Outlier-robust estimation of uncertain-input systems with applications to nonparametric FIR and Hammerstein models

Riccardo Sven Risuleo and Håkan Hjalmarsson

KTH - Royal Institute of Technology

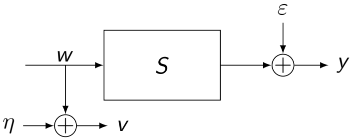
hjalmars@kth.se

Keio University, January 18, 2019

Overview

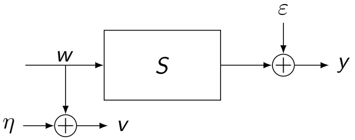
Systems with uncertain inputs

Systems with uncertain inputs

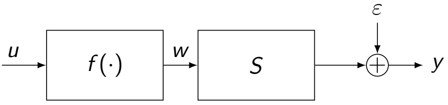


errors-in-variables

Systems with uncertain inputs

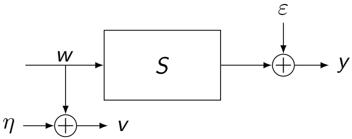


errors-in-variables

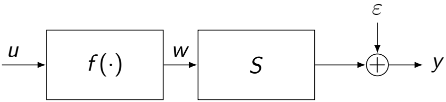


Hammerstein

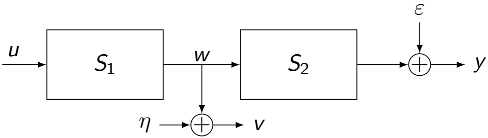
Systems with uncertain inputs



errors-in-variables

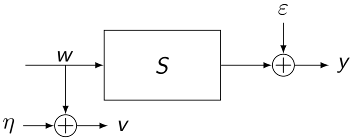


Hammerstein

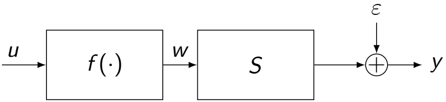


cascade

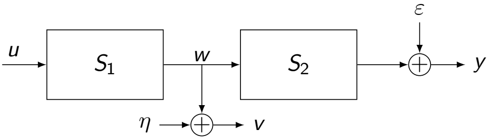
Systems with uncertain inputs



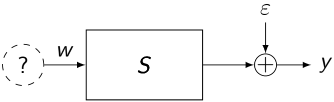
errors-in-variables



Hammerstein

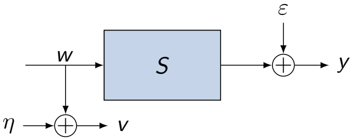


cascade

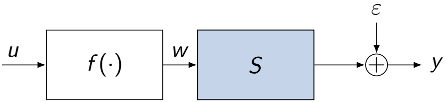


blind

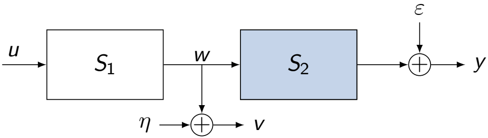
Systems with uncertain inputs



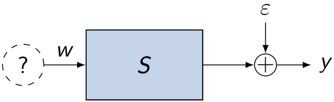
errors-in-variables



Hammerstein



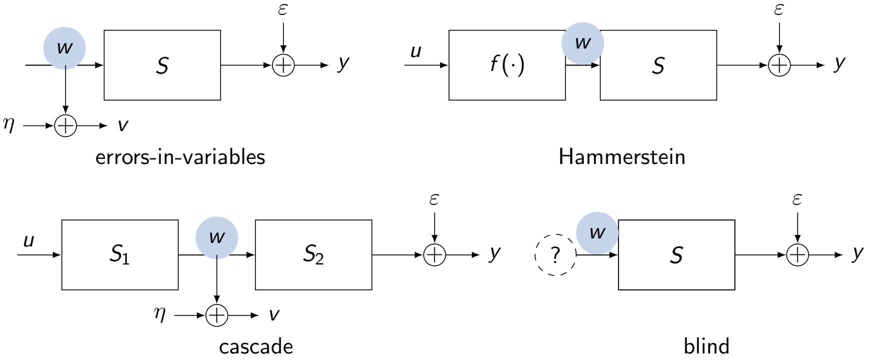
cascade



blind

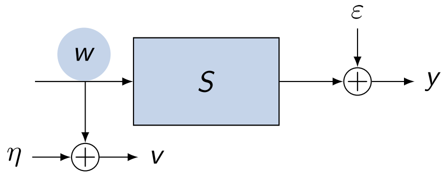
- A linear system S

Systems with uncertain inputs



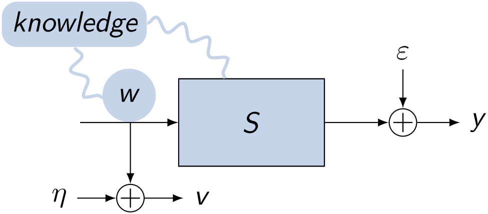
- A linear system S
- An unknown input signal w

The uncertain-input system



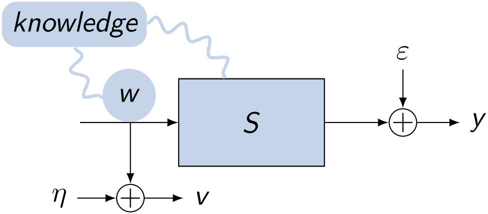
- Linear system S
- Unknown input w

The uncertain-input system



- Linear system S
- Unknown input w
- Prior knowledge

The uncertain-input system



- Linear system S
- Unknown input w
- Prior knowledge

How do we model these?

Gaussian processes

Gaussian processes

- Gaussian distribution over functions

$$f(\cdot) \sim \mathcal{N}\big(\mu(\cdot), K(\cdot, \cdot)\big)$$

Gaussian processes

- Gaussian distribution over functions

$$f(\cdot) \sim \mathcal{N}\big(\mu(\cdot), K(\cdot, \cdot)\big)$$

- The values of the function have a joint Gaussian distribution

$$\begin{bmatrix} f(x_1) \\ f(x_2) \\ f(x_3) \end{bmatrix} = \mathcal{N}\left(\begin{bmatrix} \mu(x_1) \\ \mu(x_2) \\ \mu(x_3) \end{bmatrix}, \begin{bmatrix} K(x_1, x_1) & K(x_1, x_2) & K(x_1, x_3) \\ K(x_2, x_1) & K(x_2, x_2) & K(x_2, x_3) \\ K(x_3, x_1) & K(x_3, x_2) & K(x_3, x_3) \end{bmatrix} \right)$$

Gaussian processes

- Gaussian distribution over functions

$$f(\cdot) \sim \mathcal{N}(\mu(\cdot), K(\cdot,\cdot))$$

- The values of the function have a joint Gaussian distribution

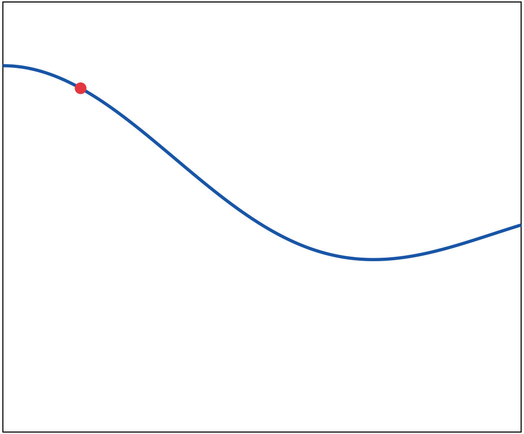
$$\begin{bmatrix} f(x_1) \\ f(x_2) \\ f(x_3) \end{bmatrix} = \mathcal{N}\left(\begin{bmatrix} \mu(x_1) \\ \mu(x_2) \\ \mu(x_3) \end{bmatrix}, \begin{bmatrix} K(x_1,x_1) & K(x_1,x_2) & K(x_1,x_3) \\ K(x_2,x_1) & K(x_2,x_2) & K(x_2,x_3) \\ K(x_3,x_1) & K(x_3,x_2) & K(x_3,x_3) \end{bmatrix} \right)$$

- Given values of the function $y = f(\tilde{x})$, we can estimate

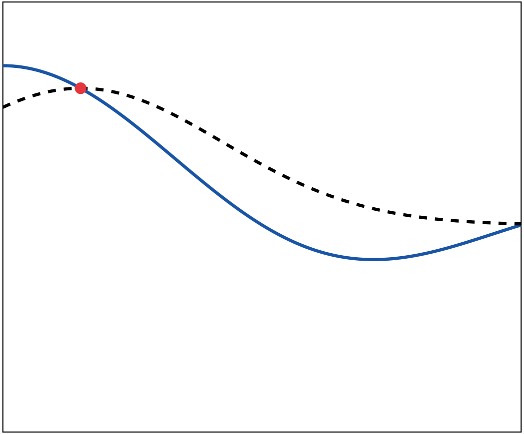
$$\hat{f}(x) = \mathbf{E}\{f(x)|y\} = \mu(x) + K(x,\tilde{x})\left[K(\tilde{x},\tilde{x})\right]^{-1}(y - \mu(\tilde{x}))$$

