# Solving Inverse Problems by Amortized Variational Inference

Jie Wang, CUHK(SZ)
Fourth Year Undergraduate Student
Pure Mathematics Major
Network Coding Lab

香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

网络编码实验室
Network Coding Lab

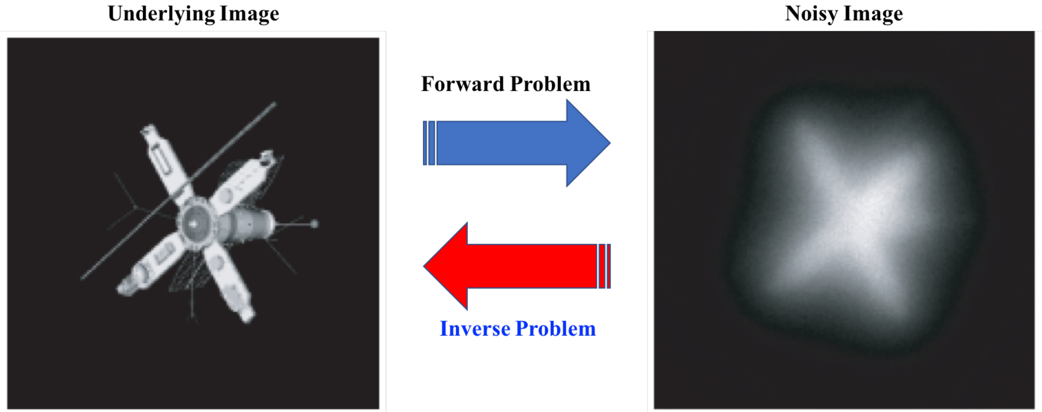
December 18, 2019

# Table of Content

# Table of Contents

# Motivation

- Inverse problems refer to the *reverse process* of a forward problem.
- Example:



# Statistical Framework for Inverse Problems

- $\boldsymbol{z} \sim g(\boldsymbol{z}; \boldsymbol{\Lambda})$ with the unknown parameter $\boldsymbol{\Lambda}$
- $\boldsymbol{x}$ is generated through a known likelihood model $p(\boldsymbol{x} \mid \boldsymbol{z})$:
- Given $n$ observed data points $\{\boldsymbol{x}_i\}_{i=1}^n$, recover $\{\boldsymbol{z}_i\}_{i=1}^n$.

$$\boxed{\{\boldsymbol{z}_i\}_{i=1}^n} \xrightarrow{p(\boldsymbol{x}_i \mid \boldsymbol{z}_i)} \boxed{\{\boldsymbol{x}_i\}_{i=1}^n} \xrightarrow{\text{Estimator}} \boxed{\{\hat{\boldsymbol{z}}_i\}_{i=1}^n}$$

- Example: $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, A\boldsymbol{I})$, and $\boldsymbol{x} \mid \boldsymbol{z} \sim \mathcal{N}(\boldsymbol{z}, \sigma^2 \boldsymbol{I})$.

  If given observations $\{\boldsymbol{x}_i\}_{i=1}^n$, then what is $\{\boldsymbol{z}_i\}_{i=1}^n$?

  1. Naive idea: $\boldsymbol{z}_i^{(\text{MLE})} = \boldsymbol{x}_i$
  2. Non-trivial idea:

$$\boldsymbol{z}_i^{(\text{JS})} = \left(1 - \frac{(n-2)\sigma^2}{\|\boldsymbol{x}_i\|^2}\right)\boldsymbol{x}_i.$$

# Statistical Framework for Inverse Problems

- $\boldsymbol{z} \sim g(\boldsymbol{z}; \boldsymbol{\Lambda})$ with the unknown parameter $\boldsymbol{\Lambda}$
- $\boldsymbol{x}$ is generated through a known likelihood model $p(\boldsymbol{x} \mid \boldsymbol{z})$:
- Given $n$ observed data points $\{\boldsymbol{x}_i\}_{i=1}^n$, recover $\{\boldsymbol{z}_i\}_{i=1}^n$.

