Scaling Bayesian inference of mixed multinomial logit models to large datasets

Filipe Rodrigues

Technical University of Denmark

MLSM
Machine Learning for Smart Mobility group
http://mlsm.man.dtu.dk

May 2022

Motivation

- Full Bayesian treatment of choice models has several advantages over maximum likelihood estimation
  - Obtain full posterior distributions over the model parameters (including the individual-specific taste parameters)
  - Handle incomplete data by marginalizing over missing variables
  - Natural support for online inference for streaming data
  - Support for automatic utility function specification approaches[1]

[1] Rodrigues, F., Ortelli, N., Bierlaire, M. and Pereira, F.C., 2020. Bayesian automatic relevance determination for utility function specification in discrete choice models. IEEE Transactions on Intelligent Transportation Systems.

Motivation

- Full Bayesian treatment of choice models has several advantages over maximum likelihood estimation
  - Obtain full posterior distributions over the model parameters (including the individual-specific taste parameters)
  - Handle incomplete data by marginalizing over missing variables
  - Natural support for online inference for streaming data
  - Support for automatic utility function specification approaches[1]
- MCMC methods carry extremely high computational costs (both in terms of time and storage)
- Variational inference (VI) can provide significant improvements in computational efficiency (see Bansal et al. (2020) and Tan (2017))

[1] Rodrigues, F., Ortelli, N., Bierlaire, M. and Pereira, F.C., 2020. Bayesian automatic relevance determination for utility function specification in discrete choice models. IEEE Transactions on Intelligent Transportation Systems.

Motivation

- Full Bayesian treatment of choice models has several advantages over maximum likelihood estimation
  - Obtain full posterior distributions over the model parameters (including the individual-specific taste parameters)
  - Handle incomplete data by marginalizing over missing variables
  - Natural support for online inference for streaming data
  - Support for automatic utility function specification approaches[1]
- MCMC methods carry extremely high computational costs (both in terms of time and storage)
- Variational inference (VI) can provide significant improvements in computational efficiency (see Bansal et al. (2020) and Tan (2017))
- However, several limitations remain:
  - Scalability to large datasets
  - Difficulty in using “modern” priors
  - Lack of flexibility to capture highly complex posteriors

---

[1] Rodrigues, F., Ortelli, N., Bierlaire, M. and Pereira, F.C., 2020. Bayesian automatic relevance determination for utility function specification in discrete choice models. IEEE Transactions on Intelligent Transportation Systems.

TL; DR

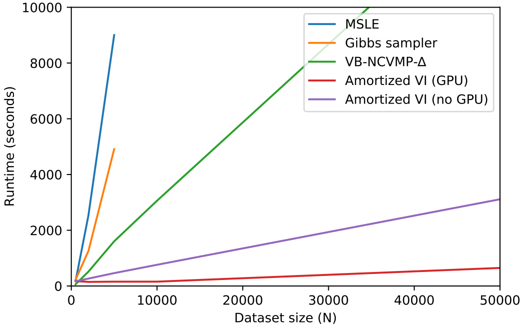


Figure: Scalability plot comparing the proposed Amortized VI approach with MSLE, Gibbs sampling and the VB-NCVMP-Δ approach from Bansal et al. (2020)

Bayesian mixed logit model

Generative process of the Bayesian mixed logit model considered:

1. Draw fixed taste parameters $\boldsymbol{\alpha} \sim \mathcal{N}(\boldsymbol{\lambda}_0, \boldsymbol{\Xi}_0)$
2. Draw mean vector $\boldsymbol{\zeta} \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$
3. Draw scales vector $\boldsymbol{\tau} \sim \text{half-Cauchy}(\boldsymbol{\sigma}_0)$
4. Draw correlation matrix $\boldsymbol{\Psi} \sim \text{LKJ}(\nu)$
5. For each decision-maker $n \in \{1, \dots, N\}$
   (a) Draw random taste parameters $\boldsymbol{\beta}_n \sim \mathcal{N}(\boldsymbol{\zeta}, \boldsymbol{\Omega})$
   (b) For each choice occasion $t \in \{1, \dots, T_n\}$
      (i) Draw observed choice $y_{nt} \sim \text{MNL}(\boldsymbol{\alpha}, \boldsymbol{\beta}_n, \mathbf{X}_{nt})$

