Variational Gibbs Inference for Statistical Model Estimation from Incomplete Data

Vaidotas Šimkus Ben Rhodes Michael Gutmann

School of Informatics
The University of Edinburgh

November 2023

Topic of the talk

THE UNIVERSITY of EDINBURGH informatics

Variational Gibbs Inference for Statistical Model Estimation from Incomplete Data

- General-purpose method for estimating statistical models from incomplete data.
- Journal of Machine Learning Research, 2023: jmlr.org/papers/v24/21-1373.html.
- Code: github.com/vsimkus/variational-gibbs-inference.
- Demo: nbviewer.org/github/vsimkus/variational-gibbs-inference/blob/main/notebooks/VGI_demo.ipynb.

Overview

1. Statistical models and the missing data issue
2. Some problems with direct estimation from incomplete data
3. Variational Gibbs Inference

Some modern statistical models

- Normalising flows

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = p(\boldsymbol{u}) \left|\det J_{T_{\boldsymbol{\theta}}}\right|^{-1},$$

$$\boldsymbol{x} = T_{\boldsymbol{\theta}}(\boldsymbol{u}), \quad T_{\boldsymbol{\theta}} = T_{\boldsymbol{\theta}}^{L} \circ \cdots \circ T_{\boldsymbol{\theta}}^{1},$$

  - $p(\boldsymbol{u})$ is a simple base distribution.
  - $T_{\boldsymbol{\theta}}^{l}$ are deterministic, invertible, and differentiable.

$p_{\boldsymbol{\theta}}(\boldsymbol{x}) =$ 

Image credit: *[Durkan et al., 2019]*

- Variational autoencoders (VAEs)

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \int p_{\boldsymbol{\theta}}(\boldsymbol{x} \mid \boldsymbol{z}) p_{\boldsymbol{\theta}}(\boldsymbol{z}) \, \mathrm{d}\boldsymbol{z}$$

  - $p_{\boldsymbol{\theta}}(\boldsymbol{z})$ is often a simple distribution such as standard Gaussian.
  - $p_{\boldsymbol{\theta}}(\boldsymbol{x} \mid \boldsymbol{z})$ is a simple distribution (e.g. Gaussian or Multinomial), parametrised via a neural network.

 $\sim p_{\boldsymbol{\theta}}(\boldsymbol{x})$

Image credit: *[Child, 2021]*

The missing data issue

THE UNIVERSITY of EDINBURGH informatics

- The models $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ are specified for fully-observed data $\boldsymbol{x}$,
- And are typically fitted via maximum-likelihood estimation (MLE)

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}^i), \quad \text{where} \quad \boldsymbol{x}^i \in \mathcal{D}.$$

- Real-world data is often incomplete due to: non-response, sensor failure, occlusion, etc.
- What can we do?
- Denote $\boldsymbol{x}_{\mathrm{o}}$ and $\boldsymbol{x}_{\mathrm{m}}$ as the observed and missing elements of $\boldsymbol{x} = \boldsymbol{x}_{\mathrm{o}} \cup \boldsymbol{x}_{\mathrm{m}}$ (with $\boldsymbol{x}_{\mathrm{m}} \cap \boldsymbol{x}_{\mathrm{o}} = \varnothing$).

Options:

1. Discard data-points with missing values → loss of information, not sustainable, bias ✗
2. Impute-then-fit → selecting appropriate imputation method, imputation incongeniality ✗
3. Direct fitting by marginalising the missing variables $\boldsymbol{x}_{\mathrm{m}}$?

Overview

1. Statistical models and the missing data issue
2. Some problems with direct estimation from incomplete data
3. Variational Gibbs Inference

Expectation Maximisation (EM)

- Marginalising the missing variables $\int p_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathrm{o}}, \boldsymbol{x}_{\mathrm{m}})\,\mathrm{d}\boldsymbol{x}_{\mathrm{m}}$ is generally not tractable.
- What can we do if simplifying assumptions cannot be inserted?
- Expectation-maximisation (EM) (assuming ignorable missingness)