$$\boxed{\{\boldsymbol{z}_i\}_{i=1}^n} \xrightarrow{p(\boldsymbol{x}_i \mid \boldsymbol{z}_i)} \boxed{\{\boldsymbol{x}_i\}_{i=1}^n} \xrightarrow{\text{Estimator}} \boxed{\{\hat{\boldsymbol{z}}_i\}_{i=1}^n}$$

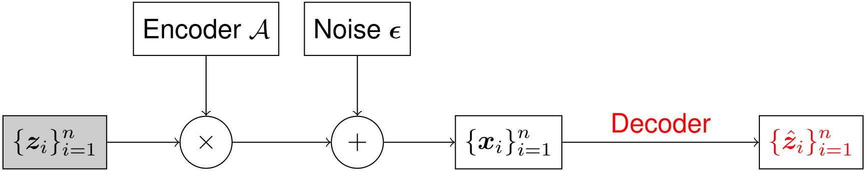
- Example: $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, A\boldsymbol{I})$, and $\boldsymbol{x} \mid \boldsymbol{z} \sim \mathcal{N}(\boldsymbol{z}, \sigma^2 \boldsymbol{I})$.

  If given observations $\{\boldsymbol{x}_i\}_{i=1}^n$, then what is $\{\boldsymbol{z}_i\}_{i=1}^n$?

  1. Naive idea: $\boldsymbol{z}_i^{\text{(MLE)}} = \boldsymbol{x}_i$
  2. Non-trivial idea:

$$\boldsymbol{z}_i^{\text{(JS)}} = \left(1 - \frac{(n-2)\sigma^2}{\|\boldsymbol{x}_i\|^2}\right)\boldsymbol{x}_i.$$

# Typical Inverse Problems




- The observation model can be represented as:

$$x_i = \mathcal{A}(z_i) + \epsilon, \quad i = 1, \ldots, n$$

- For known encoder $\mathcal{A}$ and observations $\{x_i\}_{i=1}^n$, wish to train a decoder.

# Table of Contents

# Empirical Bayes approach

- Given $n$ observed data points $\{\boldsymbol{x}_i\}_{i=1}^n$, aim to recover $\{\boldsymbol{z}_i\}_{i=1}^n$:

$$\boxed{\{\boldsymbol{z}_i\}_{i=1}^n \sim g(\boldsymbol{z};\boldsymbol{\Lambda})} \xrightarrow{p(\boldsymbol{x}_i \mid \boldsymbol{z}_i)} \boxed{\{\boldsymbol{x}_i\}_{i=1}^n} \xrightarrow{\text{Inferrer}} \boxed{\{\hat{\boldsymbol{z}}_i\}_{i=1}^n}$$

- **Empirical Bayes**:

Estimation:

$$\boxed{\{\boldsymbol{z}_i\}_{i=1}^n} \xrightarrow{p(\boldsymbol{x}_i \mid \boldsymbol{z}_i)} \boxed{\{\boldsymbol{x}_i\}_{i=1}^n} \longrightarrow \boxed{\text{Estimator}} \xrightarrow{\hat{g}(\boldsymbol{z}) := g(\boldsymbol{z};\hat{\boldsymbol{\Lambda}})}$$

Inference:

$$\xrightarrow{\boldsymbol{x}} \boxed{\text{Inferrer } \hat{g}(\boldsymbol{z} \mid \boldsymbol{x})} \xrightarrow{\hat{\boldsymbol{z}}(\boldsymbol{x})}$$

# $G$-modelling for Empirical Bayes

Estimation:

$$\boxed{\{\boldsymbol{z}_i\}_{i=1}^n \sim g(\boldsymbol{z}; \boldsymbol{\Lambda})} \xrightarrow{p(\boldsymbol{x}_i \mid \boldsymbol{z}_i)} \boxed{\{\boldsymbol{x}_i\}_{i=1}^n} \longrightarrow \boxed{\text{Estimator}} \xrightarrow{\hat{g}(\boldsymbol{z}) := g(\boldsymbol{z}; \hat{\boldsymbol{\Lambda}})}$$