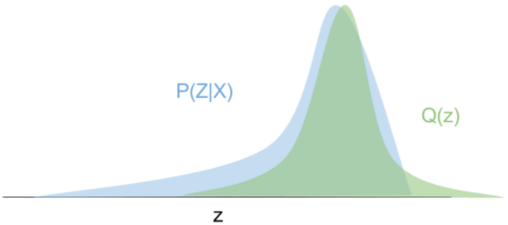
where $\boldsymbol{\Omega} = \text{diag}(\boldsymbol{\tau}) \times \boldsymbol{\Psi} \times \text{diag}(\boldsymbol{\tau})$

Variational inference: basics

- Let $\mathbf{z} = \{\boldsymbol{\alpha}, \boldsymbol{\zeta}, \boldsymbol{\tau}, \boldsymbol{\Psi}, \boldsymbol{\beta}_{1:N}\}$ denote the set of all latent variables in the model
- Goal: compute posterior of $\mathbf{z}$ given a dataset of observed choices - $p(\mathbf{z}|\mathbf{y}_{1:N})$

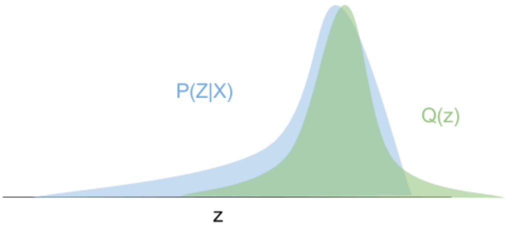
Variational inference: basics

- Let $\mathbf{z} = \{\boldsymbol{\alpha}, \boldsymbol{\zeta}, \boldsymbol{\tau}, \boldsymbol{\Psi}, \boldsymbol{\beta}_{1:N}\}$ denote the set of all latent variables in the model
- Goal: compute posterior of $\mathbf{z}$ given a dataset of observed choices - $p(\mathbf{z}|\mathbf{y}_{1:N})$
- Consider a family of tractable distributions $q_\phi(\mathbf{z}|\mathbf{y})$ parameterized by ϕ
- Find parameters ϕ that make $q_\phi(\mathbf{z}|\mathbf{y})$ as close as possible to $p(\mathbf{z}|\mathbf{y}_{1:N})$



Variational inference: basics

- Let $\mathbf{z} = \{\boldsymbol{\alpha}, \boldsymbol{\zeta}, \boldsymbol{\tau}, \boldsymbol{\Psi}, \boldsymbol{\beta}_{1:N}\}$ denote the set of all latent variables in the model
- Goal: compute posterior of $\mathbf{z}$ given a dataset of observed choices - $p(\mathbf{z}|\mathbf{y}_{1:N})$
- Consider a family of tractable distributions $q_\phi(\mathbf{z}|\mathbf{y})$ parameterized by ϕ
- Find parameters ϕ that make $q_\phi(\mathbf{z}|\mathbf{y})$ as close as possible to $p(\mathbf{z}|\mathbf{y}_{1:N})$



- Measure similarity between distributions using $\mathbb{KL}(q_\phi(\mathbf{z}|\mathbf{y})||p(\mathbf{z}|\mathbf{y}))$
- Minimize $\mathbb{KL}(q_\phi(\mathbf{z}|\mathbf{y})||p(\mathbf{z}|\mathbf{y}))$ w.r.t. ϕ

Variational inference: challenges

- We cannot minimize $\mathbb{KL}(q_\phi(\mathbf{z}|\mathbf{y})||p(\mathbf{z}|\mathbf{y}))$ directly (intractable), but...

$$\mathbb{KL}(q_\phi(\mathbf{z}|\mathbf{y})||p(\mathbf{z}|\mathbf{y})) = -\underbrace{(\mathbb{E}_{q_\phi}[\log p(\mathbf{y},\mathbf{z})] - \mathbb{E}_{q_\phi}[\log q_\phi(\mathbf{z}|\mathbf{y})])}_{\mathcal{L}(q_\phi) \text{ or } \textit{"ELBO"}} + \underbrace{\log p(\mathbf{y})}_{\text{const.}}$$