Gaussian processes

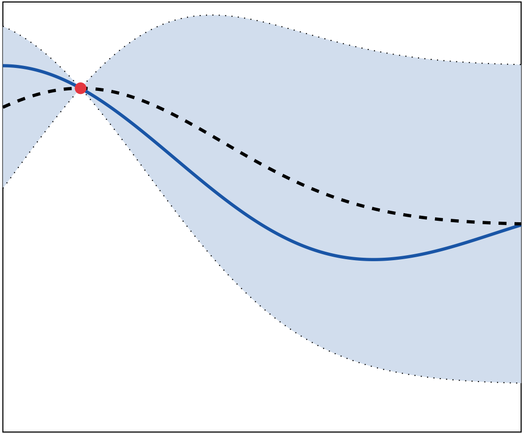
Gaussian processes



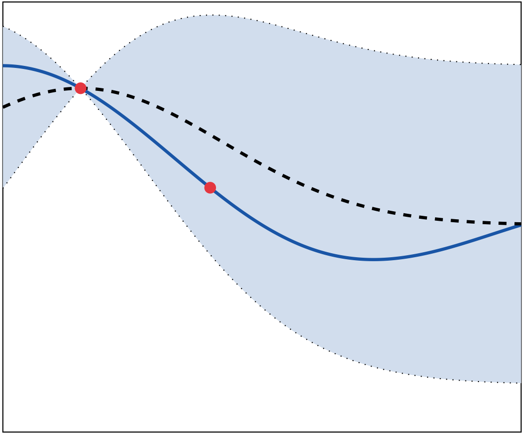
Gaussian processes



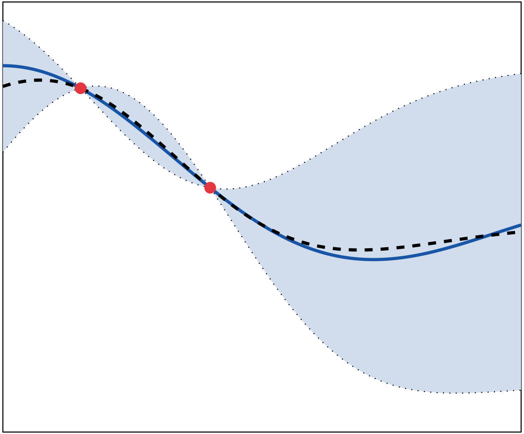
Gaussian processes



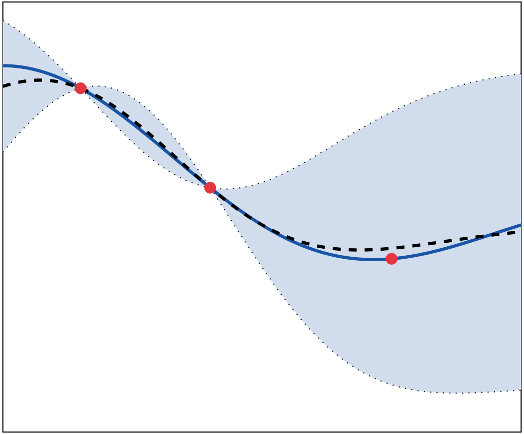
Gaussian processes



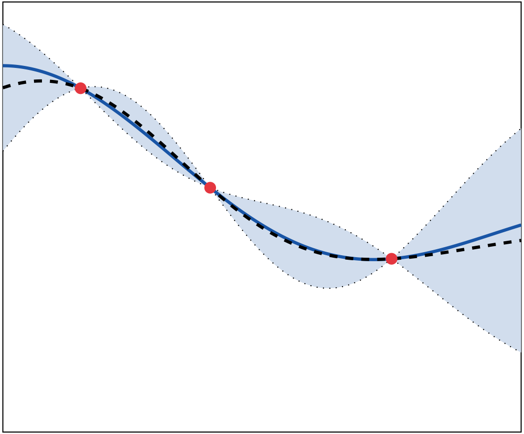
Gaussian processes



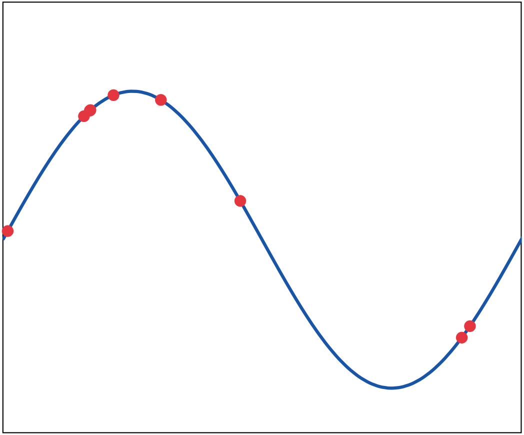
Gaussian processes



Gaussian processes

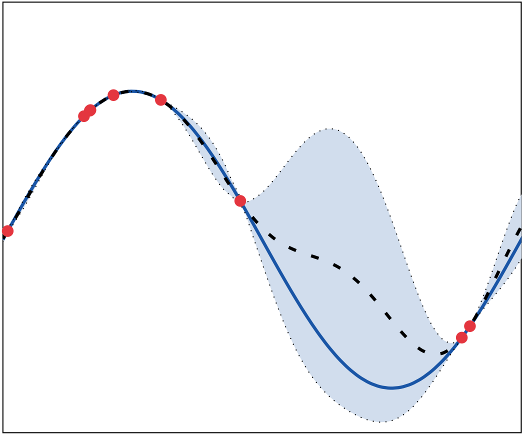


Using Gaussian processes to encode information



$$K(x_1, x_2) = e^{-\frac{1}{\theta}(x_1 - x_2)^2}$$

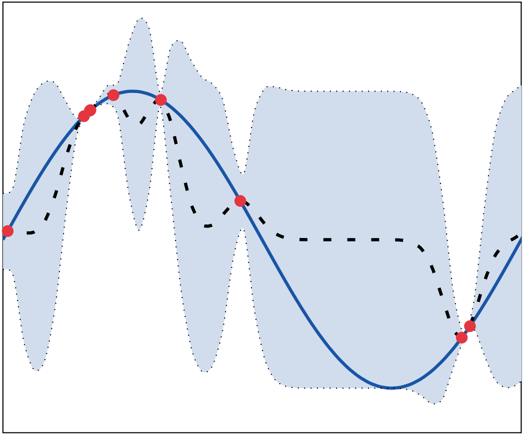
Using Gaussian processes to encode information



$$K(x_1, x_2) = e^{-\frac{1}{\theta}(x_1 - x_2)^2}$$

- $\theta = 0.1$

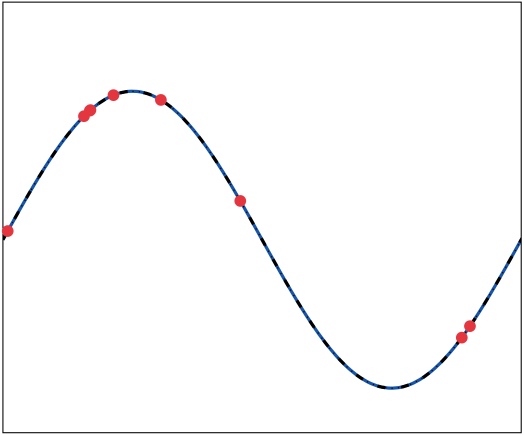
Using Gaussian processes to encode information



$$K(x_1, x_2) = e^{-\frac{1}{\theta}(x_1 - x_2)^2}$$

- $\theta = 0.1$
- $\theta = 0.01$

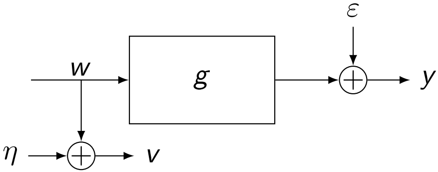
Using Gaussian processes to encode information



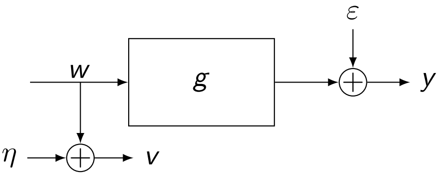
$$K(x_1, x_2) = e^{-\frac{1}{\theta}(x_1 - x_2)^2}$$

- $\theta = 0.1$
- $\theta = 0.01$
- $\theta = 1$

Modeling the uncertain input system



Modeling the uncertain input system

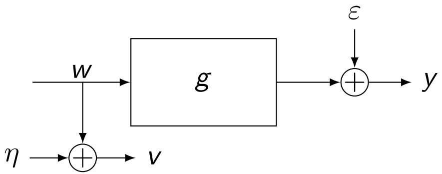


- Gaussian process prior for the impulse response

$$g \sim \mathcal{N}(\mu_g(\rho), K_g(\rho))$$

$$\left[\mu_g(\rho)\right]_i = \mathbf{E}\{g_i\} \qquad \left[K_g(\rho)\right]_{ij} = \mathbf{cov}\{g_i, g_j\}$$

Modeling the uncertain input system



- Gaussian process prior for the impulse response

$$g \sim \mathcal{N}(\mu_g(\rho), K_g(\rho))$$

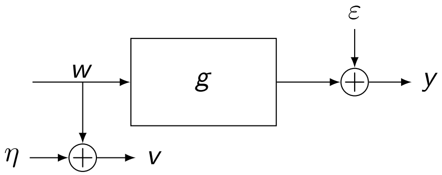
$$\left[\mu_g(\rho)\right]_i = \mathbf{E}\{g_i\} \qquad \left[K_g(\rho)\right]_{ij} = \mathbf{cov}\{g_i, g_j\}$$

- Gaussian process prior for the input

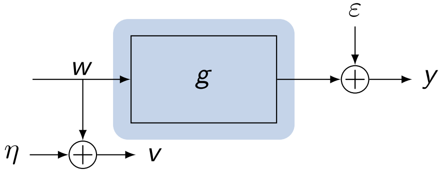
$$w \sim \mathcal{N}(\mu_w(\theta), K_w(\theta))$$

$$\left[\mu_w(\theta)\right]_i = \mathbf{E}\{w_i\}, \qquad \left[K_w(\theta)\right]_{ij} = \mathbf{cov}\{w_i, w_j\}$$

Measurement setup



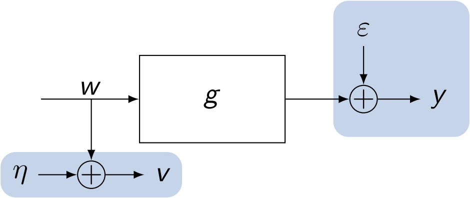
Measurement setup



- The system is stable and LTI

$$y_{\text{noiseless}} = w * g$$

Measurement setup



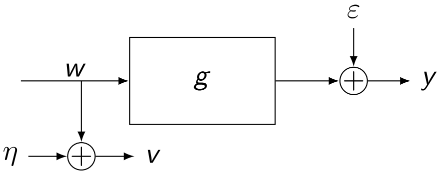
- The system is stable and LTI

$$y_{\text{noiseless}} = w * g$$

- The noises are additive, Gaussian, and white

$$v = w + \eta \qquad y = w * g + \varepsilon$$
$$\eta \sim \mathcal{N}(0, \sigma_v^2 I) \qquad \varepsilon \sim \mathcal{N}(0, \sigma_y^2 I)$$

The uncertain-input model



The uncertain-input model

$$\begin{cases} y = w * g + \varepsilon \\ v = w + \eta \\ \varepsilon \sim \mathcal{N}(0, \sigma_y^2 I) \\ \eta \sim \mathcal{N}(0, \sigma_v^2 I) \\ g \sim \mathcal{N}(\mu_g(\rho), K_g(\rho)) \\ w \sim \mathcal{N}(\mu_w(\theta), K_w(\theta)) \end{cases}$$

Inference in uncertain-input models

Inference in uncertain-input models

- Bayesian assumption

$$g \sim \mathcal{N}(\mu_g(\rho), K_g(\rho)) \qquad w \sim \mathcal{N}(\mu_w(\theta), K_w(\theta))$$

Inference in uncertain-input models

- Bayesian assumption

$$g \sim \mathcal{N}(\mu_g(\rho), K_g(\rho)) \qquad w \sim \mathcal{N}(\mu_w(\theta), K_w(\theta))$$

- We would like to have the conditional mean estimates

$$\hat{g} = \mathbf{E}\{g|v, y\} \qquad \hat{w} = \mathbf{E}\{w|v, y\}$$

Inference in uncertain-input models

- Bayesian assumption

$$g \sim \mathcal{N}(\mu_g(\rho), K_g(\rho)) \qquad w \sim \mathcal{N}(\mu_w(\theta), K_w(\theta))$$

- We would like to have the conditional mean estimates

$$\hat{g} = \mathbf{E}\{g|v,y\} \qquad \hat{w} = \mathbf{E}\{w|v,y\}$$

- We do not know the hyperparameters $\tau = \{\rho, \theta, \sigma_v^2, \sigma_y^2\}$!

$$\hat{g} = \hat{g}(\tau) \qquad \hat{w} = \hat{w}(\tau)$$

Inference in uncertain-input models

- Bayesian assumption

$$g \sim \mathcal{N}(\mu_g(\rho), K_g(\rho)) \qquad w \sim \mathcal{N}(\mu_w(\theta), K_w(\theta))$$

- We would like to have the conditional mean estimates

$$\hat{g} = \mathbf{E}\{g|v, y\} \qquad \hat{w} = \mathbf{E}\{w|v, y\}$$

- We do not know the hyperparameters $\tau = \{\rho, \theta, \sigma_v^2, \sigma_y^2\}$!

$$\hat{g} = \hat{g}(\tau) \qquad \hat{w} = \hat{w}(\tau)$$

We need to estimate them from data

Empirical Bayes

Empirical Bayes

- We choose the hyperparameters that maximize the *marginal likelihood*

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau)$$

Empirical Bayes

- We choose the hyperparameters that maximize the *marginal likelihood*

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau)$$

- Estimates:

$$\hat{g}(\hat{\tau}) = \mathbf{E}\{g|v, y; \hat{\tau}\} \qquad \hat{w}(\hat{\tau}) = \mathbf{E}\{w|v, y; \hat{\tau}\}$$

Is it really that simple?

Is it really that simple?

Short answer

Yes

Is it really that simple?

Short answer

Yes

Long answer

Yes, but...

Is it really that simple?

Short answer

Yes

Long answer

Yes, but...

- we need to calculate expected values

Is it really that simple?

Short answer

Yes

Long answer

Yes, but...

- we need to calculate expected values
- some distributions are not available in closed form

Is it really that simple?

Short answer

Yes

Long answer

Yes, but...

- we need to calculate expected values
- some distributions are not available in closed form
- we need to maximize the marginal likelihood

Is it really that simple?

Short answer

Yes

Long answer

Yes, but...

- we need to calculate expected values ← Monte Carlo
- some distributions are not available in closed form
- we need to maximize the marginal likelihood

Is it really that simple?

Short answer

Yes

Long answer

Yes, but...

- we need to calculate expected values $\leftarrow$ Monte Carlo
- some distributions are not available in closed form $\leftarrow$ Gibbs
- we need to maximize the marginal likelihood

Is it really that simple?

Short answer

Yes

Long answer

Yes, but...