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathrm{o}}) = \log \int f(\boldsymbol{x}_{\mathrm{m}} \mid \boldsymbol{x}_{\mathrm{o}}) \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathrm{o}}, \boldsymbol{x}_{\mathrm{m}})}{f(\boldsymbol{x}_{\mathrm{m}} \mid \boldsymbol{x}_{\mathrm{o}})}\,\mathrm{d}\boldsymbol{x}_{\mathrm{m}} \geqslant \mathbb{E}_{f(\boldsymbol{x}_{\mathrm{m}}|\boldsymbol{x}_{\mathrm{o}})}\left[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathrm{o}}, \boldsymbol{x}_{\mathrm{m}})}{f(\boldsymbol{x}_{\mathrm{m}} \mid \boldsymbol{x}_{\mathrm{o}})}\right], \quad \text{“ELBO”}$$

  - **E-step:** Maximise w.r.t. $f(\boldsymbol{x}_{\mathrm{m}} \mid \boldsymbol{x}_{\mathrm{o}}^i)$ for $\forall \boldsymbol{x}_{\mathrm{o}}^i \in \mathcal{D}$: $f(\boldsymbol{x}_{\mathrm{m}} \mid \boldsymbol{x}_{\mathrm{o}}^i) = p_{\boldsymbol{\theta}^t}(\boldsymbol{x}_{\mathrm{m}} \mid \boldsymbol{x}_{\mathrm{o}}^i)$.
  - **M-step:** Maximise w.r.t. $\boldsymbol{\theta}$: $\boldsymbol{\theta}^{t+1} = \arg\max_{\boldsymbol{\theta}} \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{p_{\boldsymbol{\theta}^t}(\boldsymbol{x}_{\mathrm{m}}|\boldsymbol{x}_{\mathrm{o}}^i)}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathrm{o}}^i, \boldsymbol{x}_{\mathrm{m}})\right]$
- Monte Carlo EM: Approximate the expectation using Monte Carlo average.
- Then, M-step corresponds to fitting $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ with completed data.

Issue with Monte Carlo EM

- Conditional sampling of $p_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathrm{m}} \mid \boldsymbol{x}_{\mathrm{o}})$ is generally intractable or inefficient.

Variational approximation to $p_{\boldsymbol{\theta}^t}(\boldsymbol{x}_\mathrm{m} \mid \boldsymbol{x}_\mathrm{o})$

Variational inference (VI)

- $\forall \boldsymbol{x}_\mathrm{o} \in \mathcal{D}$ specify a $f_{\boldsymbol{\phi}}(\boldsymbol{x}_\mathrm{m} \mid \boldsymbol{x}_\mathrm{o}) \in \mathcal{Q}(\boldsymbol{\phi})$.
- E-step: Maximise the ELBO w.r.t. $\boldsymbol{\phi}$.
- M-step: Sample $f_{\boldsymbol{\phi}}(\boldsymbol{x}_\mathrm{m} \mid \boldsymbol{x}_\mathrm{o})$ to approximate the expectation.

Amortised VI

- Parametrise $f_{\boldsymbol{\phi}}(\boldsymbol{x}_\mathrm{m} \mid \boldsymbol{x}_\mathrm{o})$ with a *single* neural network $\mathrm{NN}_{\boldsymbol{\phi}}(\boldsymbol{x}_\mathrm{o})$ for $\forall \boldsymbol{x}_\mathrm{o} \in \mathcal{D}$.

	d_1	d_2	d_3	d_4	$f_{\boldsymbol{\phi}}(\boldsymbol{x}^i_\mathrm{m} \mid \boldsymbol{x}^i_\mathrm{o})$
$\boldsymbol{x}^1$	x^1_1	?	x^1_3	x^1_4	$f_{\boldsymbol{\phi}}(x^1_2 \mid x^1_1, x^1_3, x^1_4)$
$\boldsymbol{x}^2$	?	x^2_2	x^2_3	?	$f_{\boldsymbol{\phi}}(x^2_1, x^2_4 \mid x^2_2, x^2_3)$
$\boldsymbol{x}^3$	?	?	?	x^3_4	$f_{\boldsymbol{\phi}}(x^3_1, x^3_2, x^3_3 \mid x^3_4)$
$\vdots$		$\vdots$			$\vdots$

Advantages of VI

- Choice of $\mathcal{Q}(\boldsymbol{\phi})$ is in our control.
- Turns inference to optimisation.
- Can fit using SGD.
- Efficient if $|\mathcal{D}|$ is small.

Disadvantages of VI

- Is inefficient if $|\mathcal{D}|$ is large.

Advantages of amortised VI

- Efficient for large $|\mathcal{D}|$.

