Overcoming Model Misspecification in Bayesian Inference of Molecular Signalling Networks





Yinghan Li[1], Heishiro Kanagawa[2], Francesca Romana Crucinio[3], Matthew A. Fisher[1], **Chris. J. Oates**[1]

[1] Newcastle University, UK
[2] Fujitsu Research of Europe, UK
[3] University of Turin, Italy

One World Approximate Bayesian Inference, October 2026

Recap: An aerial view of Post-Bayesian methods

Set-Up and Notation

For our purposes, a **statistical model** is a probability density function

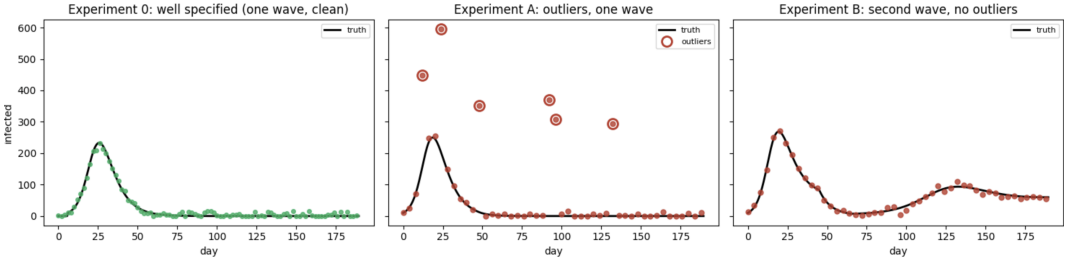
$$p_\theta(\boldsymbol{y}|\boldsymbol{x})$$

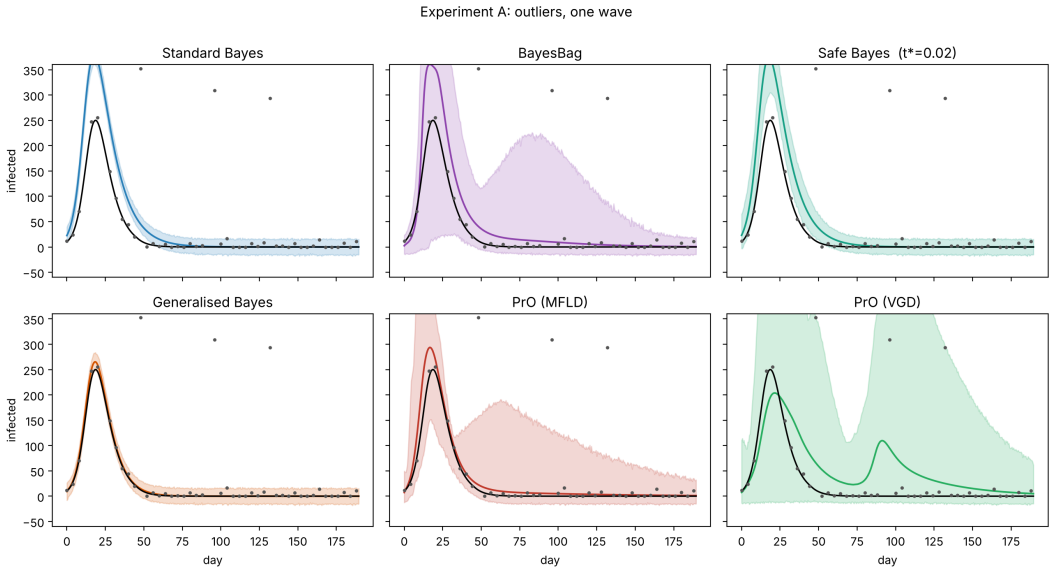
where

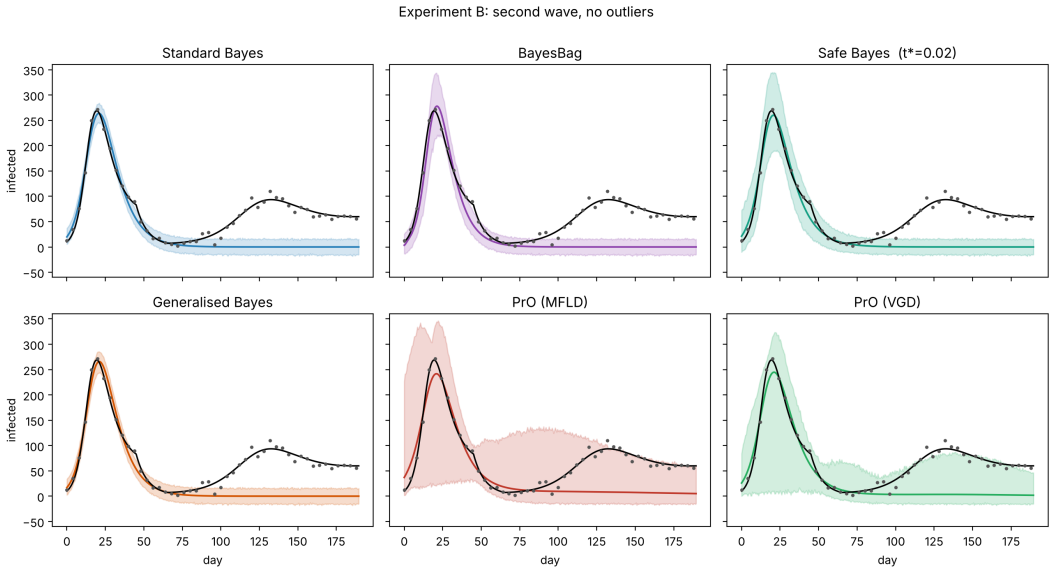
- $\boldsymbol{y}$ are the data $\boldsymbol{y} \in \mathbb{R}^d$
- $\boldsymbol{x}$ are relevant covariates $\boldsymbol{x} \in \mathcal{X}$
- $\boldsymbol{\theta}$ are **parameters** of the model $\boldsymbol{\theta} \in \mathbb{R}^p$