- we need to calculate expected values $\leftarrow$ Monte Carlo
- some distributions are not available in closed form $\leftarrow$ Gibbs
- we need to maximize the marginal likelihood $\leftarrow$ EM

Calculating the posterior mean

- We need the posterior means

$$\mathbf{E}\left\{g|y,v;\tau\right\}$$

$$\mathbf{E}\left\{w|y,v;\tau\right\}$$

Calculating the posterior mean

- We need the posterior means

$$\mathbf{E}\left\{g|y,v;\tau\right\} = \int g\, p(g,w|y,v;\tau)\,\mathrm{d}w\,\mathrm{d}g$$
$$\mathbf{E}\left\{w|y,v;\tau\right\} = \int w\, p(g,w|y,v;\tau)\,\mathrm{d}g\,\mathrm{d}w$$

Calculating the posterior mean

- We need the posterior means

$$\mathbf{E}\left\{g|y,v;\tau\right\} = \int g\, p(g,w|y,v;\tau)\,\mathrm{d}w\,\mathrm{d}g$$

$$\mathbf{E}\left\{w|y,v;\tau\right\} = \int w\, p(g,w|y,v;\tau)\,\mathrm{d}g\,\mathrm{d}w$$

Calculating the posterior mean

- We need the posterior means

$$\mathbf{E}\left\{g|y,v;\tau\right\} = \int g\, p(g,w|y,v;\tau)\,\mathrm{d}w\,\mathrm{d}g$$

$$\mathbf{E}\left\{w|y,v;\tau\right\} = \int w\, p(g,w|y,v;\tau)\,\mathrm{d}g\,\mathrm{d}w$$

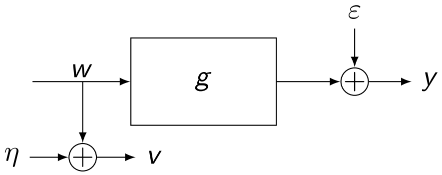
The joint distribution is problematic to compute!

Calculating the posterior mean

- We need the posterior means

$$\mathbf{E}\{g|y,v;\tau\} = \int g\, p(g,w|y,v;\tau)\,\mathrm{d}w\,\mathrm{d}g$$
$$\mathbf{E}\{w|y,v;\tau\} = \int w\, p(g,w|y,v;\tau)\,\mathrm{d}g\,\mathrm{d}w$$

The joint distribution is problematic to compute!



Monte Carlo integration

Monte Carlo integration

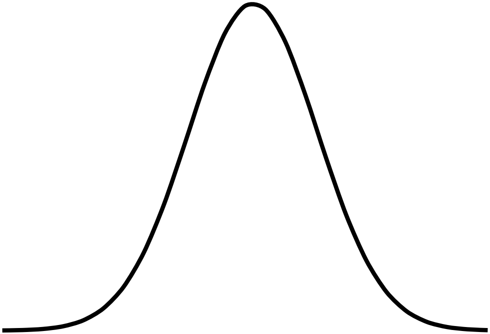
- We make a particle approximation of the distribution

$$p(g, w|y, v; \tau) \approx \frac{1}{M} \sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)})$$

Monte Carlo integration

- We make a particle approximation of the distribution

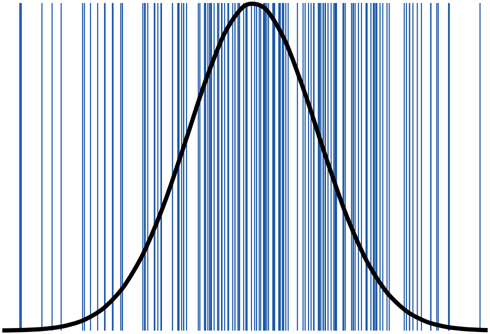
$$p(g, w|y, v; \tau) \approx \frac{1}{M} \sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)})$$



Monte Carlo integration

- We make a particle approximation of the distribution

$$p(g, w|y, v; \tau) \approx \frac{1}{M} \sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)})$$



Monte Carlo integration

- We make a particle approximation of the distribution

$$p(g, w|y, v; \tau) \approx \frac{1}{M} \sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)})$$

- We approximate the posterior mean

$$\mathbf{E}\{g|y, v\} = \int g\, p(g, w|y, v; \tau)\, \mathrm{d}w\, \mathrm{d}g$$

Monte Carlo integration

- We make a particle approximation of the distribution

$$p(g, w|y, v; \tau) \approx \frac{1}{M} \sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)})$$

- We approximate the posterior mean

$$\begin{aligned} \mathbf{E}\{g|y, v\} &= \int g\, p(g, w|y, v; \tau)\, \mathrm{d}w\, \mathrm{d}g \\ &\approx \int g \frac{1}{M} \sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)})\, \mathrm{d}w\, \mathrm{d}g \end{aligned}$$

Monte Carlo integration

- We make a particle approximation of the distribution

$$p(g, w|y, v; \tau) \approx \frac{1}{M}\sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)})$$

- We approximate the posterior mean

$$\begin{aligned} \mathbf{E}\left\{g|y, v\right\} &= \int g\, p(g, w|y, v; \tau)\,\mathrm{d}w\,\mathrm{d}g \\ &\approx \int g \frac{1}{M}\sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)})\,\mathrm{d}w\,\mathrm{d}g = \frac{1}{M}\sum_{j=1}^{M} \bar{g}^{(j)} \end{aligned}$$

Monte Carlo integration

- We make a particle approximation of the distribution

$$p(g, w|y, v; \tau) \approx \frac{1}{M} \sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)})$$

- We approximate the posterior mean

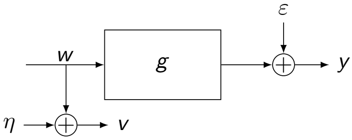
$$\begin{aligned} \mathbf{E}\{g|y, v\} &= \int g\, p(g, w|y, v; \tau)\, \mathrm{d}w\, \mathrm{d}g \\ &\approx \int g \frac{1}{M} \sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)})\, \mathrm{d}w\, \mathrm{d}g = \frac{1}{M} \sum_{j=1}^{M} \bar{g}^{(j)} \end{aligned}$$

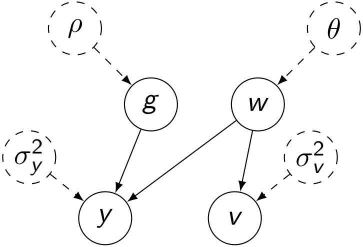
- Similarly

$$\mathbf{E}\{w|y, v\} = \int w\, p(g, w|y, v; \tau)\, \mathrm{d}g\, \mathrm{d}w \approx \frac{1}{M} \sum_{j=1}^{M} \bar{w}^{(j)}$$

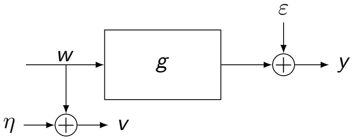
Particle approximation

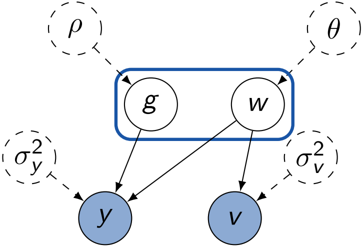
Particle approximation





Particle approximation

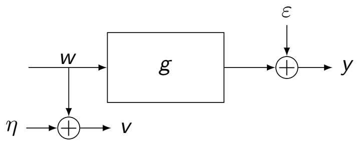


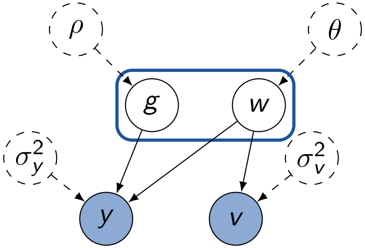


- We want samples from a joint distribution

$$p(g, w | y, v; \tau)$$

Particle approximation

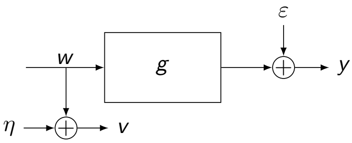


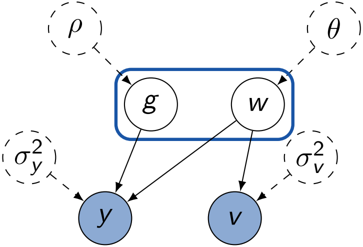


- We want samples from a joint distribution

$$p(g, w|y, v; \tau) \leftarrow \textit{difficult to evaluate!}$$

Particle approximation





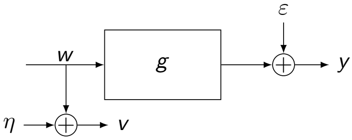
- We want samples from a joint distribution

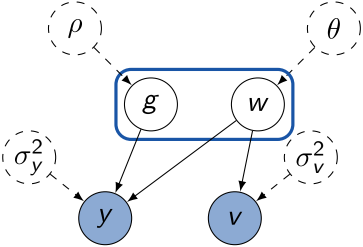
$$p(g, w|y, v; \tau) \leftarrow \textit{difficult to evaluate!}$$

- but

$$p(g|y, w; \tau)$$

Particle approximation





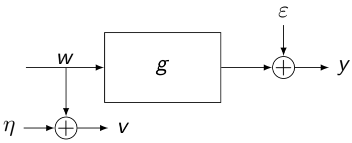
- We want samples from a joint distribution

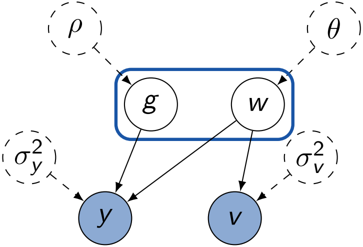
$$p(g, w|y, v; \tau) \leftarrow \textit{difficult to evaluate!}$$

- but

$$p(g|y, w; \tau) \leftarrow \textit{Gaussian!}$$

Particle approximation





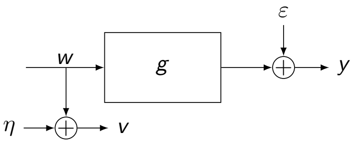
- We want samples from a joint distribution

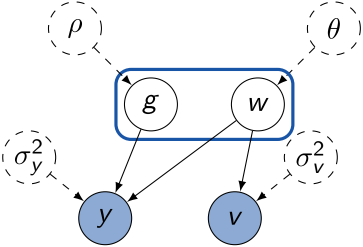
$$p(g, w|y, v; \tau) \leftarrow \textit{difficult to evaluate!}$$

- but

$$p(g|y, w; \tau) \leftarrow \textit{Gaussian!}$$
$$p(w|y, v, g; \tau)$$

Particle approximation





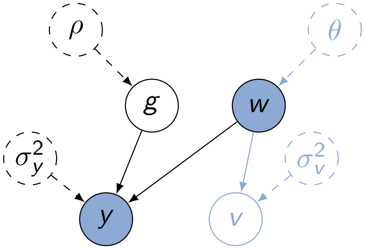
- We want samples from a joint distribution

$$p(g, w | y, v; \tau) \leftarrow \textit{difficult to evaluate!}$$

- but

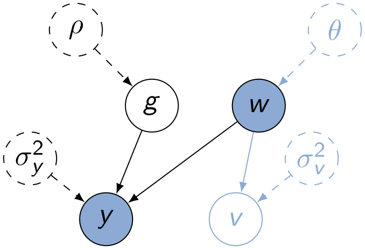
$$p(g | y, w; \tau) \leftarrow \textit{Gaussian!}$$
$$p(w | y, v, g; \tau) \leftarrow \textit{Gaussian!}$$

The Gibbs sampler



$$g|y, w \sim \mathcal{N}$$

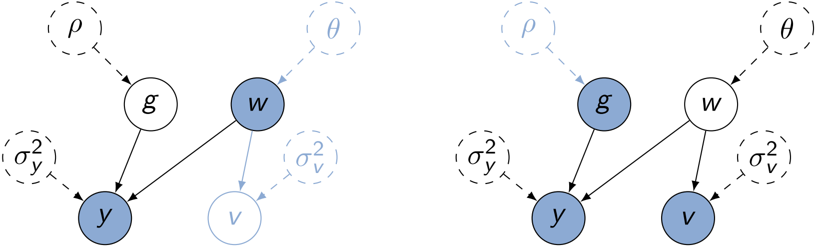
The Gibbs sampler



$g|y, w \sim \mathcal{N}$

$$\bar{g}^{(k+1)} \sim p(g|y, \bar{w}^{(k)}; \hat{\rho}, \hat{\sigma}_y^2),$$

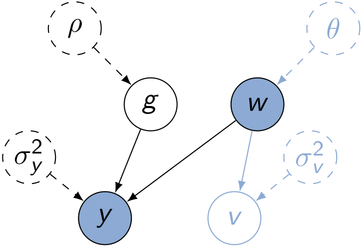
The Gibbs sampler



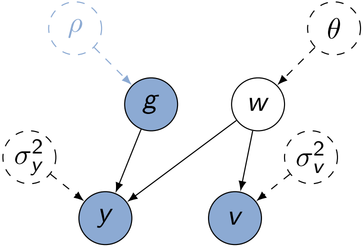
$g|y, w \sim \mathcal{N}$ $\quad$ $w|g, v, y \sim \mathcal{N}$

$$\bar{g}^{(k+1)} \sim p(g|y, \bar{w}^{(k)}; \hat{\rho}, \hat{\sigma}_y^2),$$

The Gibbs sampler



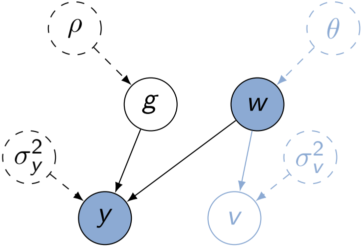
$g|y, w \sim \mathcal{N}$



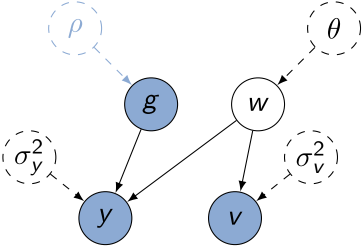
$w|g, v, y \sim \mathcal{N}$

$$\bar{g}^{(k+1)} \sim p(g|y, \bar{w}^{(k)}\,; \hat{\rho}, \hat{\sigma}_y^2),$$
$$\bar{w}^{(k+1)} \sim p(w|y, v, \bar{g}^{(k+1)}\,; \hat{\theta}, \hat{\sigma}^2)$$