- Instead, we maximize $\mathcal{L}(q_\phi)$ - referred to as the evidence lower bound (**ELBO**)

Variational inference: challenges

- We cannot minimize $\mathbb{KL}(q_\phi(\mathbf{z}|\mathbf{y})||p(\mathbf{z}|\mathbf{y}))$ directly (intractable), but...

$$\mathbb{KL}(q_\phi(\mathbf{z}|\mathbf{y})||p(\mathbf{z}|\mathbf{y})) = -\underbrace{(\mathbb{E}_{q_\phi}[\log p(\mathbf{y},\mathbf{z})] - \mathbb{E}_{q_\phi}[\log q_\phi(\mathbf{z}|\mathbf{y})])}_{\mathcal{L}(q_\phi) \text{ or } \textit{“ELBO”}} + \underbrace{\log p(\mathbf{y})}_{\text{const.}}$$

- Instead, we maximize $\mathcal{L}(q_\phi)$ - referred to as the evidence lower bound (**ELBO**)
- Challenges:
  1) ELBO is still intractable for many models of interest (including logit models)
  2) Cannot choose arbitrary priors - we rely on conjugate priors for tractability
  3) Number of variational parameters grows linearly with the number of respondents N (assuming a fully-factorized/mean-field approximation)

$$q_\phi(\mathbf{z}|\mathbf{y}) = q(\boldsymbol{\alpha})\, q(\boldsymbol{\zeta})\, q(\boldsymbol{\tau})\, q(\boldsymbol{\Psi}) \prod_{n=1}^{N} q(\boldsymbol{\beta}_n)$$

  4) $q_\phi(\mathbf{z}|\mathbf{y})$ must be sufficiently flexible to accurately approximate $p(\mathbf{z}|\mathbf{y}_{1:N})$

- Contributions: use **Stochastic Backpropagation** for 1) and 2); use **Amortization** for 3); use **Normalizing Flows** for 4)

Stochastic backpropagation

- Recall: in VI, we want to maximize the ELBO w.r.t. ϕ

$$\phi^* = \arg\max_{\phi} \left(\underbrace{\mathbb{E}_{q_\phi}[\log p(\mathbf{y}, \mathbf{z})] - \mathbb{E}_{q_\phi}[\log q_\phi(\mathbf{z}|\mathbf{y})]}_{\text{ELBO}} \right)$$

Stochastic backpropagation

- Recall: in VI, we want to maximize the ELBO w.r.t. ϕ

$$\phi^* = \arg\max_{\phi} \Big(\underbrace{\mathbb{E}_{q_\phi}[\log p(\mathbf{y}, \mathbf{z})] - \mathbb{E}_{q_\phi}[\log q_\phi(\mathbf{z}|\mathbf{y})]}_{\text{ELBO}} \Big)$$

- Reparameterize $\mathbf{z}$ in terms of a known base distribution and a differentiable transformation
  - Example: suppose $q_\phi(z) = \mathcal{N}(z|\mu, \sigma^2)$ with $\phi = \{\mu, \sigma\}$, then:

$$z \sim \mathcal{N}(z|\mu, \sigma^2) \Leftrightarrow z = \mu + \sigma\epsilon, \quad \epsilon \sim \mathcal{N}(0, 1)$$

Stochastic backpropagation

- Recall: in VI, we want to maximize the ELBO w.r.t. ϕ

$$\phi^* = \arg\max_{\phi} \left(\underbrace{\mathbb{E}_{q_\phi}[\log p(\mathbf{y}, \mathbf{z})] - \mathbb{E}_{q_\phi}[\log q_\phi(\mathbf{z}|\mathbf{y})]}_{\text{ELBO}} \right)$$

- Reparameterize **z** in terms of a known base distribution and a differentiable transformation
  - Example: suppose $q_\phi(z) = \mathcal{N}(z|\mu, \sigma^2)$ with $\phi = \{\mu, \sigma\}$, then:

$$z \sim \mathcal{N}(z|\mu, \sigma^2) \Leftrightarrow z = \mu + \sigma\epsilon, \quad \epsilon \sim \mathcal{N}(0, 1)$$