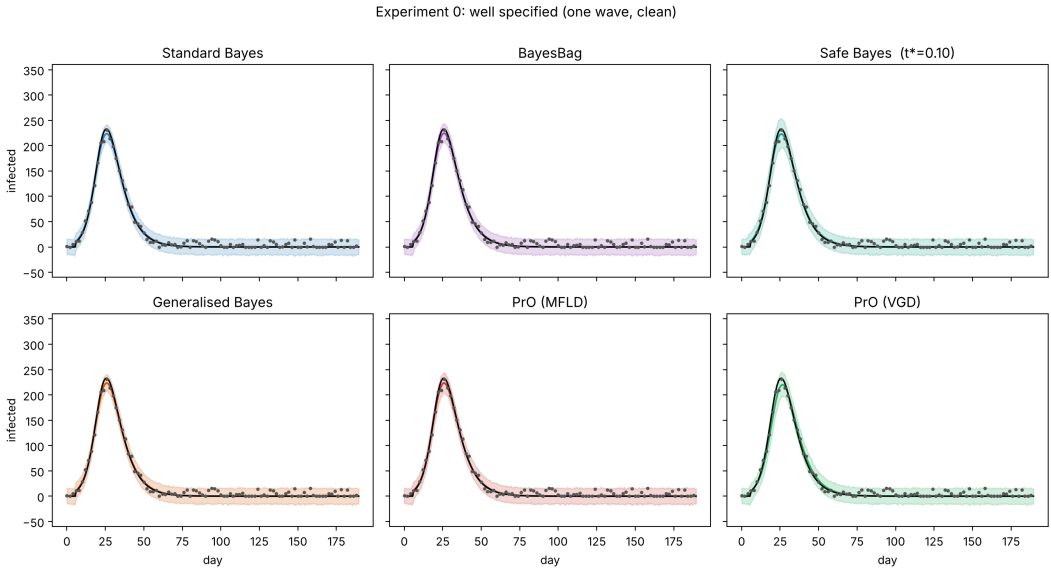
An aerial view of Post-Bayesian methods

Let's explore the effect of model misspecification, using a Covid model as a test bed:




Experiment A: outliers, one wave
Standard Bayes
BayesBag
Safe Bayes (t*=0.02)
Generalised Bayes
PrO (MFLD)
PrO (VGD)
infected
day
−50
0
50
100
150
200
250
300
350
0
25
50
75
100
125
150
175


Experiment B: second wave, no outliers
Standard Bayes
BayesBag
Safe Bayes (t*=0.02)
Generalised Bayes
PrO (MFLD)
PrO (VGD)
infected
day
−50
0
50
100
150
200
250
300
350
0
25
50
75
100
125
150
175


Experiment 0: well specified (one wave, clean)
Standard Bayes
BayesBag
Safe Bayes (t*=0.10)
Generalised Bayes
PrO (MFLD)
PrO (VGD)
infected
day
350
300
250
200
150
100
50
0
−50
0
25
50
75
100
125
150
175

Predictively-Oriented Posteriors

Given a statistical model $p_\theta(\boldsymbol{y}|\boldsymbol{x})$ that is useful but misspecified.

Step 1: Mitigate Misspecification by forming a mixture model

$$p_Q(\boldsymbol{y}|\boldsymbol{x}) = \int p_\theta(\boldsymbol{y}|\boldsymbol{x}) \, \mathrm{d}Q(\theta)$$
$$= \int p_\theta(\boldsymbol{y}|\boldsymbol{x}) q(\theta) \, \mathrm{d}\theta \text{ if } Q \text{ has pdf } q$$
$$= \sum_i w_i p_{\theta_i}(\boldsymbol{y}|\boldsymbol{x}) \text{ if } Q = \sum_i w_i \delta_{\theta_i} \text{ has discrete support}$$

Step 2: Learn Q by calibrating the predictive distribution p_Q to the dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$.

Predictively-Oriented Posteriors

Given a statistical model $p_\theta(\boldsymbol{y}|\boldsymbol{x})$ that is useful but misspecified.

Step 1: Mitigate Misspecification by forming a mixture model

$$p_Q(\boldsymbol{y}|\boldsymbol{x}) = \int p_\theta(\boldsymbol{y}|\boldsymbol{x}) \, \mathrm{d}Q(\boldsymbol{\theta})$$

$$= \int p_\theta(\boldsymbol{y}|\boldsymbol{x}) q(\theta) \, \mathrm{d}\theta \text{ if } Q \text{ has pdf } q$$

$$= \sum_i w_i p_{\theta_i}(\boldsymbol{y}|\boldsymbol{x}) \text{ if } Q = \sum_i w_i \delta_{\theta_i} \text{ has discrete support}$$

Step 2: Learn Q by calibrating the predictive distribution p_Q to the dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$.

Predictively-Oriented Posteriors

Given a statistical model $p_\theta(\boldsymbol{y}|\boldsymbol{x})$ that is useful but misspecified.

Step 1: Mitigate Misspecification by forming a mixture model

$$p_Q(\boldsymbol{y}|\boldsymbol{x}) = \int p_\theta(\boldsymbol{y}|\boldsymbol{x}) \, \mathrm{d}Q(\boldsymbol{\theta})$$
$$= \int p_\theta(\boldsymbol{y}|\boldsymbol{x}) q(\boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta} \text{ if } Q \text{ has pdf } q$$
$$= \sum_i w_i p_{\theta_i}(\boldsymbol{y}|\boldsymbol{x}) \text{ if } Q = \sum_i w_i \delta_{\theta_i} \text{ has discrete support}$$

Step 2: Learn Q by calibrating the predictive distribution p_Q to the dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$.

Predictively-Oriented Posteriors

Given a statistical model $p_{\theta}(\boldsymbol{y}|\boldsymbol{x})$ that is useful but misspecified.

Step 1: Mitigate Misspecification by forming a mixture model

$$p_Q(\boldsymbol{y}|\boldsymbol{x}) = \int p_{\theta}(\boldsymbol{y}|\boldsymbol{x}) \, \mathrm{d}Q(\boldsymbol{\theta})$$
$$= \int p_{\theta}(\boldsymbol{y}|\boldsymbol{x}) q(\boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta} \text{ if } Q \text{ has pdf } q$$
$$= \sum_i w_i p_{\theta_i}(\boldsymbol{y}|\boldsymbol{x}) \text{ if } Q = \sum_i w_i \delta_{\theta_i} \text{ has discrete support}$$

Step 2: Learn Q by calibrating the predictive distribution p_Q to the dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$.

Predictively-Oriented Posteriors

Given a statistical model $p_\theta(\boldsymbol{y}|\boldsymbol{x})$ that is useful but misspecified.

Step 1: Mitigate Misspecification by forming a mixture model

$$p_Q(\boldsymbol{y}|\boldsymbol{x}) = \int p_\theta(\boldsymbol{y}|\boldsymbol{x}) \, \mathrm{d}Q(\boldsymbol{\theta})$$
$$= \int p_\theta(\boldsymbol{y}|\boldsymbol{x}) q(\boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta} \text{ if } Q \text{ has pdf } q$$
$$= \sum_i w_i p_{\theta_i}(\boldsymbol{y}|\boldsymbol{x}) \text{ if } Q = \sum_i w_i \delta_{\theta_i} \text{ has discrete support}$$

Step 2: Learn Q by calibrating the predictive distribution p_Q to the dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$.

Predictively-Oriented Posteriors

Pick a scoring rule; e.g. the log scoring rule $S(p, \boldsymbol{y}) = -\log p(\boldsymbol{y})$.