Disadvantages of amortised VI

- Need one $f_{\boldsymbol{\phi}}(\boldsymbol{x}_\mathrm{m} \mid \boldsymbol{x}_\mathrm{o})$ for each pattern of missingness (2^M in total).

Overview

1. Statistical models and the missing data issue
2. Some problems with direct estimation from incomplete data
3. Variational Gibbs Inference

Variational Gibbs Inference: Core idea

THE UNIVERSITY *of* EDINBURGH informatics

Variational Gibbs Inference for Statistical Model Estimation from Incomplete Data, JMLR, 2023

- General-purpose method for estimating $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ from incomplete data.
- Efficient for large $|\mathcal{D}|$ and mitigates the need for 2^M conditional distributions.

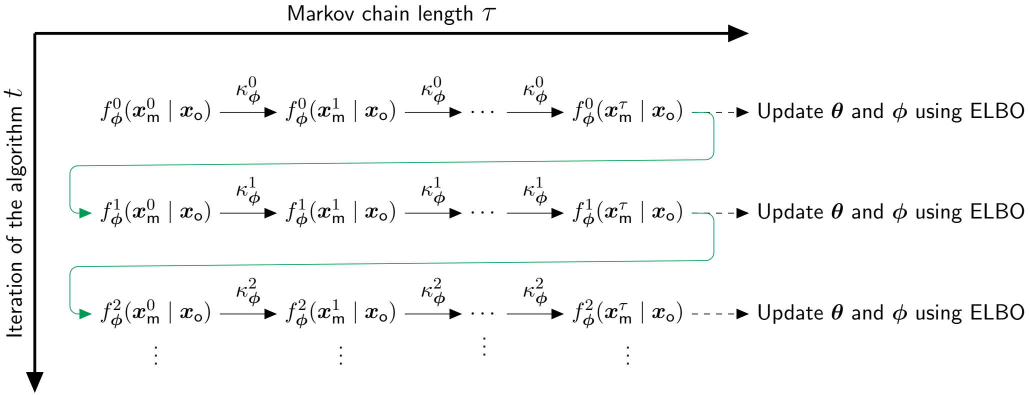
1. Core idea: Turn the 2^M conditional distribution problem into M conditional distributions.
2. To make $f^t_{\boldsymbol{\phi}}(\boldsymbol{x}_\mathrm{m} \mid \boldsymbol{x}_\mathrm{o})$ flexible:
   - Specify it to be the marginal of a Markov chain with a *learnable* kernel $\kappa_{\boldsymbol{\phi}}(\boldsymbol{x}^{\tau+1}_\mathrm{m} \mid \boldsymbol{x}_\mathrm{o}, \boldsymbol{x}^\tau_\mathrm{m})$.
3. To address the 2^M pattern problem:
   - We specify the kernel to be Gibbs (updates one dimension of $\boldsymbol{x}_\mathrm{m}$ at a time):

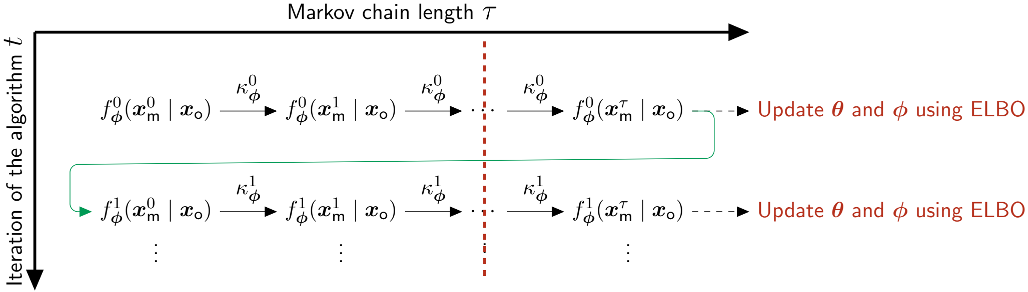
   $$\kappa_{\boldsymbol{\phi}}(\boldsymbol{x}^{\tau+1}_\mathrm{m} \mid \boldsymbol{x}^\tau_\mathrm{m}, \boldsymbol{x}_\mathrm{o}) = \mathbb{E}_{\boldsymbol{\pi}(j \mid \mathrm{idx}(\boldsymbol{m}))}\left[q_{\boldsymbol{\phi}_j}(x_j \mid \boldsymbol{x}^\tau_{\mathrm{m}\setminus j}, \boldsymbol{x}_\mathrm{o})\delta(\boldsymbol{x}^{\tau+1}_{\mathrm{m}\setminus j} - \boldsymbol{x}^\tau_{\mathrm{m}\setminus j})\right],$$

   where $\boldsymbol{\pi}(j \mid \mathrm{idx}(\boldsymbol{m}))$ is the selection probability for the j-th dimension of a Gibbs sampler.
   - Hence we have to learn only M variational Gibbs conditional $q_{\boldsymbol{\phi}_j}(x_j \mid \boldsymbol{x}_{\mathrm{m}\setminus j}, \boldsymbol{x}_\mathrm{o})$.