- Compute gradients of an arbitrary function $f(z)$ (e.g., ELBO) w.r.t. ϕ using a Monte Carlo approximation with draws from the base distribution, since

$$\nabla_\phi \mathbb{E}_{q_\phi(z)}[f(z)] \Leftrightarrow \mathbb{E}_{\mathcal{N}(\epsilon|0,1)}[\nabla_\phi f(\mu + \sigma\epsilon)]$$

- This allows us to construct flexible approximate distributions $q_\phi(\mathbf{z}|\mathbf{y})$ using Neural Networks and fit their parameters by backpropagating gradients

Amortized variational inference

- How to avoid that the number of variational parameters grows linearly with the number of respondents N?

$$q(\boldsymbol{\beta}_n) = \mathcal{N}(\boldsymbol{\beta}_n | \boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n)$$

Amortized variational inference

- How to avoid that the number of variational parameters grows linearly with the number of respondents N?

$$q(\boldsymbol{\beta}_n) = \mathcal{N}(\boldsymbol{\beta}_n | \boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n)$$

- Instead, consider a variational distribution of the form

$$q(\boldsymbol{\beta}_n | f_{\boldsymbol{\theta}}(\mathbf{y}_n, \mathbf{X}_n, \mathbf{a}_n))$$

where $f_{\boldsymbol{\theta}}(\mathbf{y}_n, \mathbf{X}_n, \mathbf{a}_n)$ maps the observed choice data of a decision-maker n to a set of variational parameters that represents his/her taste parameters

Amortized variational inference

- How to avoid that the number of variational parameters grows linearly with the number of respondents N?

$$q(\boldsymbol{\beta}_n) = \mathcal{N}(\boldsymbol{\beta}_n | \boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n)$$

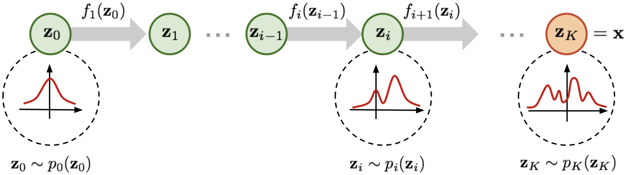
- Instead, consider a variational distribution of the form

$$q(\boldsymbol{\beta}_n | f_{\boldsymbol{\theta}}(\mathbf{y}_n, \mathbf{X}_n, \mathbf{a}_n))$$

  where $f_{\boldsymbol{\theta}}(\mathbf{y}_n, \mathbf{X}_n, \mathbf{a}_n)$ maps the observed choice data of a decision-maker n to a set of variational parameters that represents his/her taste parameters
- $f_{\boldsymbol{\theta}}$ must be flexible and differentiable - use a deep neural network!
- Estimate neural network parameters $\boldsymbol{\theta}$ through stochastic backpropagation
- Number of variational parameters no longer grows with N (it is fixed)
- Neural network $f_{\boldsymbol{\theta}}$ can extrapolate across respondents

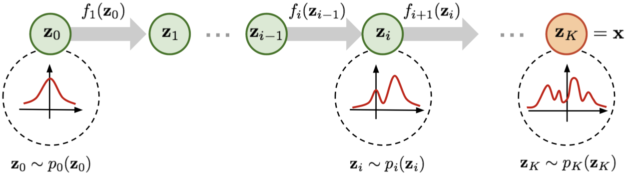
Normalizing flows

- Obtain complex density approximations $q_\phi(\mathbf{z}|\mathbf{y})$ to the true posterior
- Take a simple base distribution $p(\mathbf{u})$ (e.g. Gaussian) and apply a series of bijective differentiable transformations $T = T_K \circ \cdots \circ T_1$



Normalizing flows

- Obtain complex density approximations $q_\phi(\mathbf{z}|\mathbf{y})$ to the true posterior
- Take a simple base distribution $p(\mathbf{u})$ (e.g. Gaussian) and apply a series of bijective differentiable transformations $T = T_K \circ \cdots \circ T_1$