Introduced in Shen et al. [2026] and analysed in McLatchie et al. [2025]:

$$Q_{\mathrm{PrO}} = \arg\min_{Q} \underbrace{\sum_{i=1}^{n} S\left(p_Q(\cdot|\boldsymbol{x}_i), \boldsymbol{y}_i\right)}_{\text{predictive fit}} + \underbrace{\lambda_n \mathrm{KL}(Q||Q_0)}_{\text{regularisation}}$$

Interpretation: This is a regularised formulation of **nonparametric maximum likelihood**.

Regularisation resolves computational difficulties and non-identifiability [see e.g. Laird, 1978], else the minimising measure will generally be fully atomic, see Lindsay [1995, e.g. Theorem 21 in Chapter 5] and Jordan-Squire [2015].

Predictively-Oriented Posteriors

Pick a scoring rule; e.g. the log scoring rule $S(p, \boldsymbol{y}) = -\log p(\boldsymbol{y})$.

Introduced in Shen et al. [2026] and analysed in McLatchie et al. [2025]:

$$Q_{\mathrm{PrO}} = \arg\min_{Q} \underbrace{\sum_{i=1}^{n} S\left(p_Q(\cdot|\boldsymbol{x}_i), \boldsymbol{y}_i\right)}_{\text{predictive fit}} + \underbrace{\lambda_n \,\mathsf{KL}(Q||Q_0)}_{\text{regularisation}}$$

Interpretation: This is a regularised formulation of **nonparametric maximum likelihood**.

Regularisation resolves computational difficulties and non-identifiability [see e.g. Laird, 1978], else the minimising measure will generally be fully atomic, see Lindsay [1995, e.g. Theorem 21 in Chapter 5] and Jordan-Squire [2015].

Predictively-Oriented Posteriors

Pick a scoring rule; e.g. the log scoring rule $S(p, \boldsymbol{y}) = -\log p(\boldsymbol{y})$.

Introduced in Shen et al. [2026] and analysed in McLatchie et al. [2025]:

$$Q_{\mathrm{PrO}} = \arg\min_{Q} \underbrace{\sum_{i=1}^{n} S\left(p_Q(\cdot|\boldsymbol{x}_i), \boldsymbol{y}_i\right)}_{\text{predictive fit}} + \underbrace{\lambda_n \, \mathrm{KL}(Q||Q_0)}_{\text{regularisation}}$$

Regularisation Target
Q_0 acts on Q_{PrO} in essentially the same way that Q_0 acts on generalised posteriors; as a reference measure in a Radon–Nikodym derivative [Bissiri et al., 2016, Knoblauch et al., 2022].

That is, one can speak about **'updating'** PrO posteriors.

Predictively-Oriented Posteriors

Pick a scoring rule; e.g. the log scoring rule $S(p, \boldsymbol{y}) = -\log p(\boldsymbol{y})$.

Introduced in Shen et al. [2026] and analysed in McLatchie et al. [2025]:

$$Q_{\mathrm{PrO}} = \arg\min_{Q} \underbrace{\sum_{i=1}^{n} S\left(p_Q(\cdot|\boldsymbol{x}_i), \boldsymbol{y}_i\right)}_{\text{predictive fit}} + \underbrace{\lambda_n \mathrm{KL}(Q||Q_0)}_{\text{regularisation}}$$

Learning Rate
PrO posteriors are **relatively insensitive to the learning rate λ_n**. i.e. support of Q_{PrO} is not a singleton set when $\lambda_n \to 0$.

For the log scoring rule we can take $\lambda_n = 1$ since both terms are on the same scale (nats).

PrO Posteriors Give Better Predictions

Compare the posterior predictives:

$$P_{\text{Bayes}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\, \mathrm{d}Q_{\text{Bayes}}(\boldsymbol{\theta}) \qquad \text{vs} \qquad P_{\text{PrO}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\, \mathrm{d}Q_{\text{PrO}}(\boldsymbol{\theta})$$

Quadratic regression model: $y_i = \theta x_i^2 + \epsilon_i$, $\epsilon_i \sim N(0, 1)$

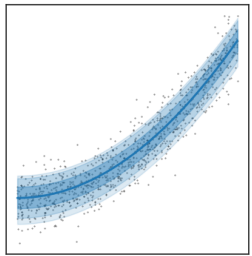
P_{Bayes} P_{PrO}

PrO Posteriors Give Better Predictions

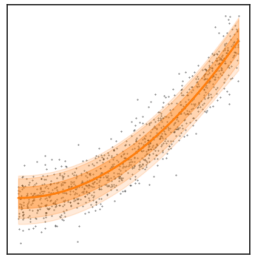
Compare the posterior predictives:

$$P_{\text{Bayes}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\, \mathrm{d}Q_{\text{Bayes}}(\boldsymbol{\theta}) \qquad \text{vs} \qquad P_{\text{PrO}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\, \mathrm{d}Q_{\text{PrO}}(\boldsymbol{\theta})$$

Quadratic regression model: $y_i = \theta x_i^2 + \epsilon_i$, $\epsilon_i \sim N(0, 1)$



P_{Bayes}



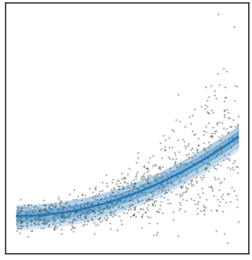
P_{PrO}

PrO Posteriors Give Better Predictions

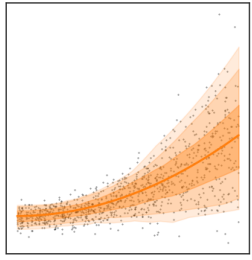
Compare the posterior predictives:

$$P_{\text{Bayes}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\, \mathrm{d}Q_{\text{Bayes}}(\boldsymbol{\theta}) \qquad \text{vs} \qquad P_{\text{PrO}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\, \mathrm{d}Q_{\text{PrO}}(\boldsymbol{\theta})$$

Quadratic regression model: $y_i = \theta x_i^2 + \epsilon_i$, $\epsilon_i \sim N(0, 1)$



P_{Bayes}



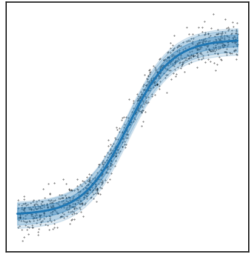
P_{PrO}

PrO Posteriors Give Better Predictions

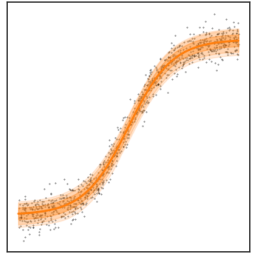
Compare the posterior predictives:

$$P_{\mathrm{Bayes}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\,\mathrm{d}Q_{\mathrm{Bayes}}(\boldsymbol{\theta}) \qquad \text{vs} \qquad P_{\mathrm{PrO}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\,\mathrm{d}Q_{\mathrm{PrO}}(\boldsymbol{\theta})$$