Variational Gibbs Inference: Persistent chains

- Sampling long Markov chains at each iteration t of the algorithm is costly.
- Use "persistent" chains: initialise the chains at the last state of the previous iteration.
- Can now use short chains, that is using small τ, at every iteration t.





- Computing the marginal density $f^t_\phi(x^\tau_m \mid x_o)$ of a Markov chain remains intractable:

$$f^t_\phi(x^\tau_m \mid x_o) = \int f^t_\phi(x^0_m \mid x_o) \prod_{h=0}^{\tau-1} \kappa_\phi(x^{h+1}_m \mid x_o, x^h_m)\, \mathrm{d}x^0_m \ldots \mathrm{d}x^{\tau-1}_m.$$

- So how can we optimise the parameters ϕ of the kernel κ_ϕ?
- Instead of optimising ϕ over the full length of the Markov chains, we "cut" the chains just before the last transition and optimise over the last step of the chain.

- Objective for learning $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$:

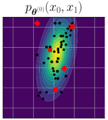
$$\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathsf{o}}) \geqslant \mathbb{E}_{\boldsymbol{\pi}(j|\mathrm{idx}(\boldsymbol{m}))f^{t-1}(\boldsymbol{x}_{\mathsf{m}\setminus j}|\boldsymbol{x}_{\mathsf{o}})q_{\boldsymbol{\phi}_j}(x_j|\boldsymbol{x}_{\mathsf{m}\setminus j},\boldsymbol{x}_{\mathsf{o}})}\left[\log \frac{p_{\boldsymbol{\theta}}(x_j, \boldsymbol{x}_{\mathsf{m}\setminus j}, \boldsymbol{x}_{\mathsf{o}})}{q_{\boldsymbol{\phi}_j}(x_j \mid \boldsymbol{x}_{\mathsf{m}\setminus j}, \boldsymbol{x}_{\mathsf{o}})}\right] + \text{Const.}$$

- We only need samples from penultimate step of the Markov chain f^{t-1}.
- Can optimise w.r.t. $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ using stochastic gradient ascent.
- Maximising the above w.r.t. $\boldsymbol{\phi}$ corresponds to minimising the KL divergence:

$$\mathbb{E}_{\boldsymbol{\pi}(j|\mathrm{idx}(\boldsymbol{m}))f^{t-1}(\boldsymbol{x}_{\mathsf{m}\setminus j}|\boldsymbol{x}_{\mathsf{o}})}\left[D_{\mathsf{KL}}(q_{\boldsymbol{\phi}_j}(x_j \mid \boldsymbol{x}_{\mathsf{m}\setminus j}, \boldsymbol{x}_{\mathsf{o}}) \,||\, p_{\boldsymbol{\theta}}(x_j \mid \boldsymbol{x}_{\mathsf{m}\setminus j}, \boldsymbol{x}_{\mathsf{o}}))\right]$$

- The fitted $\kappa_{\boldsymbol{\phi}}$ approximates the Gibbs kernel with the stationary distribution $p_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathsf{m}} \mid \boldsymbol{x}_{\mathsf{o}})$.