- Let $\mathbf{z}_k$ be the value of a sample $\mathbf{z}_0 \sim p(\mathbf{u})$ takes after the k-th transformation
- Log probability of target distribution is

$$\log p(\mathbf{z}) = \log p(\mathbf{u}) - \sum_{k=1}^{K} \log |\det \mathbf{J}_{T_k}(\mathbf{z}_{k-1})|$$

where $\mathbf{J}_T(\mathbf{u}) = \frac{\partial T}{\partial \mathbf{u}}$ is the Jacobian of the transformation

Experiments: parameter recovery

- Simulated panel data: $N = 500$ respondents, $T = 5$ menus, $J = 5$ alternatives, $L = 3$ fixed effects and $K = 5$ random effects

Experiments: parameter recovery

- Simulated panel data: $N = 500$ respondents, $T = 5$ menus, $J = 5$ alternatives, $L = 3$ fixed effects and $K = 5$ random effects
- Baselines:
  - Maximum simulated likelihood estimation (MSLE)
  - Gibbs sampling
  - Variational inference with delta-method-based approximation (NCVMP-Δ from Bansal et al., 2020)

Experiments: parameter recovery

- Simulated panel data: $N = 500$ respondents, $T = 5$ menus, $J = 5$ alternatives, $L = 3$ fixed effects and $K = 5$ random effects
- Baselines:
  - Maximum simulated likelihood estimation (MSLE)
  - Gibbs sampling
  - Variational inference with delta-method-based approximation (NCVMP-Δ from Bansal et al., 2020)
- 2 variants of the proposed approach:
  - SVI-LKJ: Only stochastic variational inference (SVI) approach (no amortization)
  - AVI-LKJ: Proposed amortized variational inference (AVI) approach

Experiments: parameter recovery

- Simulated panel data: $N = 500$ respondents, $T = 5$ menus, $J = 5$ alternatives, $L = 3$ fixed effects and $K = 5$ random effects
- Baselines:
  - Maximum simulated likelihood estimation (MSLE)
  - Gibbs sampling
  - Variational inference with delta-method-based approximation (NCVMP-Δ from Bansal et al., 2020)
- 2 variants of the proposed approach:
  - SVI-LKJ: Only stochastic variational inference (SVI) approach (no amortization)
  - AVI-LKJ: Proposed amortized variational inference (AVI) approach

$N = 500$; $T = 5$; $J = 5$; $L = 3$; $K = 5$; Batch Size = 500

Method	Runtime (s)	Sim. Loglik.	RMSE $\boldsymbol{\alpha}$	RMSE $\boldsymbol{\zeta}$	RMSE $\boldsymbol{\beta}_n$
MSLE	176 ($\pm$24)	-3475 ($\pm$34)	0.081 ($\pm$0.034)	0.094 ($\pm$0.033)	0.785 ($\pm$0.025)
Gibbs	227 ($\pm$6)	-3477 ($\pm$34)	0.080 ($\pm$0.035)	0.095 ($\pm$0.033)	0.777 ($\pm$0.024)
NCVMP-Δ	62 ($\pm$9)	-3490 ($\pm$30)	0.087 ($\pm$0.035)	0.098 ($\pm$0.034)	0.782 ($\pm$0.024)
SVI-LKJ	125 ($\pm$17)	-3483 ($\pm$34)	0.078 ($\pm$0.032)	0.093 ($\pm$0.034)	0.790 ($\pm$0.025)
AVI-LKJ	127 ($\pm$21)	-3482 ($\pm$34)	0.078 ($\pm$0.034)	0.093 ($\pm$0.033)	0.792 ($\pm$0.024)

Experiments: scalability

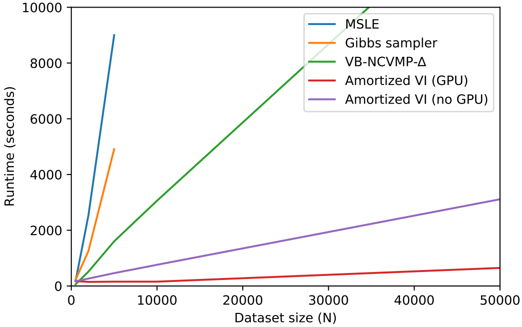
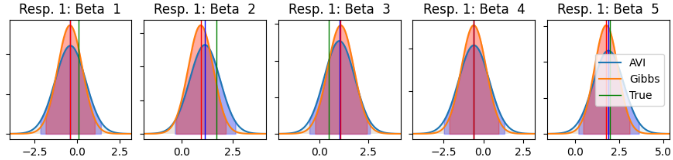
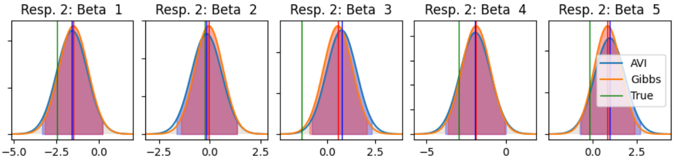