The estimation problem relies on maximizing the marginal likelihood:

$$\hat{\boldsymbol{\Lambda}} = \arg\max_{\boldsymbol{\Lambda}} \sum_{i=1}^{n} \log p(\boldsymbol{x}_i) \triangleq \max \sum_{i=1}^{n} \log \int g(\boldsymbol{z}_i; \boldsymbol{\Lambda}) p(\boldsymbol{x}_i \mid \boldsymbol{z}_i) \, \mathrm{d}\boldsymbol{z}_i \tag{1}$$

Intractable for complicated prior distribution or high dimension latent space!

# $G$-modelling for Empirical Bayes

Estimation:

$$\boxed{\{\boldsymbol{z}_i\}_{i=1}^n \sim g(\boldsymbol{z};\boldsymbol{\Lambda})} \xrightarrow{p(\boldsymbol{x}_i \mid \boldsymbol{z}_i)} \boxed{\{\boldsymbol{x}_i\}_{i=1}^n} \longrightarrow \boxed{\text{Estimator}} \xrightarrow{\hat{g}(\boldsymbol{z}) := g(\boldsymbol{z};\hat{\boldsymbol{\Lambda}})}$$

The estimation problem relies on maximizing the marginal likelihood:

$$\hat{\boldsymbol{\Lambda}} = \arg\max_{\boldsymbol{\Lambda}} \sum_{i=1}^n \log p(\boldsymbol{x}_i) \triangleq \max \sum_{i=1}^n \log \int g(\boldsymbol{z}_i;\boldsymbol{\Lambda}) p(\boldsymbol{x}_i \mid \boldsymbol{z}_i)\, \mathrm{d}\boldsymbol{z}_i \quad (1)$$

Intractable for complicated prior distribution or high dimension latent space!

# Evidence Lower Bound

## Lemma: Variational Lower Bound

$$\begin{aligned}\log p(\boldsymbol{x}) &\geq \mathbb{E}_{\boldsymbol{z}\sim q}[\log p(\boldsymbol{x}, \boldsymbol{z}) - \log q(\boldsymbol{z})] \\ &\triangleq \mathbb{E}_{\boldsymbol{z}\sim q}[\log p(\boldsymbol{z}) + \log p(\boldsymbol{x} \mid \boldsymbol{z}) - \log q(\boldsymbol{z})]\end{aligned}$$

Proof.

$$\begin{aligned}\log p(\boldsymbol{x}) &= \log \int_{\boldsymbol{z}} p(\boldsymbol{x}, \boldsymbol{z})\, \mathrm{d}\boldsymbol{z} \\ &= \log \int_{\boldsymbol{z}} p(\boldsymbol{x}, \boldsymbol{z}) \frac{q(\boldsymbol{z})}{q(\boldsymbol{z})}\, \mathrm{d}\boldsymbol{z} = \log \left( \mathbb{E}_{\boldsymbol{z}\sim q}\left[ \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})} \right] \right) \\ &\geq \mathbb{E}_{\boldsymbol{z}\sim q}\left[ \log \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})} \right] = \mathbb{E}_{\boldsymbol{z}\sim q}[\log p(\boldsymbol{x}, \boldsymbol{z}) - \log q(\boldsymbol{z})]\end{aligned}$$

□

# Variational Inference Approach

$$\boxed{\{\boldsymbol{z}_i\}_{i=1}^n} \xrightarrow{p(\boldsymbol{x}_i \mid \boldsymbol{z}_i)} \boxed{\{\boldsymbol{x}_i\}_{i=1}^n} \xrightarrow{\text{Inferrer}} \boxed{\{\hat{\boldsymbol{z}}_i\}_{i=1}^n}$$