Variational Gibbs Inference: Algorithm

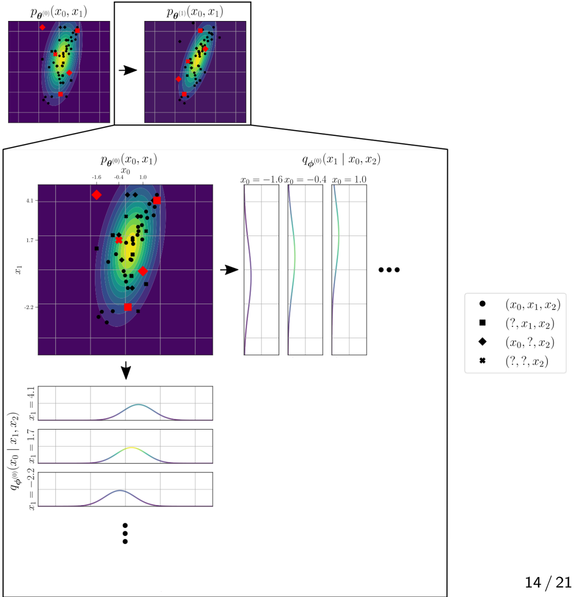


Algorithm 1 Variational Gibbs inference

1: **Create** K-times imputed data $\mathcal{D}_K$ using f_0

Algorithm 1 Variational Gibbs inference

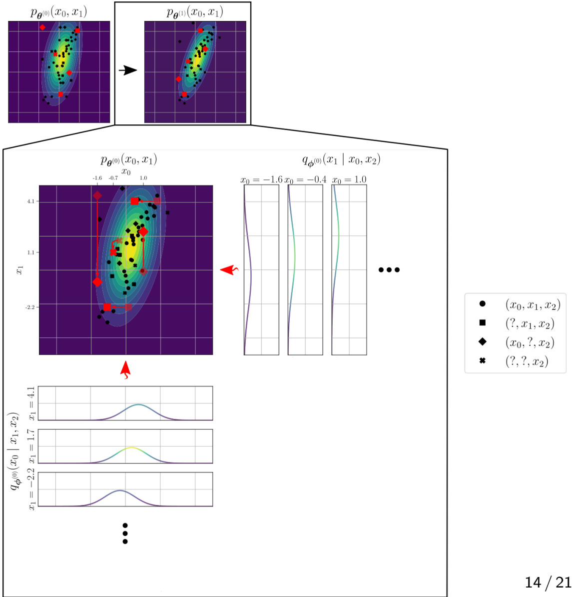
1: **Create** K-times imputed data $\mathcal{D}_K$ using f_0
2: **for** t in $[1, \mathit{max\_epochs}]$ **do**
3: **Sample** mini-batch $\mathcal{B}_K$ from $\mathcal{D}_K$
7: **end for**



Variational Gibbs Inference: Algorithm

Algorithm 1 Variational Gibbs inference

Create K-times imputed data $\mathcal{D}_K$ using f_0
for t in $[1, \textit{max\_epochs}]$ **do**
  Sample mini-batch $\mathcal{B}_K$ from $\mathcal{D}_K$
  Update the imputations in $\mathcal{B}_K$:
  $\bar{\boldsymbol{x}}_{\mathrm{m}}^{(i,k)} \sim \mathrm{Gibbs}_{\tau}(\boldsymbol{x}_{\mathrm{o}}^{i}, \kappa_{\boldsymbol{\phi}}; \boldsymbol{x}_{\mathrm{m}}^{(i,k)}), \forall \boldsymbol{x}_{\mathrm{m}}^{(i,k)} \in \mathcal{B}_K$
  Persist the imputations in $\mathcal{B}_K$ to $\mathcal{D}_K$
end for