Figure: Scalability plot comparing the proposed Amortized VI approach with MSLE, Gibbs sampling and the VB-NCVMP-Δ approach from Bansal et al. (2020)

Experiments: credible intervals



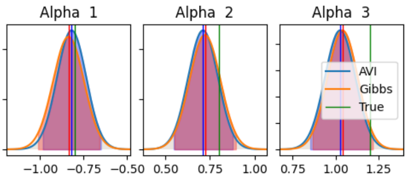
(a) Parameters of respondent 1 ($\boldsymbol{\beta}_1$)



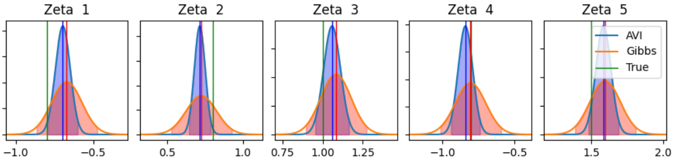
(b) Parameters of respondent 2 ($\boldsymbol{\beta}_2$)

DTU MLSM Machine Learning for Smart Mobility

Experiments: credible intervals



(c) Fixed parameters (α)



(d) Random parameters (ζ)

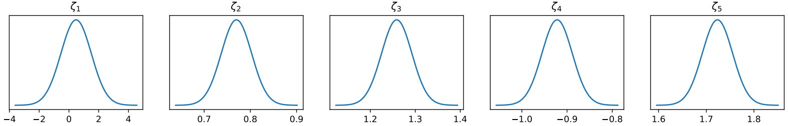
Experiments: normalizing flows

- Modified the model such that ζ_1 enters the utility function as $e^{\zeta_1} x_1$ instead of $\zeta_1 x_1$
- "Forces" a non-Gaussian posterior distribution on ζ_1 which we can try to capture with normalizing flows (NFs)

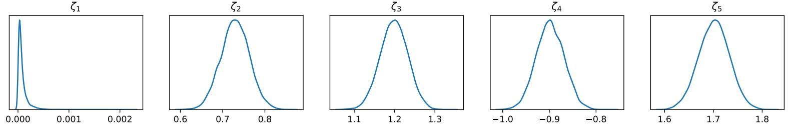
Table: Results obtained by Sylvester normalizing flows in comparison with two baseline parametric approximations

$N = 500; T = 5; J = 5; L = 3; K = 5$; Batch Size = 500		
Method	Runtime (s)	Sim. Loglik.
SVI-LKJ (Normal)	227 ($\pm$2)	-3653 ($\pm$53)
SVI-LKJ (LogNormal)	225 ($\pm$3)	-3564 ($\pm$55)
SVI-LKJ (NFs)	284 ($\pm$4)	-3492 ($\pm$61)

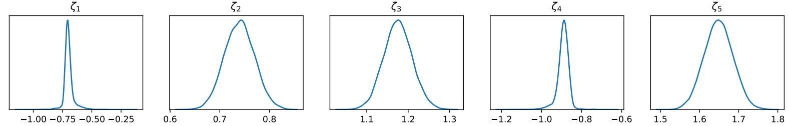
Experiments: normalizing flows



(e) Normal approximate distribution



(f) Log-Normal approximate distribution



(g) Sylvester normalizing flows (NFs)

DTU MLSM Machine Learning for Smart Mobility

Experiments: London data

- London Passenger Mode Choice dataset provided by Hillel et al. (2018)
- Revealed preferences (RP)
- 3 alternatives: walking, public transport (PT) and car
- Focus only on individuals with age between 18 and 75 years old and trips with a walking time of less than 2 hours
- Total of 43778 trips
- Alternative attributes: travel cost, travel time, trip purpose (interacted with travel time) and discount fare