- Jointly perform the estimation and inference task:

$$\log p(\boldsymbol{x}; \boldsymbol{\Lambda}) \geq \mathbb{E}_{q_\phi(\boldsymbol{z})}[\log p(\boldsymbol{z}; \boldsymbol{\Lambda}) + \log p(\boldsymbol{x} \mid \boldsymbol{z}) - \log q_\phi(\boldsymbol{z})] \triangleq \text{ELBO}(\boldsymbol{x}; \boldsymbol{\Lambda}, \phi)$$

  where $q_\phi(\boldsymbol{z})$ is the approximation of the true posterior $p(\boldsymbol{z} \mid \boldsymbol{x}; \boldsymbol{\Lambda})$.
- The optimization for ELBO relies on *mean-field approximation* technique:
  1. Large suboptimality Gap, and therefore unreliable estimator and inferrer
  2. Non-convexity Landscape with local optimal points
  3. Limited Choice of posterior approximation.

# Variational Inference Approach

$$\{\boldsymbol{z}_i\}_{i=1}^n \xrightarrow{p(\boldsymbol{x}_i \mid \boldsymbol{z}_i)} \{\boldsymbol{x}_i\}_{i=1}^n \xrightarrow{\text{Inferrer}} \{\hat{\boldsymbol{z}}_i\}_{i=1}^n$$

- Jointly perform the estimation and inference task:

$$\log p(\boldsymbol{x}; \boldsymbol{\Lambda}) \geq \mathbb{E}_{q_\phi(\boldsymbol{z})}[\log p(\boldsymbol{z}; \boldsymbol{\Lambda}) + \log p(\boldsymbol{x} \mid \boldsymbol{z}) - \log q_\phi(\boldsymbol{z})] \triangleq \text{ELBO}(\boldsymbol{x}; \boldsymbol{\Lambda}, \phi)$$

  where $q_\phi(\boldsymbol{z})$ is the approximation of the true posterior $p(\boldsymbol{z} \mid \boldsymbol{x}; \boldsymbol{\Lambda})$.
- The optimization for ELBO relies on *mean-field approximation* technique:
  1. Large suboptimality Gap, and therefore unreliable estimator and inferrer
  2. Non-convexity Landscape with local optimal points
  3. Limited Choice of posterior approximation.

# Table of Contents

# Vanilla Amortized Variational Inference

- Maximize the ELBO function via stochastic optimization techniques:

$$(\hat{\boldsymbol{\Lambda}}, \hat{\boldsymbol{\phi}}) = \arg\max_{\boldsymbol{\Lambda}, \boldsymbol{\phi}} \sum_{i=1}^{n} \mathbb{E}_{q_\phi(\boldsymbol{z}_i|\boldsymbol{x}_i)} \Big[ \log g(\boldsymbol{z}_i; \boldsymbol{\Lambda}) + \log p(\boldsymbol{x}_i \mid \boldsymbol{z}_i) - \log q_\phi(\boldsymbol{z}_i \mid \boldsymbol{x}_i) \Big]$$

  where:
  - $q_\phi(\boldsymbol{z}_i \mid \boldsymbol{x}_i)$ is the approximation of the true posterior $p(\boldsymbol{z} \mid \boldsymbol{x}; \boldsymbol{\Lambda})$:

$$(\boldsymbol{\mu}_{1:n}, \log \boldsymbol{\sigma}_{1:n}) = \text{Encoder-Neural-Net}_\phi(\boldsymbol{x}_{1:n});$$

$$q_\phi(\boldsymbol{z} \mid \boldsymbol{x}) = \prod_{i=1}^{n} q_\phi(\boldsymbol{z}_i \mid \boldsymbol{x}_i) = \prod_{i=1}^{n} \mathcal{N}(\boldsymbol{\mu}_i, \text{diag}(\boldsymbol{\sigma}_i^2));$$

$$q_\phi(\boldsymbol{z} \mid \boldsymbol{x}) \approx p(\boldsymbol{z} \mid \boldsymbol{x}; \Lambda).$$

- Optimize for $\phi$: reparametrization trick $z = \mu + \boldsymbol{\sigma} \circ \epsilon$ with $\epsilon \sim \mathcal{N}(0, 1)$.
  Optimize for $\Lambda$: parametric optimization techniques.