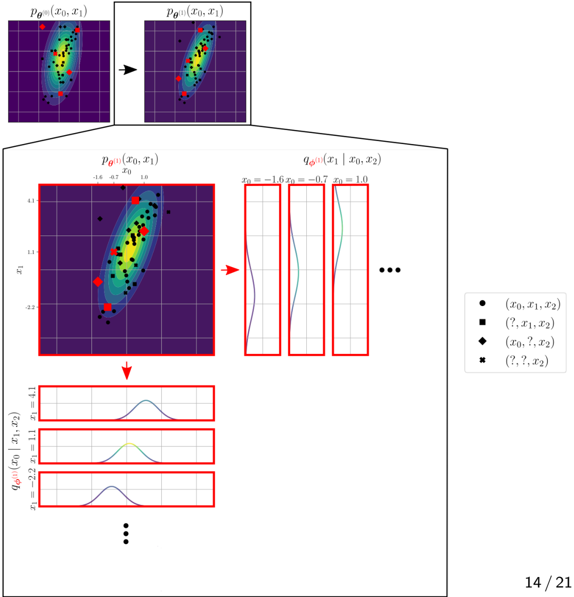


Variational Gibbs Inference: Algorithm

THE UNIVERSITY of EDINBURGH
informatics

Algorithm 1 Variational Gibbs inference

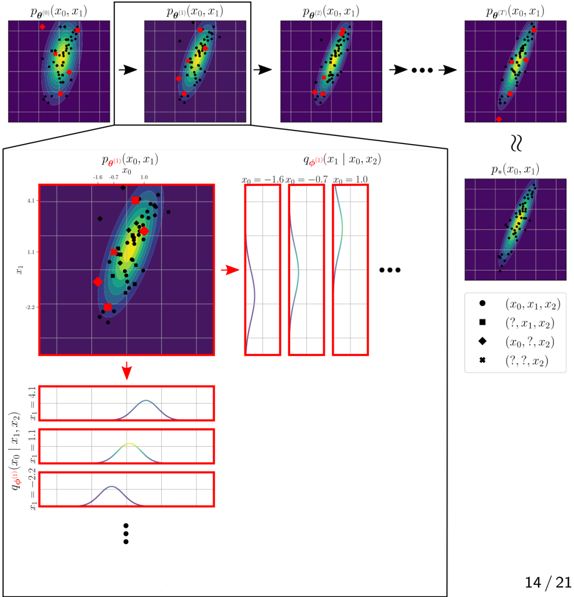
Create K-times imputed data $\mathcal{D}_K$ using f_0
for t in $[1, \mathit{max\_epochs}]$ **do**
  Sample mini-batch $\mathcal{B}_K$ from $\mathcal{D}_K$
  Update the imputations in $\mathcal{B}_K$:
$\bar{\boldsymbol{x}}_{\mathrm{m}}^{(i,k)} \sim \mathrm{Gibbs}_\tau(\boldsymbol{x}_{\mathrm{o}}^{i}, \kappa_{\boldsymbol{\phi}}; \boldsymbol{x}_{\mathrm{m}}^{(i,k)}), \forall \boldsymbol{x}_{\mathrm{m}}^{(i,k)} \in \mathcal{B}_K$
  Persist the imputations in $\mathcal{B}_K$ to $\mathcal{D}_K$
  Update $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ with SGA.
end for



Variational Gibbs Inference: Algorithm

Algorithm 1 Variational Gibbs inference

Create K-times imputed data $\mathcal{D}_K$ using f_0
for t in $[1, \mathit{max\_epochs}]$ **do**
  Sample mini-batch $\mathcal{B}_K$ from $\mathcal{D}_K$
  Update the imputations in $\mathcal{B}_K$:
$\bar{\boldsymbol{x}}_{\mathrm{m}}^{(i,k)} \sim \mathrm{Gibbs}_{\tau}(\boldsymbol{x}_{\mathrm{o}}^{i}, \kappa_{\phi}; \boldsymbol{x}_{\mathrm{m}}^{(i,k)}), \forall \boldsymbol{x}_{\mathrm{m}}^{(i,k)} \in \mathcal{B}_K$
  Persist the imputations in $\mathcal{B}_K$ to $\mathcal{D}_K$
  Update $\boldsymbol{\theta}$ and ϕ with SGA.
end for



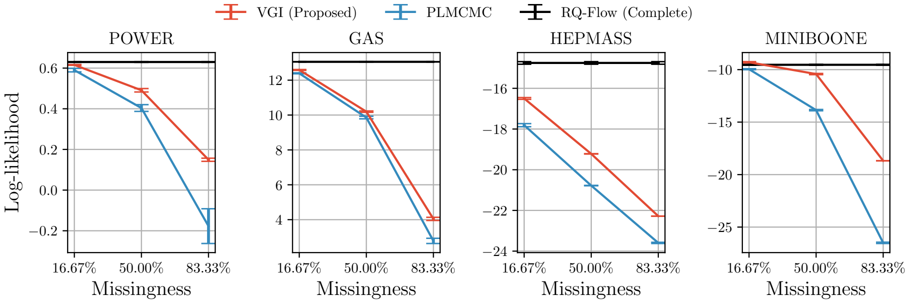
Variational Gibbs Inference: Summary

THE UNIVERSITY *of* EDINBURGH informatics

- Direct fitting by (approximately) marginalising the missing variables $\boldsymbol{x}_\mathrm{m}$ ✓