Method	Estimation time (s)	Sim. Loglik.
MSLE	10588 ($\pm$36)	-23216 ($\pm$168)
Gibbs	4856 ($\pm$89)	-23337 ($\pm$195)
SVI-LKJ	652 ($\pm$58)	-23697 ($\pm$176)
AVI-LKJ	496 ($\pm$51)	-23683 ($\pm$157)

Experiments: London data

Table: Comparison between the estimated means and credible intervals of AVI-LKJ with Gibbs sampling

Parameter	Mean			Std. Dev.	
	MSLE	Gibbs	AVI	Gibbs	AVI
ASC Walk (α_1)	2.333	2.242	2.328	0.080	0.034
ASC PT Full Ticket (α_2)	0.424	0.449	0.315	0.052	0.038
ASC PT Disabled (α_3)	0.862	0.800	0.845	0.058	0.043
Travel Time x Purpose B (ζ_1)	-1.815	-1.718	-1.929	0.051	0.016
Travel Time x Purpose HBE (ζ_2)	-2.034	-1.943	-2.043	0.070	0.017
Travel Time x Purpose HBO (ζ_3)	-1.740	-1.709	-1.651	0.039	0.011
Travel Time x Purpose HBW (ζ_4)	-1.398	-1.350	-1.491	0.061	0.022
Travel Time x Purpose NHBO (ζ_5)	-1.835	-1.787	-1.956	0.059	0.048
Travel Cost (ζ_6)	-0.611	-0.683	-0.412	0.059	0.058

Conclusions

- Scaled VI in Bayesian Mixed Logit models to large datasets
- Relaxed constraints on the choice of priors (e.g., conjugacy)
- Allowed for flexible posterior approximations (Normalizing Flows)
- Increased the support for the interaction between choice models and advanced ML methods
  - Created a new Python library - PyDCML (in collaboration with Rico Krueger)
  - Easy-to-use formula interface:

    ```
    V1 = BETA_COST*ALT1_COST + BETA_DUR*ALT1_DURATION + ...
    ```
  - Fast implementation and scalable inference of Bayesian Discrete Choice Models
  - GPU support (PyTorch backend)

PyDCML

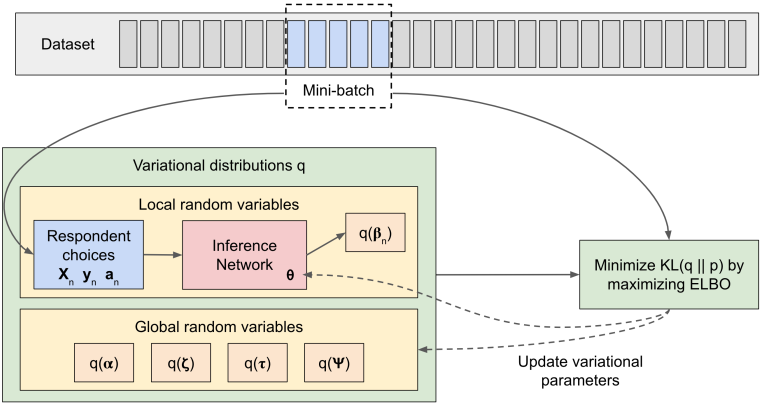
- GitHub: `https://github.com/fmpr/pyDCML`
- Documentation: `https://mlsm.man.dtu.dk/pydcml/intro.html`
- PyDCML aims at enabling flexible and expressive Choice Modeling, unifying the best of modern Machine Learning and Bayesian modeling with Discrete Choice Theory
- Easy to extend to new models/ideas
- Currently supports:
  - Mixed Logit models with neural networks in the utilities[2]

    ```
    V1 = BETA_COST*ALT1_COST + BETA_DUR*ALT1_DURATION
              + NNET(INCOME, AGE, GENDER) + ...
    ```
  - Mixed Logit models with Automatic Relevance Determination[3]
  - Mixed Ordered Logit models
- Upcoming: Context-aware Bayesian choice models (presented at ICMC 2022)
- All with fast and scalable Bayesian inference!