# Vanilla Amortized Variational Inference

- Maximize the ELBO function via stochastic optimization techniques:

$$(\hat{\boldsymbol{\Lambda}}, \hat{\boldsymbol{\phi}}) = \arg\max_{\boldsymbol{\Lambda}, \boldsymbol{\phi}} \sum_{i=1}^{n} \mathbb{E}_{q_\phi(\boldsymbol{z}_i|\boldsymbol{x}_i)} \left[ \log g(\boldsymbol{z}_i; \boldsymbol{\Lambda}) + \log p(\boldsymbol{x}_i \mid \boldsymbol{z}_i) - \log q_\phi(\boldsymbol{z}_i \mid \boldsymbol{x}_i) \right]$$

  where:

  - $q_\phi(\boldsymbol{z}_i \mid \boldsymbol{x}_i)$ is the approximation of the true posterior $p(\boldsymbol{z} \mid \boldsymbol{x}; \boldsymbol{\Lambda})$:

$$(\boldsymbol{\mu}_{1:n}, \log \boldsymbol{\sigma}_{1:n}) = \text{Encoder-Neural-Net}_\phi(\boldsymbol{x}_{1:n});$$

$$q_\phi(\boldsymbol{z} \mid \boldsymbol{x}) = \prod_{i=1}^{n} q_\phi(\boldsymbol{z}_i \mid \boldsymbol{x}_i) = \prod_{i=1}^{n} \mathcal{N}(\boldsymbol{\mu}_i, \text{diag}(\boldsymbol{\sigma}_i^2));$$

$$q_\phi(\boldsymbol{z} \mid \boldsymbol{x}) \approx p(\boldsymbol{z} \mid \boldsymbol{x}; \Lambda).$$

- Optimize for $\phi$: reparametrization trick $\boldsymbol{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \circ \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(0, 1)$.
  Optimize for $\boldsymbol{\Lambda}$: parametric optimization techniques.

# Vanilla Amortized Variational Inference (AVI)

**Algorithm 1** Algorithm for Vanilla AVI approach. All experiments in the paper used the default values $\alpha = 0.000001$, $B = 128$, $n_{\text{critic}} = 30$

**Input:** $\{\boldsymbol{x}_i\}_{i=1}^{n}$, $p(\boldsymbol{x} \mid \boldsymbol{z})$, $\eta$, and $n_{\text{critic}}$, the number of iterations of the $\phi$ update per $\boldsymbol{\Lambda}$ estimation.
**Output:** $\hat{\boldsymbol{\Lambda}}, \hat{\phi}$: learnt parameters.

1: $\hat{\boldsymbol{\Lambda}}, \hat{\phi} \leftarrow$ initialize parameters
2: **for** $t = 0, \ldots, n_{\text{critic}}$ **do**
3: Generate $\{z_{(i)}\}_{i=1}^{B}$ by the reparameriation trick.
4: Compute the objective function $\tilde{L}_{\boldsymbol{\Lambda},\boldsymbol{\phi}}$ and its gradients:

$$\tilde{L}_{\boldsymbol{\Lambda},\boldsymbol{\phi}} \triangleq \sum_{i=1}^{B} -\log(\boldsymbol{z}_{(i)}; \boldsymbol{\Lambda}) - \log p(\boldsymbol{x}_{(i)} \mid \boldsymbol{z}_{(i)}) - \sum_{j} \log |\boldsymbol{\sigma}_{(i),j}|$$

5: Update $\hat{\phi}$ using Adam optimizer
6: **end for**
7: Generate $\{z_{(i)}\}_{i=1}^{B}$, and update $\boldsymbol{\Lambda}$ by optimizing $\sum_{i=1}^{B} \log(\boldsymbol{z}_{(i)}; \boldsymbol{\Lambda})$.