The Gibbs sampler



$g|y, w \sim \mathcal{N}$

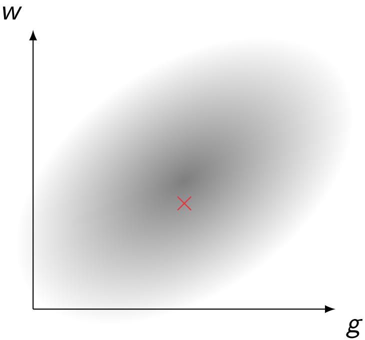


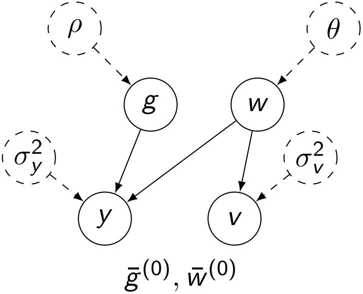
$w|g, v, y \sim \mathcal{N}$

$$\bar{g}^{(k+1)} \sim p(g|y, \bar{w}^{(k)}; \hat{\rho}, \hat{\sigma}_y^2),$$
$$\bar{w}^{(k+1)} \sim p(w|y, v, \bar{g}^{(k+1)}; \hat{\theta}, \hat{\sigma}^2)$$

$(\bar{w}^{(k)}, \bar{g}^{(k)})$ are samples from $p(g, w|y, v; \hat{\tau})$

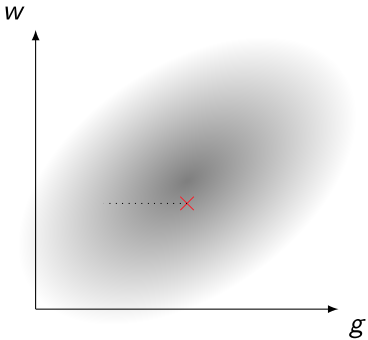
Gibbs sampling the UI model

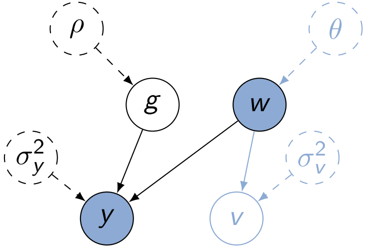




$$\left\{ (\bar{g}^{(0)}, \bar{w}^{(0)}) \right\}$$

Gibbs sampling the UI model

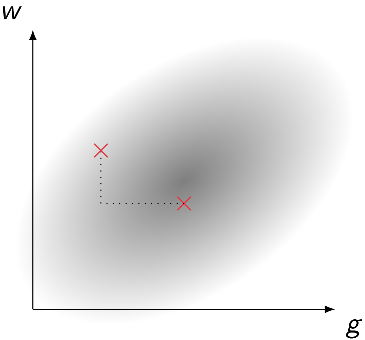


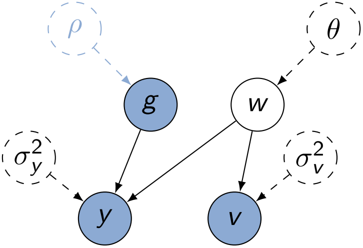


$$g | y, w \sim \mathcal{N}$$

$$\left\{ (\bar{g}^{(0)}, \bar{w}^{(0)}) \quad (\bar{g}^{(1)}, \qquad) \qquad \right\}$$

Gibbs sampling the UI model

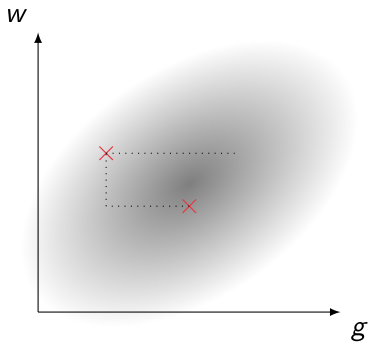


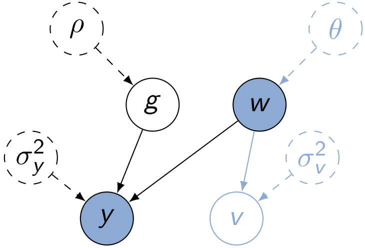


$$w | g, v, y \sim \mathcal{N}$$

$$\left\{ (\bar{g}^{(0)}, \bar{w}^{(0)}) \quad (\bar{g}^{(1)}, \bar{w}^{(1)}) \right\}$$

Gibbs sampling the UI model

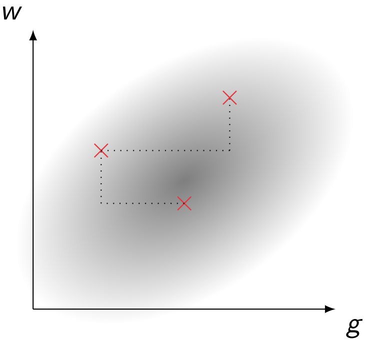


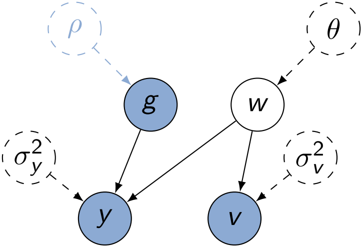


$g|y, w \sim \mathcal{N}$

$$\left\{ (\bar{g}^{(0)}, \bar{w}^{(0)}) \quad (\bar{g}^{(1)}, \bar{w}^{(1)}) \quad (\bar{g}^{(2)}, \qquad) \right\}$$

Gibbs sampling the UI model

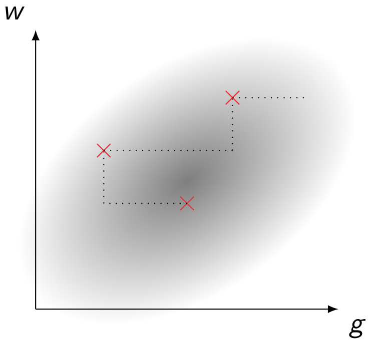


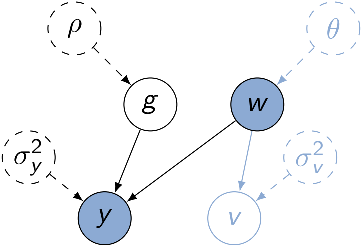


$$w | g, v, y \sim \mathcal{N}$$

$$\left\{ (\bar{g}^{(0)}, \bar{w}^{(0)}) \quad (\bar{g}^{(1)}, \bar{w}^{(1)}) \quad (\bar{g}^{(2)}, \bar{w}^{(2)}) \right\}$$

Gibbs sampling the UI model

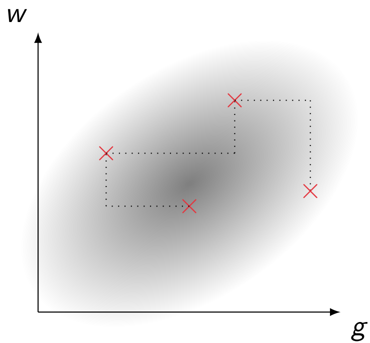


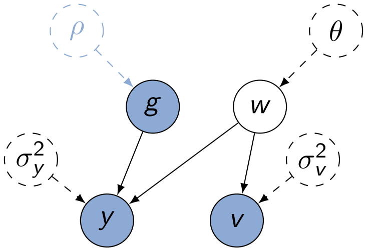


$$g | y, w \sim \mathcal{N}$$

$$\left\{ (\bar{g}^{(0)}, \bar{w}^{(0)}) \quad (\bar{g}^{(1)}, \bar{w}^{(1)}) \quad (\bar{g}^{(2)}, \bar{w}^{(2)}) \quad (\bar{g}^{(3)}, \qquad) \right\}$$

Gibbs sampling the UI model

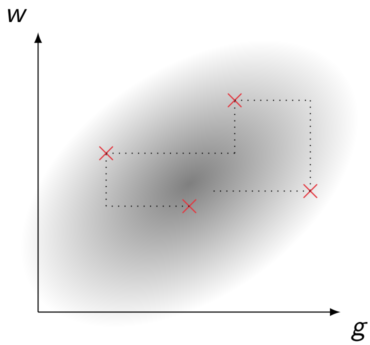


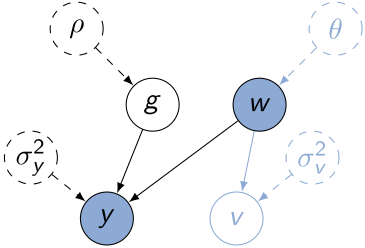


$$w|g, v, y \sim \mathcal{N}$$

$$\left\{ (\bar{g}^{(0)}, \bar{w}^{(0)}) \quad (\bar{g}^{(1)}, \bar{w}^{(1)}) \quad (\bar{g}^{(2)}, \bar{w}^{(2)}) \quad (\bar{g}^{(3)}, \bar{w}^{(3)}) \qquad \right\}$$

Gibbs sampling the UI model

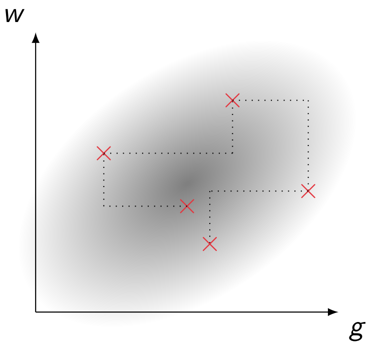


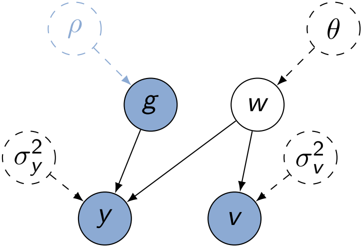


$g | y, w \sim \mathcal{N}$

$$\left\{ (\bar{g}^{(0)}, \bar{w}^{(0)}) \quad (\bar{g}^{(1)}, \bar{w}^{(1)}) \quad (\bar{g}^{(2)}, \bar{w}^{(2)}) \quad (\bar{g}^{(3)}, \bar{w}^{(3)}) \quad (\bar{g}^{(4)}, \qquad) \right\}$$

Gibbs sampling the UI model





$$w | g, v, y \sim \mathcal{N}$$

$$\left\{ (\bar{g}^{(0)}, \bar{w}^{(0)}) \quad (\bar{g}^{(1)}, \bar{w}^{(1)}) \quad (\bar{g}^{(2)}, \bar{w}^{(2)}) \quad (\bar{g}^{(3)}, \bar{w}^{(3)}) \quad (\bar{g}^{(4)}, \bar{w}^{(4)}) \right\}$$

Maximizing the marginal likelihood

- We want to compute the *marginal likelihood estimate*

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau)$$

Maximizing the marginal likelihood

- We want to compute the *marginal likelihood estimate*

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau)$$

- but

$$p(y, v; \tau) = \int p(y, v, g, w; \tau) \, \mathrm{d}g \, \mathrm{d}w$$

Maximizing the marginal likelihood

- We want to compute the *marginal likelihood estimate*

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau)$$

- but

$$\begin{aligned} p(y, v; \tau) &= \int p(y, v, g, w; \tau) \, \mathrm{d}g \, \mathrm{d}w \\ &= \int p(y, v | g, w; \tau) p(g, w; \tau) \, \mathrm{d}g \, \mathrm{d}w \end{aligned}$$

Maximizing the marginal likelihood

- We want to compute the *marginal likelihood estimate*

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau)$$

- but

$$\begin{aligned} p(y, v; \tau) &= \int p(y, v, g, w; \tau) \, \mathrm{d}g \, \mathrm{d}w \\ &= \int p(y, v | g, w; \tau) p(g, w; \tau) \, \mathrm{d}g \, \mathrm{d}w \end{aligned}$$

A maximum likelihood problem with missing data!