Sigmoid regression model: $y_i = \mathrm{sigmoid}(\theta x_i) + \epsilon_i$, $\epsilon_i \sim N(0,1)$



P_{Bayes}



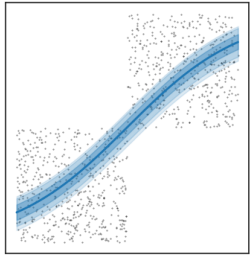
P_{PrO}

PrO Posteriors Give Better Predictions

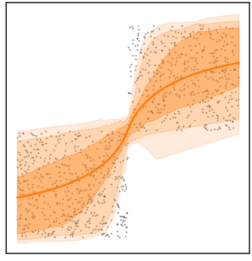
Compare the posterior predictives:

$$P_{\text{Bayes}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\,\mathrm{d}Q_{\text{Bayes}}(\boldsymbol{\theta}) \qquad \text{vs} \qquad P_{\text{PrO}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\,\mathrm{d}Q_{\text{PrO}}(\boldsymbol{\theta})$$

Sigmoid regression model: $y_i = \text{sigmoid}(\theta x_i) + \epsilon_i$, $\epsilon_i \sim N(0, 1)$



P_{Bayes}



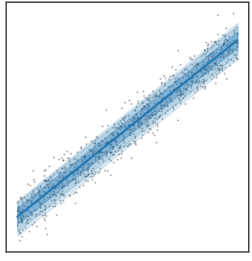
P_{PrO}

PrO Posteriors Give Better Predictions

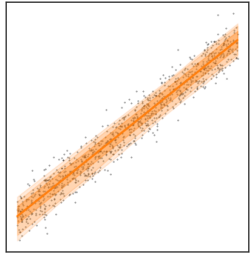
Compare the posterior predictives:

$$P_{\mathrm{Bayes}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\,\mathrm{d}Q_{\mathrm{Bayes}}(\boldsymbol{\theta}) \qquad \text{vs} \qquad P_{\mathrm{PrO}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\,\mathrm{d}Q_{\mathrm{PrO}}(\boldsymbol{\theta})$$

Linear regression model: $y_i = \theta_1 + \theta_2 x_i + \epsilon_i$, $\epsilon_i \sim N(0,1)$



P_{Bayes}



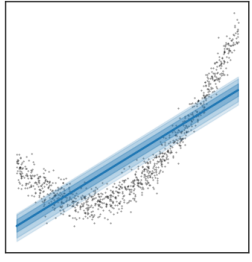
P_{PrO}

PrO Posteriors Give Better Predictions

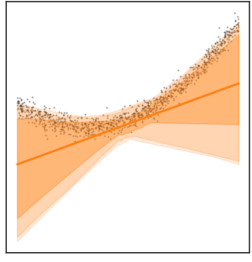
Compare the posterior predictives:

$$P_{\text{Bayes}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\, \mathrm{d}Q_{\text{Bayes}}(\boldsymbol{\theta}) \qquad \text{vs} \qquad P_{\text{PrO}}(\cdot|\boldsymbol{x}^*) = \int p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}^*)\, \mathrm{d}Q_{\text{PrO}}(\boldsymbol{\theta})$$

Linear regression model: $y_i = \theta_1 + \theta_2 x_i + \epsilon_i$, $\epsilon_i \sim N(0, 1)$



P_{Bayes}



P_{PrO}

PrO Posteriors as an *Inference* Tool

For $(\boldsymbol{x}_i, \boldsymbol{y}_i) \sim \mathbb{P}$, the PrO posterior based on $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ contracts as $n \to \infty$ to

$$\arg\min_Q \ \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathbb{P}}[S(p_Q(\cdot|\boldsymbol{x}), \boldsymbol{y})]$$

e.g. for the IID case see McLatchie et al. [2025].

Thus, *if the statistical model is well-specified and identifiable*, then

$$\delta_{\boldsymbol{\theta}^\dagger} = \arg\min_Q \ \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathbb{P}}[S(p_Q(\cdot|\boldsymbol{x}), \boldsymbol{y})]$$

is a *Dirac distribution* (i.e. a point mass) and

$$Q_{\text{PrO}} \to \delta_{\boldsymbol{\theta}^\dagger}$$

contracts to the true parameter $\boldsymbol{\theta}^\dagger$.

PrO Posteriors as an *Inference* Tool

For $(\boldsymbol{x}_i, \boldsymbol{y}_i) \sim \mathbb{P}$, the PrO posterior based on $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ contracts as $n \to \infty$ to

$$\arg\min_Q \ \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathbb{P}}[S(p_Q(\cdot|\boldsymbol{x}), \boldsymbol{y})]$$

e.g. for the IID case see McLatchie et al. [2025].

Thus, *if the statistical model is well-specified and identifiable*, then

$$\delta_{\boldsymbol{\theta}^\dagger} = \arg\min_Q \ \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathbb{P}}[S(p_Q(\cdot|\boldsymbol{x}), \boldsymbol{y})]$$

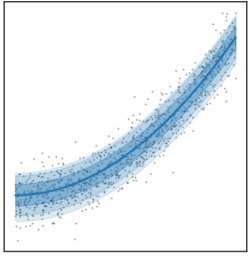
is a *Dirac distribution* (i.e. a point mass) and

$$Q_{\text{PrO}} \to \delta_{\boldsymbol{\theta}^\dagger}$$

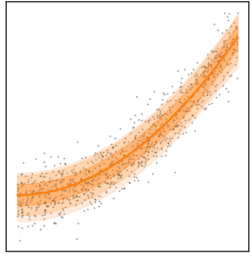
contracts to the true parameter $\boldsymbol{\theta}^\dagger$.