# AVI with Inverse Autoregressive Flow

- Vanilla AVI suffices from the inexact approximation posterior.

## Inverse Autoregressive Flow Trick

Initialize with

$$\begin{aligned}\boldsymbol{\epsilon}_0 &\sim \mathcal{N}(0, \boldsymbol{I}),\\ (\boldsymbol{\mu}_0, \log(\boldsymbol{\sigma}_0), \boldsymbol{h}) &= \text{Encoder-Neural-Net}(\boldsymbol{x}; \psi)\\ \boldsymbol{z}_0 &= \boldsymbol{\mu}_0 + \boldsymbol{\sigma}_0 \circ \boldsymbol{\epsilon}_0\end{aligned}$$

Then apply the following transformations for $t = 1, \ldots, T$:

$$\begin{aligned}(\boldsymbol{m}_t, \boldsymbol{s}_t) &= \text{Auto-regressive-Neural-Net}_t(\boldsymbol{\epsilon}_{t-1}, \boldsymbol{h}; \psi)\\ \boldsymbol{\sigma}_t &= \text{sigmoid}(\boldsymbol{s}_t)\\ \boldsymbol{\epsilon}_t &= \boldsymbol{\sigma}_t \circ \boldsymbol{\epsilon}_{t-1} + (1 - \boldsymbol{\sigma}_t) \circ \boldsymbol{m}_t\end{aligned}$$

and finally $\boldsymbol{z} \triangleq \boldsymbol{\epsilon}_t$.

# AVI with Inverse Autoregressive Flow

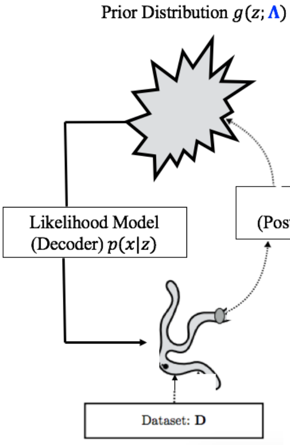
Two things to be modified based on Vanilla AVI:

$$(\hat{\boldsymbol{\Lambda}}, \hat{\boldsymbol{\phi}}) = \arg\max_{\boldsymbol{\Lambda}, \boldsymbol{\phi}} \sum_{i=1}^{n} \mathbb{E}_{q_\phi(\boldsymbol{z}_i | \boldsymbol{x}_i)} \left[ \log g(\boldsymbol{z}_i; \boldsymbol{\Lambda}) + \log p(\boldsymbol{x}_i \mid \boldsymbol{z}_i) - \log q_\phi(\boldsymbol{z}_i \mid \boldsymbol{x}_i) \right]$$

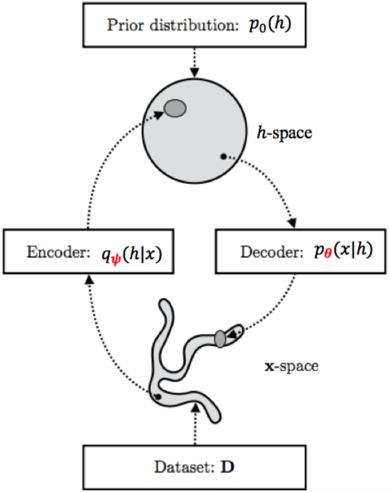
1. The sampling for $\boldsymbol{z}$ via $q_\phi(\boldsymbol{z}_i \mid \boldsymbol{x}_i)$ follows Inverse Autoregressive Flow process;
2. Substitute the evaluation for the term $\log q_\phi(\boldsymbol{z}_i \mid \boldsymbol{x}_i)$:

$$\log q_\phi(\boldsymbol{z} \triangleq \boldsymbol{\epsilon}_T \mid \boldsymbol{x}) = -\sum_{i=1}^{n} \left( \frac{1}{2} \epsilon_i^2 + \frac{1}{2} \log(2\pi) + \sum_{t=0}^{T} \sigma_{t,i} \right)$$