EM for marginal likelihood estimation

- Maximum likelihood problem with missing data

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau) = \arg\max_{\tau} \int p(y, v, g, w; \tau)\, \mathrm{d}g\, \mathrm{d}w$$

EM for marginal likelihood estimation

- Maximum likelihood problem with missing data

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau) = \arg\max_{\tau} \int p(y, v, g, w; \tau) \, \mathrm{d}g \, \mathrm{d}w$$

- We can use the EM-method!

$$\begin{aligned} \text{E-step} \quad & Q(\tau, \tau^{(k)}) = \mathbf{E}\left\{\log p(y, v, g, w; \tau)\right\} \\ & \qquad \text{w.r.t. } p(g, w | y, v; \hat{\tau}^{(k)}) \\ \text{M-step} \quad & \tau^{(k+1)} = \arg\max Q(\tau, \tau^{(k)}) \end{aligned}$$

EM for marginal likelihood estimation

- Maximum likelihood problem with missing data

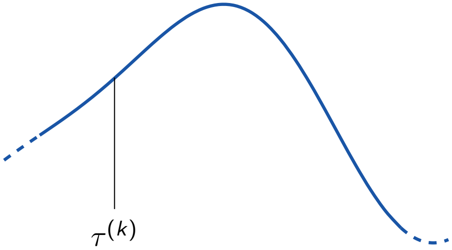
$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau) = \arg\max_{\tau} \int p(y, v, g, w; \tau)\, \mathrm{d}g\, \mathrm{d}w$$

- We can use the EM-method!

E-step $\quad Q(\tau, \tau^{(k)}) = \mathbf{E}\left\{\log p(y, v, g, w; \tau)\right\}$

$\qquad$ w.r.t. $p(g, w | y, v; \hat{\tau}^{(k)})$

M-step $\quad \tau^{(k+1)} = \arg\max Q(\tau, \tau^{(k)})$



EM for marginal likelihood estimation

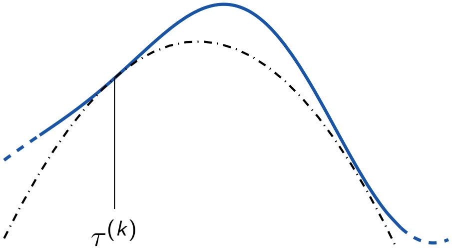
- Maximum likelihood problem with missing data

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau) = \arg\max_{\tau} \int p(y, v, g, w; \tau)\,\mathrm{d}g\,\mathrm{d}w$$

- We can use the EM-method!

E-step $\quad Q(\tau, \tau^{(k)}) = \mathbf{E}\left\{\log p(y, v, g, w; \tau)\right\}$ w.r.t. $p(g, w | y, v; \hat{\tau}^{(k)})$

M-step $\quad \tau^{(k+1)} = \arg\max Q(\tau, \tau^{(k)})$



EM for marginal likelihood estimation

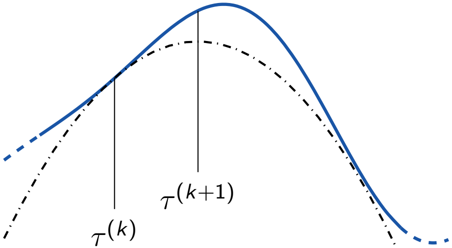
- Maximum likelihood problem with missing data

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau) = \arg\max_{\tau} \int p(y, v, g, w; \tau)\, \mathrm{d}g\, \mathrm{d}w$$

- We can use the EM-method!

E-step $\quad Q(\tau, \tau^{(k)}) = \mathbf{E}\{\log p(y, v, g, w; \tau)\}$ w.r.t. $p(g, w | y, v; \hat{\tau}^{(k)})$

M-step $\quad \tau^{(k+1)} = \arg\max Q(\tau, \tau^{(k)})$



EM for marginal likelihood estimation

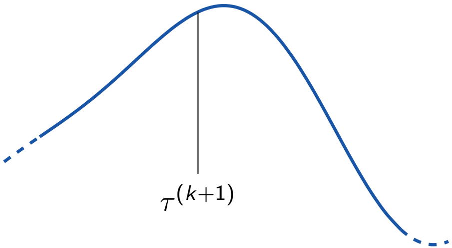
- Maximum likelihood problem with missing data

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau) = \arg\max_{\tau} \int p(y, v, g, w; \tau) \, \mathrm{d}g \, \mathrm{d}w$$

- We can use the EM-method!

E-step $\quad Q(\tau, \tau^{(k)}) = \mathbf{E}\{\log p(y, v, g, w; \tau)\}$ w.r.t. $p(g, w | y, v; \hat{\tau}^{(k)})$

M-step $\quad \tau^{(k+1)} = \arg\max Q(\tau, \tau^{(k)})$



EM for marginal likelihood estimation

- Maximum likelihood problem with missing data

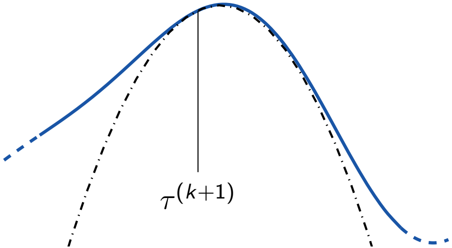
$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau) = \arg\max_{\tau} \int p(y, v, g, w; \tau)\, \mathrm{d}g\, \mathrm{d}w$$

- We can use the EM-method!

E-step $\quad Q(\tau, \tau^{(k)}) = \mathbf{E}\{\log p(y, v, g, w; \tau)\}$

$\qquad$ w.r.t. $p(g, w | y, v; \hat{\tau}^{(k)})$

M-step $\quad \tau^{(k+1)} = \arg\max Q(\tau, \tau^{(k)})$



EM for marginal likelihood estimation

- Maximum likelihood problem with missing data

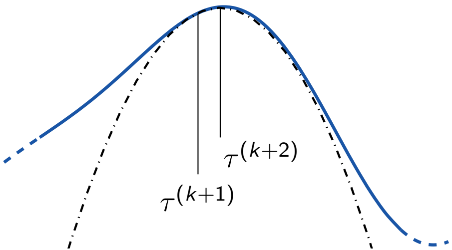
$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau) = \arg\max_{\tau} \int p(y, v, g, w; \tau)\, \mathrm{d}g\, \mathrm{d}w$$

- We can use the EM-method!

E-step $\quad Q(\tau, \tau^{(k)}) = \mathbf{E}\left\{\log p(y, v, g, w; \tau)\right\}$

$\quad\quad$ w.r.t. $p(g, w | y, v; \hat{\tau}^{(k)})$

M-step $\quad \tau^{(k+1)} = \arg\max Q(\tau, \tau^{(k)})$



EM for marginal likelihood estimation

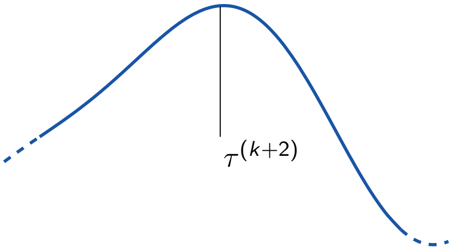
- Maximum likelihood problem with missing data

$$\hat{\tau} = \arg\max_{\tau} p(y, v; \tau) = \arg\max_{\tau} \int p(y, v, g, w; \tau)\, \mathrm{d}g\, \mathrm{d}w$$

- We can use the EM-method!

E-step $\quad Q(\tau, \tau^{(k)}) = \mathbf{E}\{\log p(y, v, g, w; \tau)\}$ w.r.t. $p(g, w | y, v; \hat{\tau}^{(k)})$

M-step $\quad \tau^{(k+1)} = \arg\max Q(\tau, \tau^{(k)})$



Monte Carlo Expectation Maximization

- The E-step is difficult to evaluate

$$Q(\tau, \tau^{(k)}) = \mathbf{E}\left\{\log p(y, v, g, w; \tau)\right\}$$

Monte Carlo Expectation Maximization

- The E-step is difficult to evaluate

$$
\begin{aligned}
Q(\tau, \tau^{(k)}) &= \mathbf{E}\left\{\log p(y, v, g, w; \tau)\right\} \\
&= \int \log\left[p(y, v, g, w; \tau)\right] p(g, w | y, v; \tau^{(k)}) \mathrm{d}g \, \mathrm{d}w
\end{aligned}
$$

Monte Carlo Expectation Maximization

- The E-step is difficult to evaluate

$$
\begin{aligned}
Q(\tau, \tau^{(k)}) &= \mathbf{E}\left\{\log p(y, v, g, w; \tau)\right\} \\
&= \int \log\left[p(y, v, g, w; \tau)\right] p(g, w | y, v; \tau^{(k)}) \mathrm{d}g \, \mathrm{d}w
\end{aligned}
$$

Monte Carlo Expectation Maximization

- The E-step is difficult to evaluate

$$
\begin{aligned}
Q(\tau, \tau^{(k)}) &= \mathbf{E}\left\{\log p(y, v, g, w; \tau)\right\} \\
&= \int \log\left[p(y, v, g, w; \tau)\right] p(g, w | y, v; \tau^{(k)}) \mathrm{d}g\, \mathrm{d}w
\end{aligned}
$$

- We can use the same particle approximation as before!

Monte Carlo Expectation Maximization

- The E-step is difficult to evaluate

$$Q(\tau, \tau^{(k)}) = \mathbf{E}\left\{\log p(y, v, g, w; \tau)\right\}$$
$$= \int \log\left[p(y, v, g, w; \tau)\right] p(g, w | y, v; \tau^{(k)}) \mathrm{d}g\, \mathrm{d}w$$

- We can use the same particle approximation as before!

$$Q(\tau, \tau^{(k)})$$
$$\approx \int \log\left[p(y, v, g, w; \tau)\right] \frac{1}{M} \sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)}) \mathrm{d}g\, \mathrm{d}w$$

Monte Carlo Expectation Maximization

- The E-step is difficult to evaluate

$$Q(\tau, \tau^{(k)}) = \mathbf{E}\left\{\log p(y, v, g, w; \tau)\right\}$$
$$= \int \log\left[p(y, v, g, w; \tau)\right] p(g, w | y, v; \tau^{(k)}) \mathrm{d}g \, \mathrm{d}w$$

- We can use the same particle approximation as before!

$$Q(\tau, \tau^{(k)})$$
$$\approx \int \log\left[p(y, v, g, w; \tau)\right] \frac{1}{M} \sum_{j=1}^{M} \delta(g - \bar{g}^{(j)}, w - \bar{w}^{(j)}) \mathrm{d}g \, \mathrm{d}w$$
$$= \frac{1}{M} \sum_{j=1}^{M} \log\left[p(y, v, \bar{g}^{(j)}, \bar{w}^{(j)}; \tau)\right]$$

Monte Carlo Expectation Maximization

MCEM for hyperparameter estimation

Monte Carlo Expectation Maximization

MCEM for hyperparameter estimation

MC-step $\{g^{(j)}, w^{(j)}\}_{j=1}^{M}$ = particle approximation of $p(g, w|y, v; \tau^{(k)})$

Monte Carlo Expectation Maximization

MCEM for hyperparameter estimation

MC-step $\{g^{(j)}, w^{(j)}\}_{j=1}^{M} =$ particle approximation of $p(g, w|y, v; \tau^{(k)})$

E-step $$\bar{Q}(\tau, \tau^{(k)}) = \frac{1}{M} \sum_{j=1}^{M} \log p(y, v, g^{(j)}, w^{(j)}; \tau)$$

Monte Carlo Expectation Maximization

MCEM for hyperparameter estimation

MC-step $\{g^{(j)}, w^{(j)}\}_{j=1}^{M} = \text{particle approximation of } p(g, w | y, v; \tau^{(k)})$

E-step $$\bar{Q}(\tau, \tau^{(k)}) = \frac{1}{M} \sum_{j=1}^{M} \log p(y, v, g^{(j)}, w^{(j)}; \tau)$$

M-step $$\tau^{(k+1)} = \arg\max \bar{Q}(\tau, \tau^{(k)})$$

Monte Carlo inference in UI models

Monte Carlo inference in UI models

- MCEM for hyperparameter estimation

MC-step $\{g^{(j)}, w^{(j)}\}_{j=1}^{M} = \textsc{GibbsSampler}(\tau^{(k)})$

E-step $$\bar{Q}(\tau, \tau^{(k)}) = \frac{1}{M} \sum_{j=1}^{M} \log p(y, v, g^{(j)}, w^{(j)}; \tau)$$