PrO Posteriors as an *Inference* Tool

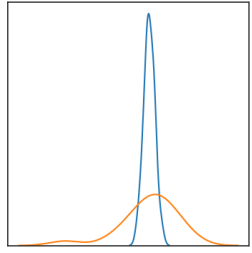
Quadratic regression model: $y_i = \theta x_i^2 + \epsilon_i,\ \epsilon_i \sim N(0, 1)$



P_{Bayes}



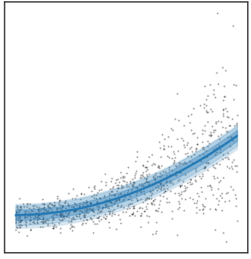
P_{PrO}



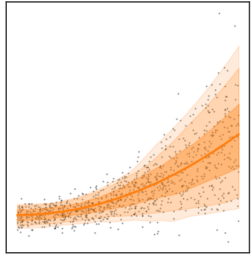
Q_{Bayes} vs Q_{PrO}

PrO Posteriors as an *Inference* Tool

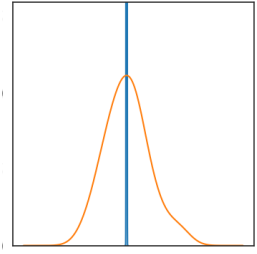
Quadratic regression model: $y_i = \theta x_i^2 + \epsilon_i,\ \epsilon_i \sim N(0, 1)$



P_{Bayes}



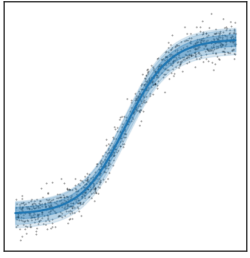
P_{PrO}



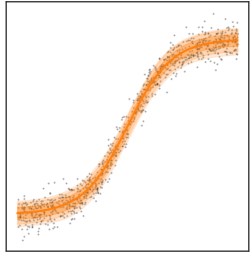
Q_{Bayes} vs Q_{PrO}

PrO Posteriors as an *Inference* Tool

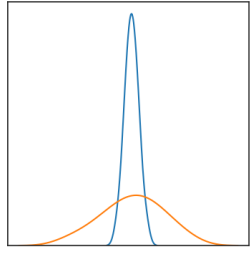
Sigmoid regression model: $y_i = \text{sigmoid}(\theta x_i) + \epsilon_i$, $\epsilon_i \sim N(0, 1)$



P_{Bayes}



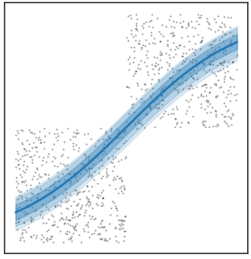
P_{PrO}



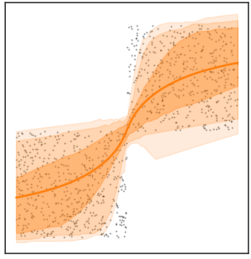
Q_{Bayes} vs Q_{PrO}

PrO Posteriors as an *Inference* Tool

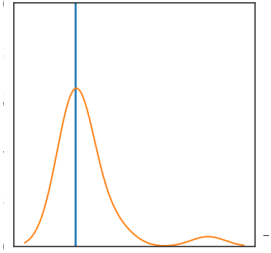
Sigmoid regression model: $y_i = \text{sigmoid}(\theta x_i) + \epsilon_i$, $\epsilon_i \sim N(0, 1)$



P_{Bayes}



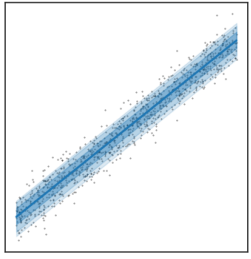
P_{PrO}



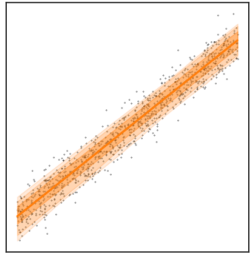
Q_{Bayes} vs Q_{PrO}

PrO Posteriors as an *Inference* Tool

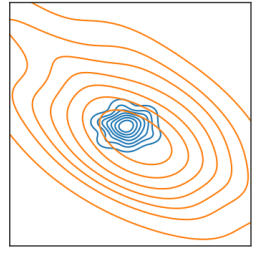
Linear regression model: $y_i = \theta_1 + \theta_2 x_i + \epsilon_i$, $\epsilon_i \sim N(0, 1)$



P_{Bayes}



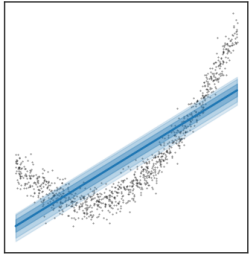
P_{PrO}



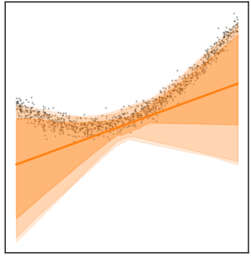
Q_{Bayes} vs Q_{PrO}

PrO Posteriors as an *Inference* Tool

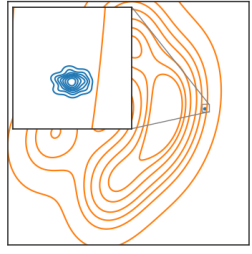
Linear regression model: $y_i = \theta_1 + \theta_2 x_i + \epsilon_i$, $\epsilon_i \sim N(0, 1)$