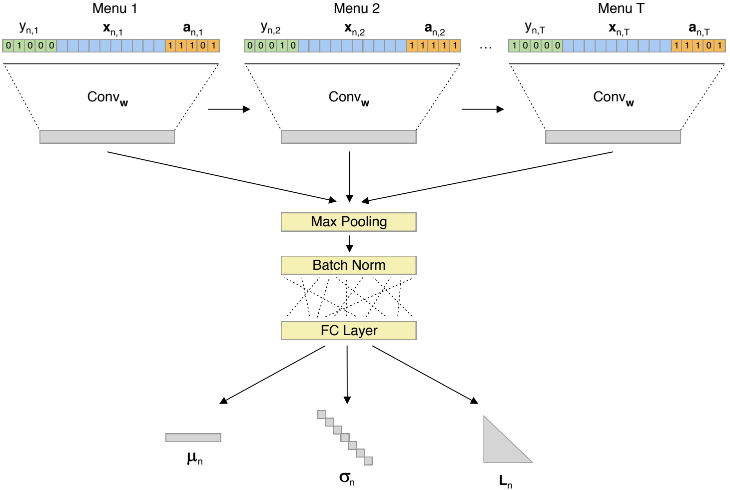
[2] Extends: *Sifringer, B., Lurkin, V. and Alahi, A., 2020. Enhancing discrete choice models with representation learning. Transp. Res. Part B*

[3] Extends: *Rodrigues, F., Ortelli, N., Bierlaire, M. and Pereira, F.C., 2020. Bayesian automatic relevance determination for utility function specification in discrete choice models. IEEE Transactions on Intelligent Transportation Systems.*

Amortized variational inference



Inference network architecture



Sylvester normalizing flows

- Careful: T must be chosen such that $\mathbf{J}_T$ is efficient to compute
- Sylvester normalizing flows (Berg et al., 2018) assume:

$$\mathbf{z}_k = \mathbf{z}_{k-1} \mathbf{Q}\mathbf{R}\, h(\tilde{\mathbf{R}}\mathbf{Q}^T \mathbf{z}_{k-1} + \mathbf{b})$$

  where h is a smooth activation function and $\{\mathbf{R}, \tilde{\mathbf{R}}, \mathbf{Q}, \mathbf{b}\}$ are (constrained) learnable parameters
- This transformation is invertible and $\mathbf{J}_T$ can be computed in linear time
- Resembles a multi-layer fully-connected neural network
- Expressive building block for constructing arbitrarily complex distributions!

Experiments: parameter recovery

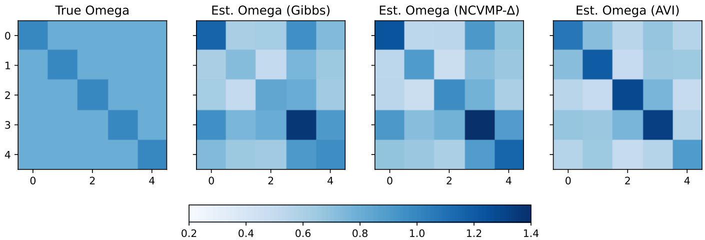


Figure: Visualization of the posterior mean of covariance matrix Ω obtained by different approaches

Experiments: out-of-sample generalization

- Once estimated, the inference network can be applied to infer the preference parameters $\boldsymbol{\beta}_n$ of unseen respondents "on-the-fly"

N	Loglikelihood (normalized by N) Train	Test	RMSE $\boldsymbol{\beta}_n$ Train	Test
500	12.278 ($\pm$0.158)	13.604 ($\pm$0.098)	0.670 ($\pm$0.014)	0.883 ($\pm$0.018)
2000	12.392 ($\pm$0.076)	13.022 ($\pm$0.053)	0.661 ($\pm$0.007)	0.767 ($\pm$0.011)
10000	12.576 ($\pm$0.028)	12.732 ($\pm$0.028)	0.659 ($\pm$0.003)	0.694 ($\pm$0.004)

Table: Results for the generalization ability of the inference network ($T = 10$; $J = 5$; $L = 3$; $K = 5$)