# Connection to VAE

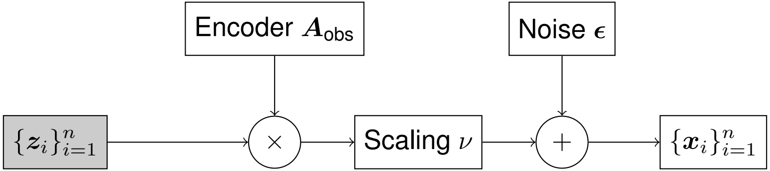



$g(z;\Lambda) = G_\theta \# p_0(h)$



# Table of Contents

# Problem Setting




$$\begin{aligned} &\text{(Latent Space)} && \boldsymbol{z}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Lambda}^{-1}), \quad \boldsymbol{\Lambda} \text{ sparse}, \\ &\text{(Observation Space)} && \boldsymbol{x}_i \mid \boldsymbol{z}_i \sim \mathcal{N}(\nu(\boldsymbol{A}_{\text{obs}}\boldsymbol{z}_i), \sigma^2_{\text{obs}}\boldsymbol{I}) \end{aligned}$$

where $\nu$ is the sigmoid function.

# Vanilla AVI Approach

1. Estimation for $\phi := (\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$: generate $\boldsymbol{z}_{(i)}$'s and then minimize

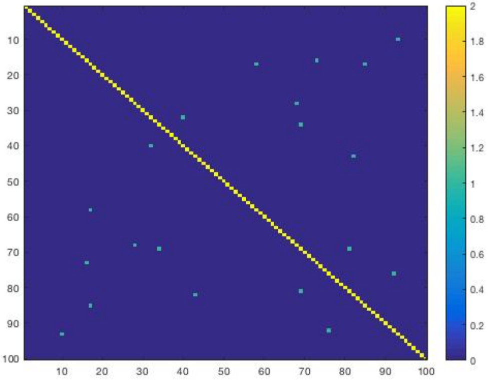
$$\frac{1}{2}\,\text{Trace}\left(\boldsymbol{\Lambda}\cdot\sum_{i=1}^{B}(\boldsymbol{z}_{(i)})(\boldsymbol{z}_{(i)})^{\mathrm{T}}\right)+\frac{1}{2\sigma_{\text{obs}}^2}\sum_{i=1}^{B}\|\boldsymbol{x}_{(i)}-\nu(\boldsymbol{A}_{\text{obs}}\boldsymbol{z}_{(i)})\|^2-\sum_{i=1,j}^{B}\log|\sigma_{(i),j}|$$

2. Estimation for $\boldsymbol{\Lambda}$: generate $\boldsymbol{z}_{(i)}$'s and then solve the graphical lasso subproblem:

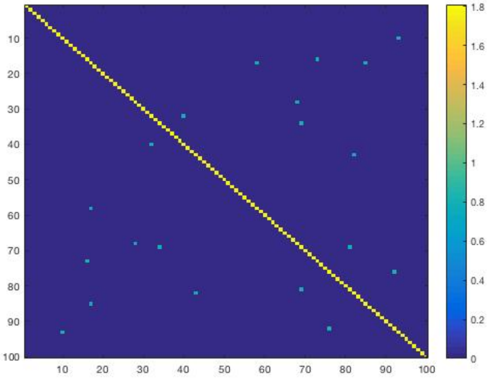
$$\arg\max_{\boldsymbol{\Lambda}}\quad -\lambda\cdot\|\boldsymbol{\Lambda}\|_{\ell_1,\text{off}}+\frac{B}{2}\log|\boldsymbol{\Lambda}|-\frac{B}{2}\,\text{Trace}\left[\boldsymbol{\Lambda}\cdot\frac{1}{B}\sum_{i=1}^{B}(\boldsymbol{z}_{(i)})(\boldsymbol{z}_{(i)})^{\mathrm{T}}\right]$$

Disadvantage: graphical lasso problem is computationally expansive!