P_{Bayes}



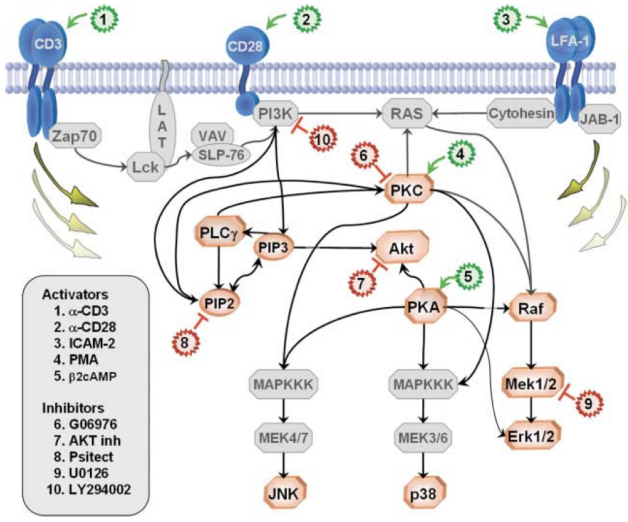
P_{PrO}



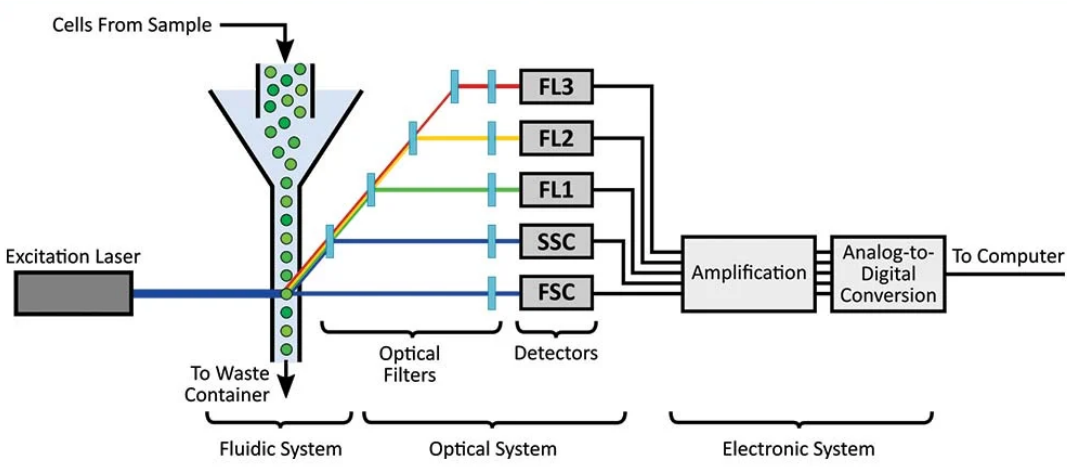
Q_{Bayes} vs Q_{PrO}

Aim: Inference of molecular signalling networks

Example: Protein Kinase Signalling Network



Example: Protein Kinase Signalling Network

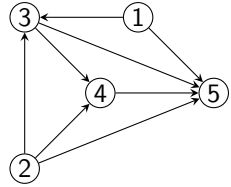


Tool: Bayesian Inference for DAGs

Given a directed acyclic graph G,

$$p_\theta(\boldsymbol{y}|G) = \prod_{i=1}^{d} p_\theta(\boldsymbol{y}_i|\boldsymbol{y}_{\pi_G(i)})$$

where $\pi_G(i)$ denotes the parents of node i in G.

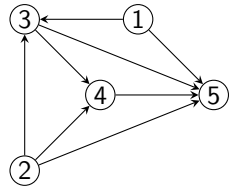


- the number of DAGs on d vertices is $\geq 2^{\binom{d}{2}}$ (**very large!**)
- this motivates:
  - restricting the set $\mathcal{G}$ of candidate DAGs
  - a complexity prior $p(G) \propto \rho^{|E(G)|}$ where $E(G)$ is the set of edges in G
  - a closed-form marginal likelihood $p(\boldsymbol{y}|G)$
- possible with **linear-Gaussian** model and conjugate normal-inverse-Wishart $p(\theta|G)$
- Note: linear-Gaussian model introduces a non-identifiability (called **Markov equivalence**)
- Note: a rich literature on learning DAGs, but going to sidestep it today.

Tool: Bayesian Inference for DAGs

Given a directed acyclic graph G,

$$p_\theta(\boldsymbol{y}|G) = \prod_{i=1}^{d} p_\theta(\boldsymbol{y}_i|\boldsymbol{y}_{\pi_G(i)})$$



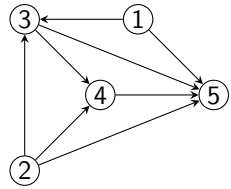
where $\pi_G(i)$ denotes the parents of node i in G.

- the number of DAGs on d vertices is $\geq 2^{\binom{d}{2}}$ (**very large!**)
- this motivates:
  - restricting the set $\mathcal{G}$ of candidate DAGs
  - a complexity prior $p(G) \propto \rho^{|E(G)|}$ where $E(G)$ is the set of edges in G
  - a closed-form marginal likelihood $p(\boldsymbol{y}|G)$
- possible with **linear-Gaussian** model and conjugate normal-inverse-Wishart $p(\theta|G)$
- Note: linear-Gaussian model introduces a non-identifiability (called **Markov equivalence**)
- Note: a rich literature on learning DAGs, but going to sidestep it today.

Tool: Bayesian Inference for DAGs

Given a directed acyclic graph G,

$$p_\theta(\boldsymbol{y}|G) = \prod_{i=1}^{d} p_\theta(\boldsymbol{y}_i|\boldsymbol{y}_{\pi_G(i)})$$



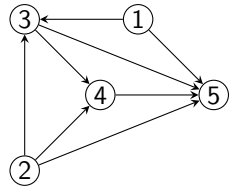
where $\pi_G(i)$ denotes the parents of node i in G.

- the number of DAGs on d vertices is $\geq 2^{\binom{d}{2}}$ (**very large!**)
- this motivates:
  - restricting the set $\mathcal{G}$ of candidate DAGs
  - a complexity prior $p(G) \propto \rho^{|E(G)|}$ where $E(G)$ is the set of edges in G
  - a closed-form marginal likelihood $p(\boldsymbol{y}|G)$
- possible with **linear-Gaussian** model and conjugate normal-inverse-Wishart $p(\theta|G)$
- Note: linear-Gaussian model introduces a non-identifiability (called **Markov equivalence**)
- Note: a rich literature on learning DAGs, but going to sidestep it today.

Tool: Bayesian Inference for DAGs

Given a directed acyclic graph G,

$$p_\theta(\boldsymbol{y}|G) = \prod_{i=1}^{d} p_\theta(\boldsymbol{y}_i|\boldsymbol{y}_{\pi_G(i)})$$



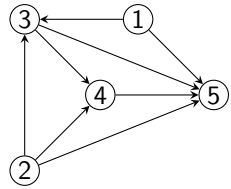
where $\pi_G(i)$ denotes the parents of node i in G.

- the number of DAGs on d vertices is $\geq 2^{\binom{d}{2}}$ (**very large!**)
- this motivates:
  - restricting the set $\mathcal{G}$ of candidate DAGs
  - a complexity prior $p(G) \propto \rho^{|E(G)|}$ where $E(G)$ is the set of edges in G
  - a closed-form marginal likelihood $p(\boldsymbol{y}|G)$
- possible with **linear-Gaussian** model and conjugate normal-inverse-Wishart $p(\theta|G)$
- Note: linear-Gaussian model introduces a non-identifiability (called **Markov equivalence**)
- Note: a rich literature on learning DAGs, but going to sidestep it today.

Tool: Bayesian Inference for DAGs

Given a directed acyclic graph G,

$$p_\theta(\boldsymbol{y}|G) = \prod_{i=1}^{d} p_\theta(\boldsymbol{y}_i|\boldsymbol{y}_{\pi_G(i)})$$



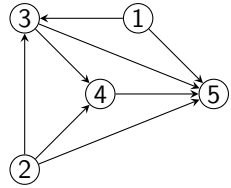
where $\pi_G(i)$ denotes the parents of node i in G.

- the number of DAGs on d vertices is $\geq 2^{\binom{d}{2}}$ (**very large!**)
- this motivates:
  - restricting the set $\mathcal{G}$ of candidate DAGs
  - a complexity prior $p(G) \propto \rho^{|E(G)|}$ where $E(G)$ is the set of edges in G
  - a closed-form marginal likelihood $p(\boldsymbol{y}|G)$
- possible with **linear-Gaussian** model and conjugate normal-inverse-Wishart $p(\theta|G)$
- Note: linear-Gaussian model introduces a non-identifiability (called **Markov equivalence**)
- Note: a rich literature on learning DAGs, but going to sidestep it today.

Tool: Bayesian Inference for DAGs

Given a directed acyclic graph G,

$$p_\theta(\boldsymbol{y}|G) = \prod_{i=1}^{d} p_\theta(\boldsymbol{y}_i|\boldsymbol{y}_{\pi_G(i)})$$



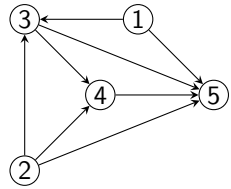
where $\pi_G(i)$ denotes the parents of node i in G.