M-step $$\tau^{(k+1)} = \arg\max \bar{Q}(\tau, \tau^{(k)})$$

Monte Carlo inference in UI models

- MCEM for hyperparameter estimation

MC-step $$\{g^{(j)}, w^{(j)}\}_{j=1}^{M} = \textsc{GibbsSampler}(\tau^{(k)})$$

E-step $$\bar{Q}(\tau, \tau^{(k)}) = \frac{1}{M}\sum_{j=1}^{M} \log p(y, v, g^{(j)}, w^{(j)}; \tau)$$

M-step $$\tau^{(k+1)} = \arg\max \bar{Q}(\tau, \tau^{(k)})$$

- Posterior means

$$\{\bar{g}^{(j)}, \bar{w}^{(j)}\}_{j=1}^{M} = \textsc{GibbsSampler}(\hat{\tau})$$

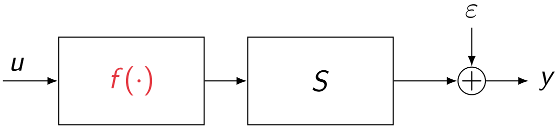
$$\mathbf{E}\{g|y, v\} \approx \frac{1}{M}\sum_{j=1}^{M} \bar{g}^{(j)}$$

$$\mathbf{E}\{w|y, v\} \approx \frac{1}{M}\sum_{j=1}^{M} \bar{w}^{(j)}$$

Example: Hammerstein models

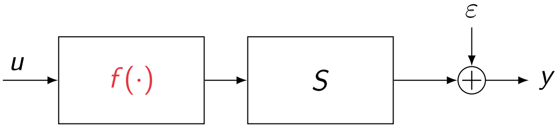
Example: Hammerstein models

- Nonparametric Hammerstein model



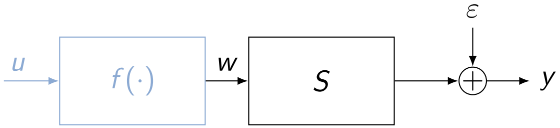
Example: Hammerstein models

- Nonparametric Hammerstein model



Example: Hammerstein models

- Nonparametric Hammerstein model

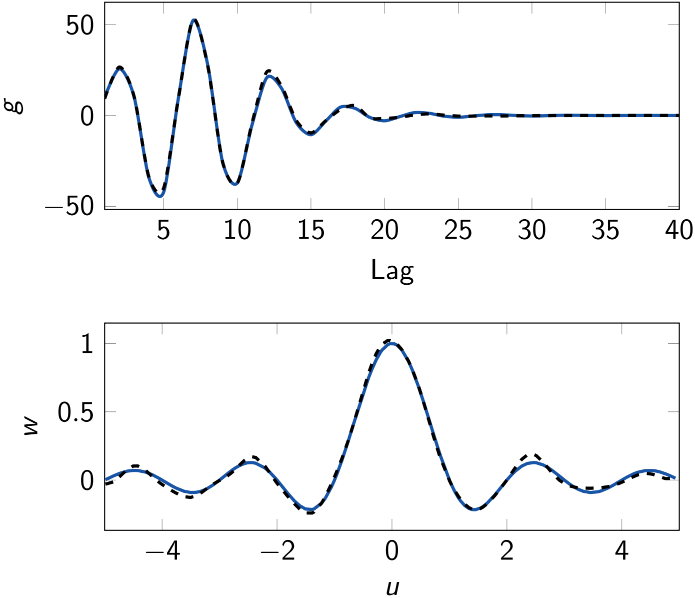


- uncertain-input model with

$$g \sim \mathcal{N}\big(0, K_g(\rho)\big) \quad w \sim \mathcal{N}\big(0, K_w(\theta)\big)$$

$$K_g(\rho)_{i,j} = \underbrace{\rho_1^{\max(i,j)}}_{\text{Stable-spline kernel}} \qquad [K_w(\theta)]_{ij} = \exp\left[-\frac{1}{\theta}(u_i - u_j)^2\right]$$

Example: Hammerstein models



$n_{\mathrm{lti}} = 40,\ N = 400,\ \mathrm{SNR} = 10,$ ▬▬ = true, - - - = estimated

Under the hood

- MCEM for hyperparameter estimation

Under the hood

- MCEM for hyperparameter estimation
  1. MC Step: Gibbs sampler
     - Reduce the covariance matrices with SVD (Bad conditioning!)
     - Draw g and w
     - Discard 200 burn-in particles
     - Generate 500 particles for w and g

Under the hood

- MCEM for hyperparameter estimation
  1. MC Step: Gibbs sampler
     - Reduce the covariance matrices with SVD (Bad conditioning!)
     - Draw g and w
     - Discard 200 burn-in particles
     - Generate 500 particles for w and g
  2. E-step and M-step available in (almost) closed form

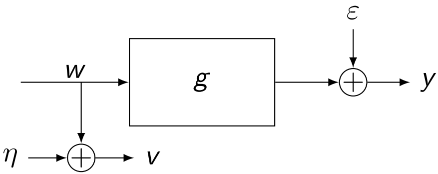
Under the hood

- MCEM for hyperparameter estimation
  1. MC Step: Gibbs sampler
     - Reduce the covariance matrices with SVD (Bad conditioning!)
     - Draw g and w
     - Discard 200 burn-in particles
     - Generate 500 particles for w and g
  2. E-step and M-step available in (almost) closed form
  3. Iterate until convergence

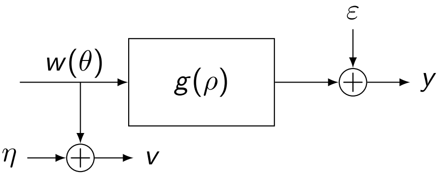
Under the hood

- MCEM for hyperparameter estimation
  1. MC Step: Gibbs sampler
     - Reduce the covariance matrices with SVD (Bad conditioning!)
     - Draw g and w
     - Discard 200 burn-in particles
     - Generate 500 particles for w and g
  2. E-step and M-step available in (almost) closed form
  3. Iterate until convergence
- Run new Gibbs sampler
  1. Reduce the covariance matrices with SVD (Bad conditioning!)
  2. Discard 500 burn-in particles
  3. Generate 1000 particles for w and g

Special classes of systems



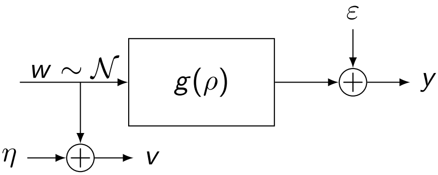
Special classes of systems



- PIPS: parametric-input parametric-system models

$$K_g(\rho) = 0 \qquad K_w(\theta) = 0$$

Special classes of systems



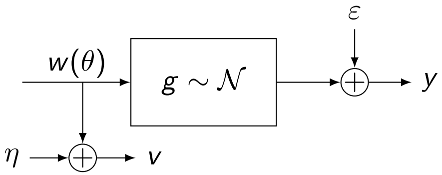
- PIPS: parametric-input parametric-system models

$$K_g(\rho) = 0 \qquad K_w(\theta) = 0$$

- GIPS: Gaussian-input parametric-system models

$$K_g(\rho) = 0$$

Special classes of systems



- PIPS: parametric-input parametric-system models

$$K_g(\rho) = 0 \qquad K_w(\theta) = 0$$

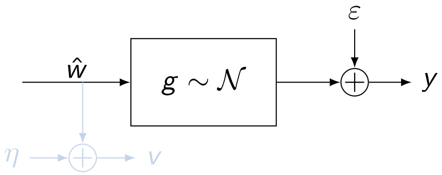
- GIPS: Gaussian-input parametric-system models

$$K_g(\rho) = 0$$

- PIGS: parametric-input Gaussian-system models

$$K_w(\theta) = 0$$

Special classes of systems



- PIPS: parametric-input parametric-system models

$$K_g(\rho) = 0 \qquad K_w(\theta) = 0$$

- GIPS: Gaussian-input parametric-system models

$$K_g(\rho) = 0$$

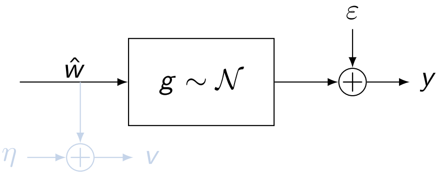
- PIGS: parametric-input Gaussian-system models

$$K_w(\theta) = 0$$

- EIGS: Estimated-input Gaussian-system models

$$w = \hat{w}$$

Special classes of systems



- PIPS: parametric-input parametric-system models

$$K_g(\rho) = 0 \qquad K_w(\theta) = 0 \leftarrow$$ ML and PEM!

- GIPS: Gaussian-input parametric-system models

$$K_g(\rho) = 0$$

- PIGS: parametric-input Gaussian-system models

$$K_w(\theta) = 0 \leftarrow$$ Bayesian FIR models!

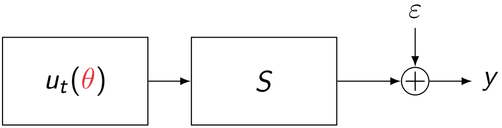
- EIGS: Estimated-input Gaussian-system models

$$w = \hat{w}$$

Example: Semi-blind models

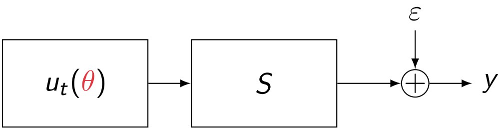
Example: Semi-blind models

- Semi-blind model

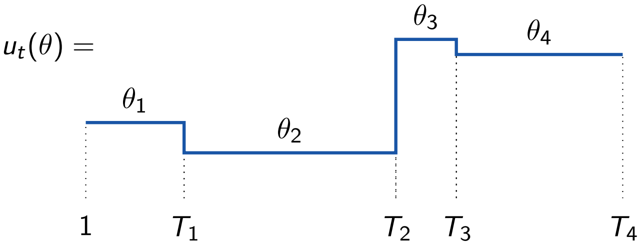


Example: Semi-blind models

- Semi-blind model

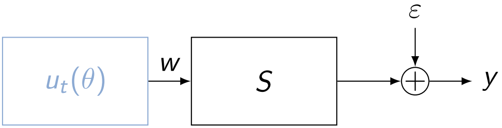


- Piecewise constant input

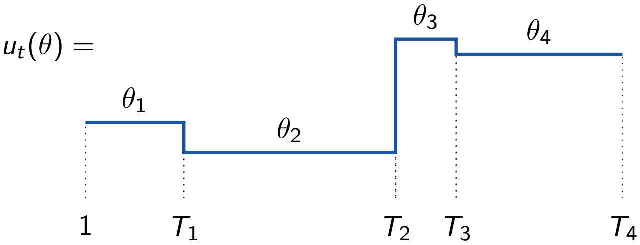


Example: Semi-blind models

- Semi-blind model



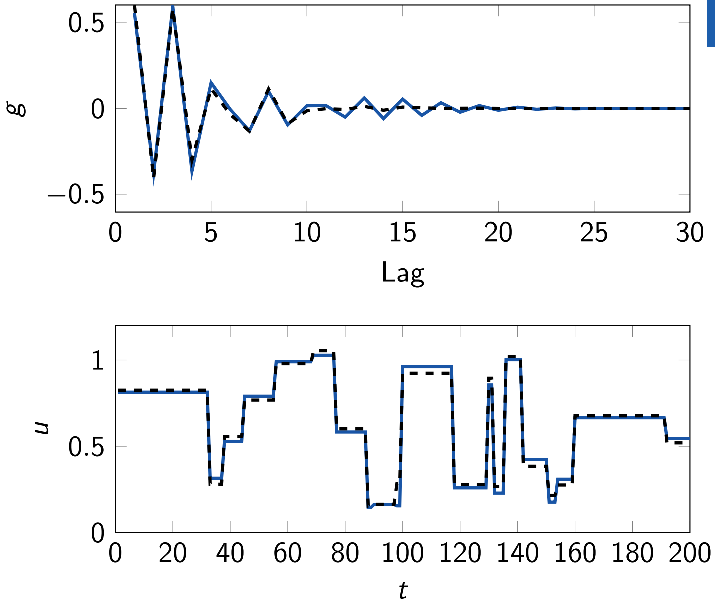
- Piecewise constant input



- PIGS uncertain-input model with

$$w \sim \mathcal{N}(H\theta, 0) \qquad g \sim \mathcal{N}(0, K(\rho))$$

Example: Semi-blind models

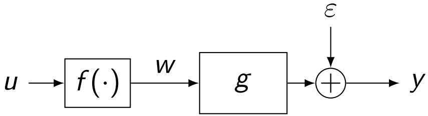


$n_{\mathrm{lti}} = 20,\ N = 200,\ p = 20,\ \mathrm{SNR} = 10,$ —— = true, - - - = estimated