# Simulation Results

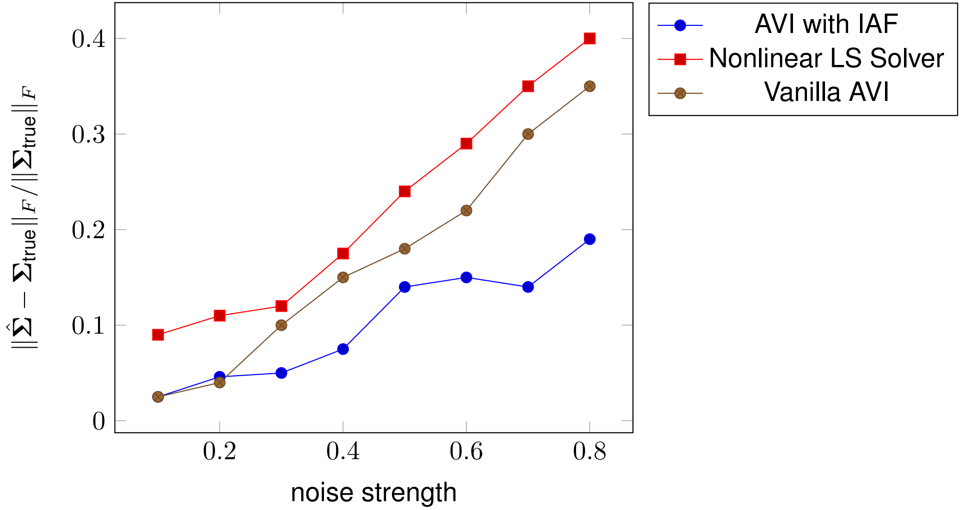



(a) Underlying Precision matrix $\mathbf{\Lambda}_{\text{true}}$




(b) Estimated Precision matrix $\hat{\mathbf{\Lambda}}$

# Simulation Results



# Table of Contents

# Conclusion

- The inverse problem is solved by amoritzed variational inference with IAF trick:

$$\boxed{\{\boldsymbol{z}_i\}_{i=1}^n} \xrightarrow{p(\boldsymbol{x}_i \mid \boldsymbol{z}_i)} \boxed{\{\boldsymbol{x}_i\}_{i=1}^n} \xrightarrow{\text{Estimator}} \boxed{\{\hat{\boldsymbol{z}}_i\}_{i=1}^n}$$

  This approach is to jointly estimating $g(\boldsymbol{z})$ and $p(\boldsymbol{z} \mid \boldsymbol{x})$.
- Applicable to general prior and likelihood model.
- Sub-optimality gap and Generalization bound is still open.

# Acknowledgement

This work is done as the final project for the course DDA6201: Selected Topics in Data and Decision Analytics I. The author would like to thank the course instructor Prof. Hongyuan Zha on his helpful suggestions on the simulation part.

**Selected References**:


- Carl Doersch. 2016. Tutorial on Variational Autoencoders. ArXiv abs/1606.05908 (2016).
- Bradley Efron. 2019. Bayes, Oracle Bayes and Empirical Bayes. Statist. Sci. 34, 2 (05 2019), 177201. https://doi.org/10.1214/18- STS674
- Wenhua Jiang and Cun-Hui Zhang.2009. General maximum likelihood empirical Bayes estimation of normal means. Ann. Statist. 37, 4 (08 2009), 16471684. https://doi.org/10.1214/08- AOS638
- Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. 2016. Improved Variational Inference with Inverse Autoregressive Flow. In Advances in Neural Information Processing Systems 29.