- the number of DAGs on d vertices is $\geq 2^{\binom{d}{2}}$ (**very large!**)
- this motivates:
  - restricting the set $\mathcal{G}$ of candidate DAGs
  - a complexity prior $p(G) \propto \rho^{|E(G)|}$ where $E(G)$ is the set of edges in G
  - a closed-form marginal likelihood $p(\boldsymbol{y}|G)$
- possible with **linear-Gaussian** model and conjugate normal-inverse-Wishart $p(\theta|G)$
- Note: linear-Gaussian model introduces a non-identifiability (called **Markov equivalence**)
- Note: a rich literature on learning DAGs, but going to sidestep it today.

Tool: Bayesian Inference for DAGs

Given a directed acyclic graph G,

$$p_{\boldsymbol{\theta}}(\boldsymbol{y}|G) = \prod_{i=1}^{d} p_{\boldsymbol{\theta}}(\boldsymbol{y}_i|\boldsymbol{y}_{\pi_G(i)})$$



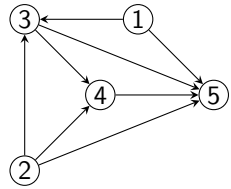
where $\pi_G(i)$ denotes the parents of node i in G.

- the number of DAGs on d vertices is $\geq 2^{\binom{d}{2}}$ (**very large!**)
- this motivates:
  - restricting the set $\mathcal{G}$ of candidate DAGs
  - a complexity prior $p(G) \propto \rho^{|E(G)|}$ where $E(G)$ is the set of edges in G
  - a closed-form marginal likelihood $p(\boldsymbol{y}|G)$
- possible with **linear-Gaussian** model and conjugate normal-inverse-Wishart $p(\boldsymbol{\theta}|G)$
- Note: linear-Gaussian model introduces a non-identifiability (called **Markov equivalence**)
- Note: a rich literature on learning DAGs, but going to sidestep it today.

Tool: Bayesian Inference for DAGs

Given a directed acyclic graph G,

$$p_{\boldsymbol{\theta}}(\boldsymbol{y}|G) = \prod_{i=1}^{d} p_{\boldsymbol{\theta}}(\boldsymbol{y}_i|\boldsymbol{y}_{\pi_G(i)})$$



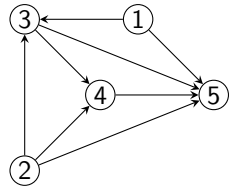
where $\pi_G(i)$ denotes the parents of node i in G.

- the number of DAGs on d vertices is $\geq 2^{\binom{d}{2}}$ (**very large!**)
- this motivates:
  - restricting the set $\mathcal{G}$ of candidate DAGs
  - a complexity prior $p(G) \propto \rho^{|E(G)|}$ where $E(G)$ is the set of edges in G
  - a closed-form marginal likelihood $p(\boldsymbol{y}|G)$
- possible with **linear-Gaussian** model and conjugate normal-inverse-Wishart $p(\boldsymbol{\theta}|G)$
- Note: linear-Gaussian model introduces a non-identifiability (called **Markov equivalence**)
- Note: a rich literature on learning DAGs, but going to sidestep it today.

Tool: Bayesian Inference for DAGs

Given a directed acyclic graph G,

$$p_{\theta}(\boldsymbol{y}|G) = \prod_{i=1}^{d} p_{\theta}(\boldsymbol{y}_i|\boldsymbol{y}_{\pi_G(i)})$$



where $\pi_G(i)$ denotes the parents of node i in G.

- the number of DAGs on d vertices is $\geq 2^{\binom{d}{2}}$ (**very large!**)
- this motivates:
  - restricting the set $\mathcal{G}$ of candidate DAGs
  - a complexity prior $p(G) \propto \rho^{|E(G)|}$ where $E(G)$ is the set of edges in G
  - a closed-form marginal likelihood $p(\boldsymbol{y}|G)$
- possible with **linear-Gaussian** model and conjugate normal-inverse-Wishart $p(\boldsymbol{\theta}|G)$
- Note: linear-Gaussian model introduces a non-identifiability (called **Markov equivalence**)
- Note: a rich literature on learning DAGs, but going to sidestep it today.

Tool: Bayesian Inference for DAGs

Unfortunately, this set-up is grossly misspecified for learning molecular signalling networks:

1. biological knowledge is **inconsistent with a DAG**
2. conjugate regression models are **misspecified**
3. cells are **not identical**
4. there are huge numbers of **unmeasured latent confounders**
5. the effect of the interventions on the protein signalling network is **not well-understood**

Yet, given enough data (and flow cytometry generates a *lot* of data), the Bayesian posterior **collapses onto a single** G (up to Markov equivalence).

Q: Can we use PrO posteriors for learning DAGs?

Tool: Bayesian Inference for DAGs

Unfortunately, this set-up is grossly misspecified for learning molecular signalling networks:

1. biological knowledge is **inconsistent with a DAG**
2. conjugate regression models are **misspecified**
3. cells are **not identical**
4. there are huge numbers of **unmeasured latent confounders**
5. the effect of the interventions on the protein signalling network is **not well-understood**

Yet, given enough data (and flow cytometry generates a *lot* of data), the Bayesian posterior **collapses onto a single** G (up to Markov equivalence).

Q: Can we use PrO posteriors for learning DAGs?

Tool: Bayesian Inference for DAGs

Unfortunately, this set-up is grossly misspecified for learning molecular signalling networks:

1. biological knowledge is **inconsistent with a DAG**
2. conjugate regression models are **misspecified**
3. cells are **not identical**
4. there are huge numbers of **unmeasured latent confounders**
5. the effect of the interventions on the protein signalling network is **not well-understood**

Yet, given enough data (and flow cytometry generates a *lot* of data), the Bayesian posterior **collapses onto a single** G (up to Markov equivalence).

Q: Can we use PrO posteriors for learning DAGs?

Tool: Bayesian Inference for DAGs

Unfortunately, this set-up is grossly misspecified for learning molecular signalling networks:

1. biological knowledge is **inconsistent with a DAG**
2. conjugate regression models are **misspecified**
3. cells are **not identical**
4. there are huge numbers of **unmeasured latent confounders**
5. the effect of the interventions on the protein signalling network is **not well-understood**

Yet, given enough data (and flow cytometry generates a *lot* of data), the Bayesian posterior **collapses onto a single** G (up to Markov equivalence).

Q: Can we use PrO posteriors for learning DAGs?

