = o(1)$ and $C_n \delta_n = o(\epsilon_n)$ for some arbitrarily slowly increasing sequences $M_n, C_n > 0$,

$$P_0^{(n)} \Pi[K(p_{\theta_0}, p_\theta) > \lambda_n \mid \hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^\theta) \leq \epsilon_n] = o(1),$$

where

$$\lambda_n = M_n C_n \underbrace{\delta_n}_{|\hat{K} - K_n|} \epsilon_n^{-\kappa} + \sqrt{M_n} \underbrace{n^{-1/2}}_{|K_n - K|} \epsilon_n^{-\kappa/2} + \epsilon_n \tag{5}$$

Estimation error between the empirical KL and true KL.

AR-ABC: Posterior Concentration

Our results can be viewed as a special case of Frazier et al. [2018].

Theorem. Under some mild assumptions, as $n \to \infty$ and $\epsilon_n = o(1)$ and $C_n \delta_n = o(\epsilon_n)$ for some arbitrarily slowly increasing sequences $M_n, C_n > 0$,

$$P_0^{(n)} \Pi[K(p_{\theta_0}, p_\theta) > \lambda_n \mid \hat{K}(\boldsymbol{X}, \widetilde{\boldsymbol{X}}^\theta) \leq \epsilon_n] = o(1),$$

where

$$\lambda_n = M_n C_n \underbrace{\delta_n}_{|\hat{K}-K_n|} \epsilon_n^{-\kappa} + \sqrt{M_n} \underbrace{n^{-1/2}}_{|K_n - K|} \epsilon_n^{-\kappa/2} + \underbrace{\epsilon_n}_{\text{threshold}} . \quad (6)$$

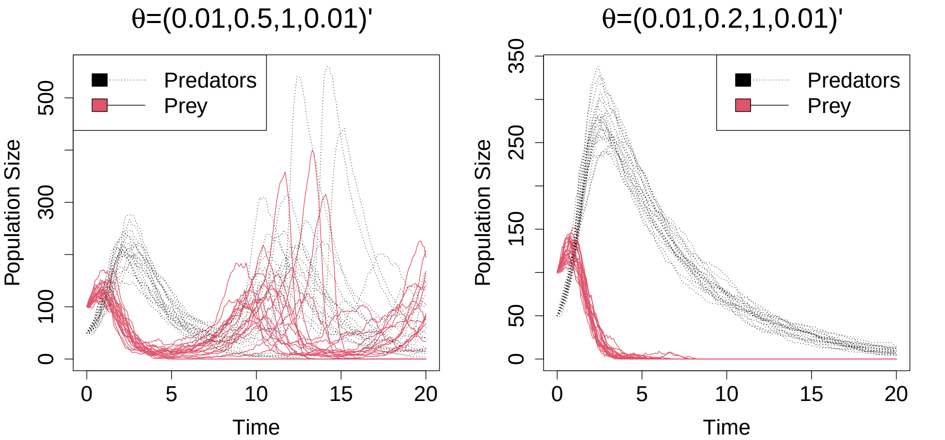
- ☺ Consistency is guaranteed as long as $\epsilon_n \to 0$
- ☹ The proper choice of ϵ_n is unclear for complex models

Other ABC Methods

- (CA) Classification Accuracy [Gutmann et al., 2018]
- (W2) 2-Wasserstein distance [Bernton et al., 2019]
- (SS) ℓ_2-distance between summary statistics and we use the (SA) semi-automatic method [Fearnhead and Prangle, 2012] if no candidate summary statistics are given
- (DNN) approximated posterior mean of the parameters predicted by trained deep neural network [Jiang et al., 2017]
- (MM) Maximum Mean Discrepancy [Park et al., 2016]
- (ES) Energy Statistics [Nguyen et al., 2020]
- (AL) Auxiliary Likelihood [Drovandi et al., 2011]

For each ABC method, we ran 100,000 samples and accepted the top 1%.

Lotka-Volterra: A Closer Look



- Patterns in the predator-prey interactions are very sensitive to parameter changes

B-GAN-VB

We implement the Wasserstein analogue.

The generator function is updated only locally on $X_0^{(n)}$.

- Update the critic function f_ω globally on $\{\theta_j, X_j^{(n)}\}_{j=1}^{T}$

$$\max_{\omega: f_\omega \in \mathcal{F}} \mathbb{P}_{X^{(n)} \sim \pi(X^{(n)}), Z \sim \pi_Z} f_\omega(g_\beta(Z, X^{(n)}), X^{(n)}) - \mathbb{P}_{(\theta, X^{(n)}) \sim \pi(\theta, X^{(n)})} f_\omega(\theta, X^{(n)}).$$

- Update the generator g_β locally on $X_0^{(n)}$

$$\min_{\beta: g_\beta \in \mathcal{G}} \mathbb{P}_{Z \sim \pi_Z} f_\omega(g_\beta(Z, X_0^{(n)}), X_0^{(n)}).$$