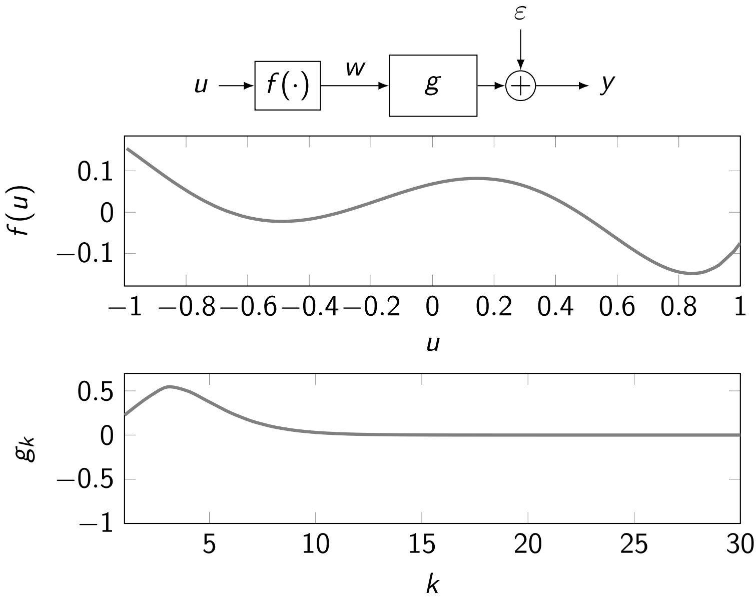
Other examples

- Errors-in-variables
- Cascaded models
- Estimation of initial conditions
- Systems with missing data

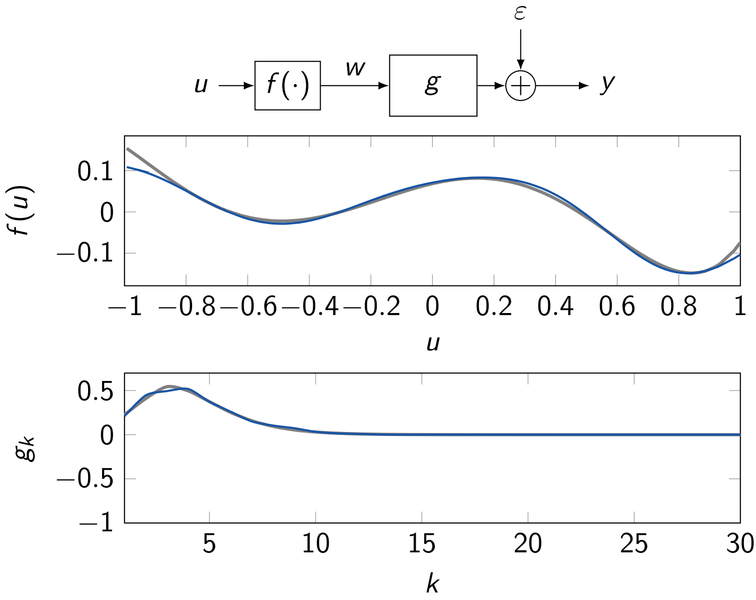
Another Hammerstein example



Another Hammerstein example

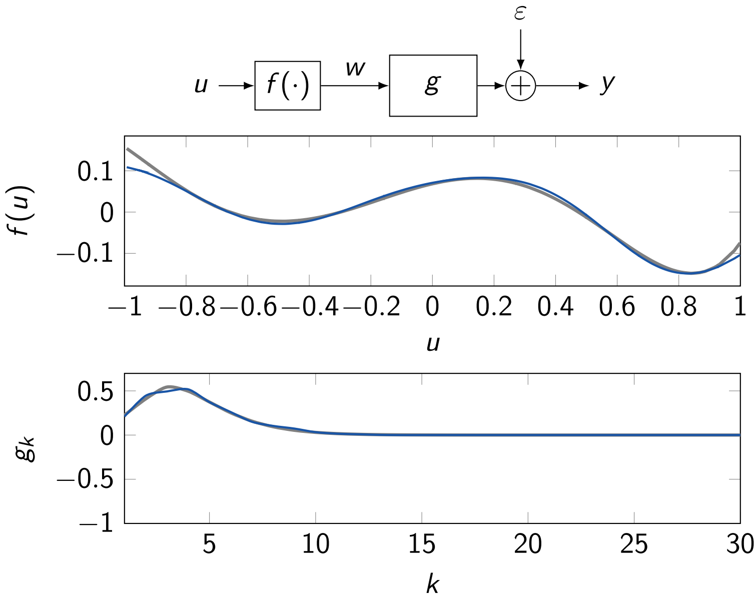


Another Hammerstein example



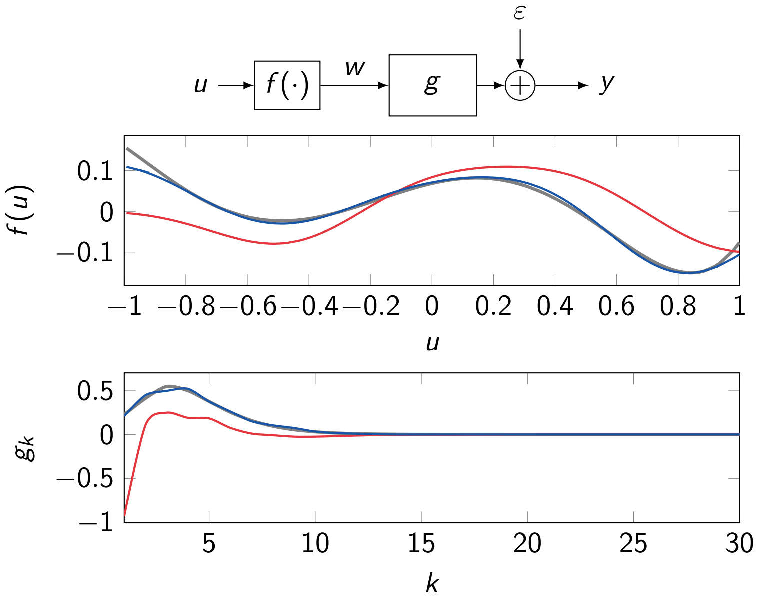
$\varepsilon \sim \mathcal{N}(0, \sigma^2)$

Another Hammerstein example



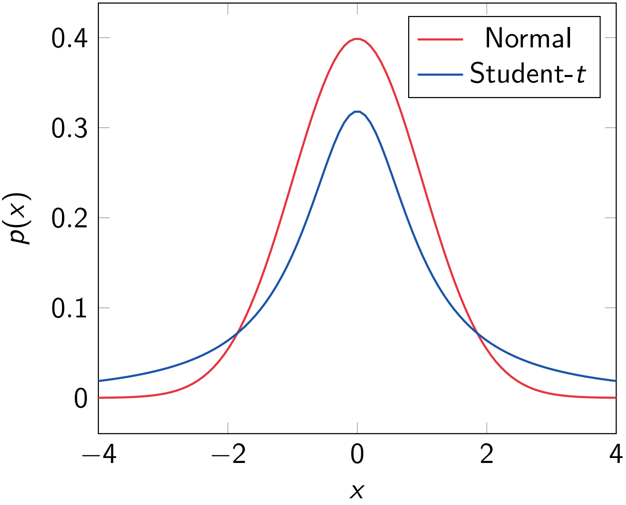
$\varepsilon \sim \mathcal{N}(0, \sigma^2)$ $\varepsilon \sim 0.8\mathcal{N}(0, \sigma^2) + 0.2\mathcal{N}(0, 10\sigma^2)$

Another Hammerstein example



$\varepsilon \sim \mathcal{N}(0, \sigma^2)$ $\varepsilon \sim 0.8\mathcal{N}(0, \sigma^2) + 0.2\mathcal{N}(0, 10\sigma^2)$

We need a noise model with *heavy tails*



Compounded Gaussian noise model

- Student-t model for the noise

$$\varepsilon_t \sim \mathrm{St}(\nu, \eta)$$

Compounded Gaussian noise model

- Student-t model for the noise

$$\varepsilon_t \sim \mathrm{St}(\nu, \eta)$$

- Equivalently, each noise precision is Gamma distributed

$$\lambda_t \sim \mathrm{Ga}(\alpha, \beta) \longleftarrow \text{ Prior density!}$$

Compounded Gaussian noise model

- Student-t model for the noise

$$\varepsilon_t \sim \mathrm{St}(\nu, \eta)$$

- Equivalently, each noise precision is Gamma distributed

$$\begin{aligned} \lambda_t &\sim \mathrm{Ga}(\alpha, \beta) \longleftarrow \text{ Prior density!} \\ \varepsilon_t | \lambda_t &\sim \mathcal{N}\left(0, \lambda_t^{-1}\right) \end{aligned}$$

Compounded Gaussian noise model

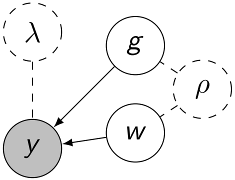
- Student-t model for the noise

$$\varepsilon_t \sim \mathrm{St}(\nu, \eta)$$

- Equivalently, each noise precision is Gamma distributed

$$\lambda_t \sim \mathrm{Ga}(\alpha, \beta) \longleftarrow \text{ Prior density!}$$

$$\varepsilon_t | \lambda_t \sim \mathcal{N}(0, \lambda_t^{-1})$$



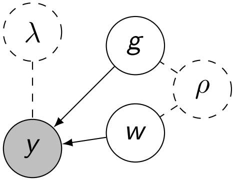
Compounded Gaussian noise model

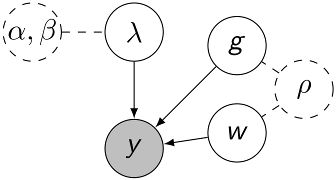
- Student-t model for the noise

$$\varepsilon_t \sim \mathrm{St}(\nu, \eta)$$

- Equivalently, each noise precision is Gamma distributed

$$\lambda_t \sim \mathrm{Ga}(\alpha, \beta) \longleftarrow \text{ Prior density!}$$
$$\varepsilon_t | \lambda_t \sim \mathcal{N}(0, \lambda_t^{-1})$$





Model estimation

- Priors

$$g \sim \mathcal{N}\big(0,\, K_g(\rho)\big), \quad w \sim \mathcal{N}\big(0,\, K_w(\rho)\big), \quad \lambda_t \sim \mathrm{Ga}(\alpha, \beta)$$

$$t = 1, \ldots, N$$

- Data model

$$y|g, w, \lambda_1, \ldots, \lambda_N \sim \mathcal{N}\big(Wg,\, \mathrm{Diag}\{\lambda_t^{-1}\}\big)$$

Model estimation

- Priors

$$g \sim \mathcal{N}\big(0,\, K_g(\rho)\big), \quad w \sim \mathcal{N}\big(0,\, K_w(\rho)\big), \quad \lambda_t \sim \mathrm{Ga}(\alpha, \beta)$$

$$t = 1, \ldots, N$$

- Data model

$$y | g, w, \lambda_1, \ldots, \lambda_N \sim \mathcal{N}\big(Wg,\, \mathrm{Diag}\{\lambda_t^{-1}\}\big)$$

- Same type of model as before only more hyperparameters ($\{\lambda_t\}_{t=1}^N$ instead of σ_y^2)

Model estimation

- Priors

$$g \sim \mathcal{N}\big(0,\, K_g(\rho)\big), \quad w \sim \mathcal{N}\big(0,\, K_w(\rho)\big), \quad \lambda_t \sim \mathrm{Ga}(\alpha, \beta)$$

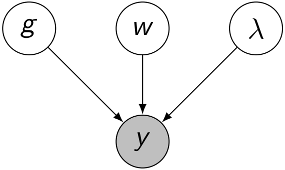
$$t = 1, \dots, N$$

- Data model

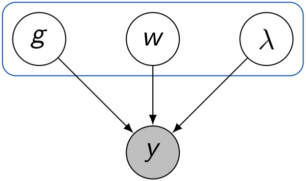
$$y | g, w, \lambda_1, \dots, \lambda_N \sim \mathcal{N}\big(Wg,\, \mathrm{Diag}\{\lambda_t^{-1}\}\big)$$