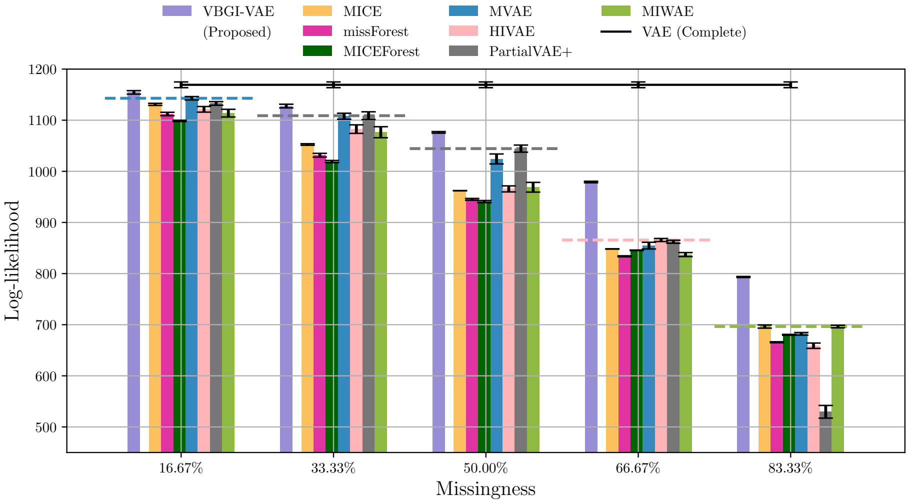
- General-purpose method for estimating $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ from incomplete data.
- Mitigated the need for 2^M conditional distributions to just M by representing the variational distribution via a learnable Gibbs kernel.
- Used "persistent" chains to efficiently sample imputations using the learnt Gibbs kernel.
- "Cut" the Markov chains to make optimisation of ϕ efficient.

Variational Gibbs Inference: Results (Flows)



	POWER	GAS	HEPMASS	MINIBOONE
Model parameters	∼2M	∼2M	∼1M	∼129K
Dimensionality	6	8	21	43

Variational Gibbs Inference: Results (VAE)



Model parameters: ∼682K. Dimensionality: 560.

Summary

- Statistical models and the missing data issue.
  - Modern models, such as normalising flows and VAEs, are very flexible.
  - But, they are formulated for complete data.
- Some problems with direct estimation from incomplete data.
  - Marginalisation $\int p_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathsf{o}}, \boldsymbol{x}_{\mathsf{m}})\, \mathrm{d}\boldsymbol{x}_{\mathsf{m}}$ is generally intractable.
  - EM algorithm requires sampling conditionals $p_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathsf{m}} \mid \boldsymbol{x}_{\mathsf{o}})$ for $\forall \boldsymbol{x}_{\mathsf{o}} \in \mathcal{D}$, which is expensive.
  - Standard amortised VI requires 2^M variational distributions, which is inefficient.
- Variational Gibbs Inference.
  - General purpose method for model estimation from incomplete data.
  - Achieves good performance on normalising flow and VAE estimation, compared to other methods.

Thank you for listening.
Questions?

References I

Child, R. (2021). Very Deep VAEs Generalize Autoregressive Models and Can Outperform Them on Images. In *International Conference on Learning Representations (ICLR)*. (Cited on slide 4)

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22. (Cited on slide 7)

Durkan, C., Bekasov, A., Murray, I., and Papamakarios, G. (2019). Neural Spline Flows. In *Advances in Neural Information Processing Systems (NeurIPS)*. (Cited on slide 4)

Kingma, D. P. and Welling, M. (2013). Auto-Encoding Variational Bayes. In *International Conference on Learning Representations (ICLR)*. (Cited on slide 4)

Rezende, D. J. and Mohamed, S. (2015). Variational inference with normalizing flows. In *International Conference on Machine Learning (ICML)*. (Cited on slide 4)

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic Backpropagation and Approximate Inference. In *International Conference on Machine Learning (ICML)*, Beijing, China. (Cited on slide 4)

Simkus, V., Rhodes, B., and Gutmann, M. U. (2023). Variational Gibbs Inference for Statistical Model Estimation from Incomplete Data. *Journal of Machine Learning Research*, 24(196):1–72. (Cited on slide 2, 10)

References II

Tieleman, T. (2008). Training restricted Boltzmann machines using approximations to the likelihood gradient. In *International Conference on Machine Learning (ICML)*, pages 1064–1071. (Cited on slide 11)

Wei, G. C. G. and Tanner, M. A. (1990). A Monte Carlo Implementation of the EM Algorithm and the Poor Man's Data Augmentation Algorithms. *Journal of the American Statistical Association*, 85(411):699–704. (Cited on slide 7)