- Same type of model as before only more hyperparameters ($\{\lambda_t\}_{t=1}^N$ instead of σ_y^2)
- Same tools as before can be used (MCEM, Gibbs sampling)

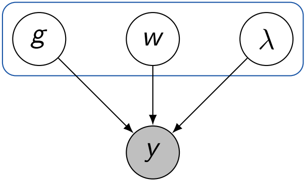
Gibbs sampling



Gibbs sampling

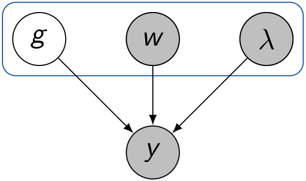


Gibbs sampling



In sequence

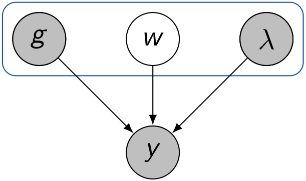
Gibbs sampling



In sequence

1. sample $\bar{g}^{(i)} | \bar{w}^{(i-1)}, \bar{\lambda}^{(i-1)}, y$

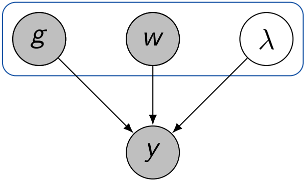
Gibbs sampling



In sequence

1. sample $\bar{g}^{(i)}|\bar{w}^{(i-1)}, \bar{\lambda}^{(i-1)}, y$
2. sample $\bar{w}^{(i)}|\bar{g}^{(i)}, \bar{\lambda}^{(i-1)}, y$

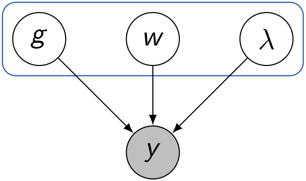
Gibbs sampling



In sequence

1. sample $\bar{g}^{(i)}|\bar{w}^{(i-1)}, \bar{\lambda}^{(i-1)}, y$
2. sample $\bar{w}^{(i)}|\bar{g}^{(i)}, \bar{\lambda}^{(i-1)}, y$
3. sample $\bar{\lambda}^{(i)}|\bar{g}^{(i)}, \bar{w}^{(i)} y$

Gibbs sampling



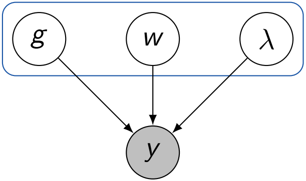
In sequence

1. sample $\bar{g}^{(i)}|\bar{w}^{(i-1)}, \bar{\lambda}^{(i-1)}, y$
2. sample $\bar{w}^{(i)}|\bar{g}^{(i)}, \bar{\lambda}^{(i-1)}, y$
3. sample $\bar{\lambda}^{(i)}|\bar{g}^{(i)}, \bar{w}^{(i)} y$

Result

A Markov chain with $(g, w, \lambda|y)$ as its stationary distribution

Gibbs sampling



In sequence

1. sample $\bar{g}^{(i)}|\bar{w}^{(i-1)}, \bar{\lambda}^{(i-1)}, y$
2. sample $\bar{w}^{(i)}|\bar{g}^{(i)}, \bar{\lambda}^{(i-1)}, y$
3. sample $\bar{\lambda}^{(i)}|\bar{g}^{(i)}, \bar{w}^{(i)} y$

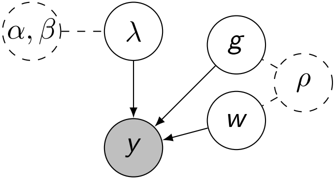
Result

A Markov chain with $(g, w, \lambda|y)$ as its stationary distribution

Known expressions for

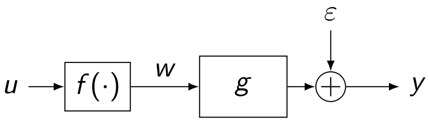
$$g|y, w, \lambda \sim \mathcal{N}(m_g, P_g)$$
$$w|y, g, \lambda \sim \mathcal{N}(m_w, P_w)$$
$$\lambda_t|y, g, w \sim \mathrm{Ga}(\alpha_t, \beta_t)$$

Approximate inference algorithm for Robust UI models

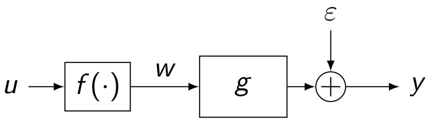



```
procedure ESTIMATE-ROBUST(data)
    Initialize ρ̂, α̂, β̂
    while not converged do                                    ▷ EM
        Approximate Q(ρ, α, β)                                ▷ Gibbs sampling
        ρ̂, α̂, β̂ ← arg max_{ρ,α,β} Q(ρ, α, β)                ▷ Scalar optimization
    end while
    ĝ, ŵ, λ̂ ← E {[} g, w, λ|y; ρ̂, α̂, β]                    ▷ Gibbs sampling
    return ĝ, ŵ, λ̂
end procedure
```

Simulation study: Hammerstein systems

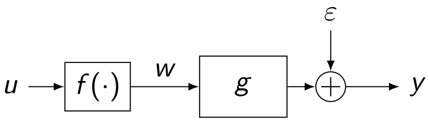


Simulation study: Hammerstein systems



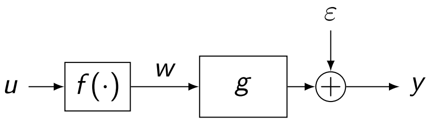
- Polynomial nonlinearity of order $p \in 5, \ldots, 10$

Simulation study: Hammerstein systems



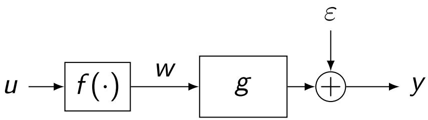
- Polynomial nonlinearity of order $p \in 5, \ldots, 10$
- Transfer function of order $m \in 3, 4, 5$

Simulation study: Hammerstein systems



- Polynomial nonlinearity of order $p \in 5, \ldots, 10$
- Transfer function of order $m \in 3, 4, 5$
- $N = 300$ samples of output, uniform white input in $-1, 1$

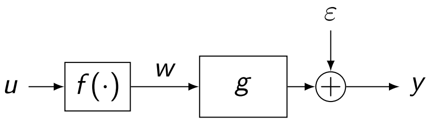
Simulation study: Hammerstein systems



- Polynomial nonlinearity of order $p \in 5, \ldots, 10$
- Transfer function of order $m \in 3, 4, 5$
- $N = 300$ samples of output, uniform white input in $-1, 1$

Two methods

Simulation study: Hammerstein systems

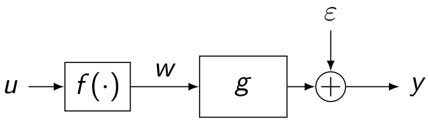


- Polynomial nonlinearity of order $p \in 5, \ldots, 10$
- Transfer function of order $m \in 3, 4, 5$
- $N = 300$ samples of output, uniform white input in $-1, 1$

Two methods

H-Gaussian with Gaussian noise model

Simulation study: Hammerstein systems



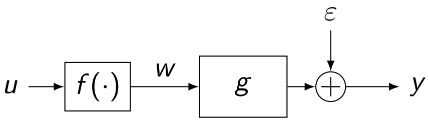
- Polynomial nonlinearity of order $p \in 5, \ldots, 10$
- Transfer function of order $m \in 3, 4, 5$
- $N = 300$ samples of output, uniform white input in $-1, 1$

Two methods

H-Gaussian with Gaussian noise model

H-Robust with Student-t noise model

Simulation study: Hammerstein systems



- Polynomial nonlinearity of order $p \in 5, \ldots, 10$
- Transfer function of order $m \in 3, 4, 5$
- $N = 300$ samples of output, uniform white input in $-1, 1$

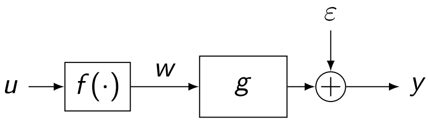
Two methods

H-Gaussian with Gaussian noise model

H-Robust with Student-t noise model

Two experiments

Simulation study: Hammerstein systems



- Polynomial nonlinearity of order $p \in 5, \ldots, 10$
- Transfer function of order $m \in 3, 4, 5$
- $N = 300$ samples of output, uniform white input in $-1, 1$

Two methods

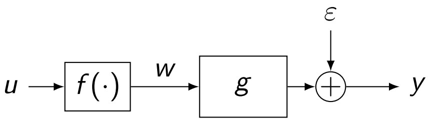
H-Gaussian with Gaussian noise model

H-Robust with Student-t noise model

Two experiments

- Varying number of outliers (with fixed variance $10\sigma^2$)

Simulation study: Hammerstein systems



- Polynomial nonlinearity of order $p \in 5, \ldots, 10$
- Transfer function of order $m \in 3, 4, 5$
- $N = 300$ samples of output, uniform white input in $-1, 1$

Two methods

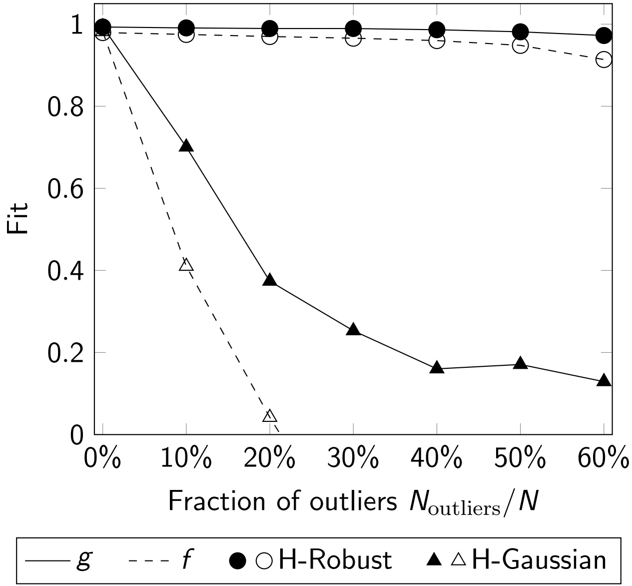
H-Gaussian with Gaussian noise model

H-Robust with Student-t noise model

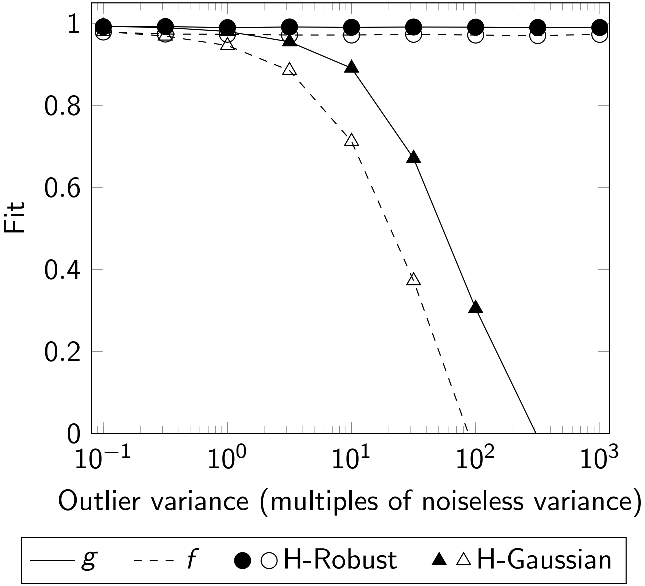
Two experiments

- Varying number of outliers (with fixed variance $10\sigma^2$)
- Varying outlier variance (with fixed fraction 15%)

Outlier fraction results



Outlier variance results



Conclusions

Conclusions

- Many classical problems are uncertain-inputs problems

Conclusions

- Many classical problems are uncertain-inputs problems
- Proposed a model for uncertain-input systems

Conclusions

- Many classical problems are uncertain-inputs problems
- Proposed a model for uncertain-input systems
- Estimated the parameters
  - Gibbs sampling
  - MCEM
  - Monte Carlo integration

Conclusions

- Many classical problems are uncertain-inputs problems
- Proposed a model for uncertain-input systems
- Estimated the parameters
  - Gibbs sampling
  - MCEM
  - Monte Carlo integration
- Examples of classical problems
  - Hammerstein model
  - Semi-blind model

Conclusions

- Many classical problems are uncertain-inputs problems
- Proposed a model for uncertain-input systems
- Estimated the parameters
  - Gibbs sampling
  - MCEM
  - Monte Carlo integration
- Examples of classical problems
  - Hammerstein model
  - Semi-blind model
- Proposed a robust extension with Student-t likelihood