Tool: Bayesian Inference for DAGs

Unfortunately, this set-up is grossly misspecified for learning molecular signalling networks:

1. biological knowledge is **inconsistent with a DAG**
2. conjugate regression models are **misspecified**
3. cells are **not identical**
4. there are huge numbers of **unmeasured latent confounders**
5. the effect of the interventions on the protein signalling network is **not well-understood**

Yet, given enough data (and flow cytometry generates a *lot* of data), the Bayesian posterior **collapses onto a single** G (up to Markov equivalence).

Q: Can we use PrO posteriors for learning DAGs?

Tool: Bayesian Inference for DAGs

Unfortunately, this set-up is grossly misspecified for learning molecular signalling networks:

1. biological knowledge is **inconsistent with a DAG**
2. conjugate regression models are **misspecified**
3. cells are **not identical**
4. there are huge numbers of **unmeasured latent confounders**
5. the effect of the interventions on the protein signalling network is **not well-understood**

Yet, given enough data (and flow cytometry generates a *lot* of data), the Bayesian posterior **collapses onto a single** G (up to Markov equivalence).

Q: Can we use PrO posteriors for learning DAGs?

Tool: Bayesian Inference for DAGs

Unfortunately, this set-up is grossly misspecified for learning molecular signalling networks:

1. biological knowledge is **inconsistent with a DAG**
2. conjugate regression models are **misspecified**
3. cells are **not identical**
4. there are huge numbers of **unmeasured latent confounders**
5. the effect of the interventions on the protein signalling network is **not well-understood**

Yet, given enough data (and flow cytometry generates a *lot* of data), the Bayesian posterior **collapses onto a single** G (up to Markov equivalence).

Q: Can we use PrO posteriors for learning DAGs?

Tool: Bayesian Inference for DAGs

Unfortunately, this set-up is grossly misspecified for learning molecular signalling networks:

1. biological knowledge is **inconsistent with a DAG**
2. conjugate regression models are **misspecified**
3. cells are **not identical**
4. there are huge numbers of **unmeasured latent confounders**
5. the effect of the interventions on the protein signalling network is **not well-understood**

Yet, given enough data (and flow cytometry generates a *lot* of data), the Bayesian posterior **collapses onto a single** G (up to Markov equivalence).

Q: Can we use PrO posteriors for learning DAGs?

Off-the-Shelf PrO Posteriors Fail

Off-the-shelf PrO posteriors construct a mixture model

$$p_Q(\boldsymbol{y}) = \underbrace{\sum_{G \in \mathcal{G}} q(G) p(\boldsymbol{y}|G)}_{\text{marginal likelihood}}, \qquad Q = \sum_{G \in \mathcal{G}} q(G)\delta_G$$

and solve

$$Q_{\mathrm{PrO}} = \arg\min_Q \underbrace{\sum_{i=1}^{n} S\left(p_Q(\cdot), \boldsymbol{y}_i\right)}_{\text{predictive fit}} + \underbrace{\lambda_n \,\mathrm{KL}(Q||Q_0)}_{\text{regularisation}}.$$

But: The marginal likelihood is a **prior predictive**

$$p(\boldsymbol{y}|G) = \int \left[\prod_{i=1}^{n} p_\theta(\boldsymbol{y}_i|G)\right] p(\theta)\, \mathrm{d}\theta$$

and this is **not useful**...

Off-the-Shelf PrO Posteriors Fail

Off-the-shelf PrO posteriors construct a mixture model

$$p_Q(\boldsymbol{y}) = \underbrace{\sum_{G \in \mathcal{G}} q(G) p(\boldsymbol{y}|G)}_{\text{marginal likelihood}}, \qquad Q = \sum_{G \in \mathcal{G}} q(G)\delta_G$$

and solve

$$Q_{\mathrm{PrO}} = \arg\min_Q \underbrace{\sum_{i=1}^{n} S\left(p_Q(\cdot), \boldsymbol{y}_i\right)}_{\text{predictive fit}} + \underbrace{\lambda_n \,\mathrm{KL}(Q||Q_0)}_{\text{regularisation}}.$$

But: The marginal likelihood is a **prior predictive**

$$p(\boldsymbol{y}|G) = \int \left[\prod_{i=1}^{n} p_{\boldsymbol{\theta}}(\boldsymbol{y}_i|G)\right] p(\boldsymbol{\theta}) \,\mathrm{d}\boldsymbol{\theta}$$

and this is **not useful**...

Contribution: PrO posteriors for dependent data

PrO Posteriors for Dependent Data

Let $\boldsymbol{y}_{(-i)}$ denote the dataset $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{i-1}, \boldsymbol{y}_{i+1}, \ldots, \boldsymbol{y}_n\}$.

Idea: Instead of the prior predictive, consider the leave-one-out predictive

$$p_Q(\boldsymbol{y}|\boldsymbol{y}_{(-i)}) = \sum_{G \in \mathcal{G}} q(G) p(\boldsymbol{y}|\boldsymbol{y}_{(-i)}, G)$$

based on the **leave-one-out posterior predictive**

$$p(\boldsymbol{y}|\boldsymbol{y}_{(-i)}, G) = \int p_\theta(\boldsymbol{y}|\boldsymbol{y}_{(-i)}, G) p(\boldsymbol{\theta}|\boldsymbol{y}_{(-i)}) \, \mathrm{d}\boldsymbol{\theta}.$$

PrO Posteriors for Dependent Data

Let $\boldsymbol{y}_{(-i)}$ denote the dataset $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{i-1}, \boldsymbol{y}_{i+1}, \ldots, \boldsymbol{y}_n\}$.

Idea: Instead of the prior predictive, consider the leave-one-out predictive

$$p_Q(\boldsymbol{y}|\boldsymbol{y}_{(-i)}) = \sum_{G \in \mathcal{G}} q(G) p(\boldsymbol{y}|\boldsymbol{y}_{(-i)}, G)$$

based on the **leave-one-out posterior predictive**

$$p(\boldsymbol{y}|\boldsymbol{y}_{(-i)}, G) = \int p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{y}_{(-i)}, G) p(\boldsymbol{\theta}|\boldsymbol{y}_{(-i)}) \, \mathrm{d}\boldsymbol{\theta}.$$

PrO Posteriors for Dependent Data

Let $\boldsymbol{y}_{(-i)}$ denote the dataset $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{i-1}, \boldsymbol{y}_{i+1}, \ldots, \boldsymbol{y}_n\}$.

The *leave-one-out PrO posterior* is defined as

$$Q_{\mathrm{LOO-PrO}} = \arg\min_{Q} \underbrace{\sum_{i=1}^{n} S\left(p_Q(\cdot|\boldsymbol{y}_{(-i)}), \boldsymbol{y}_i\right)}_{\text{LOO predictive fit}} + \underbrace{\lambda_n \, \mathrm{KL}(Q||Q_0)}_{\text{regularisation}}$$

- Could also include covariates $\boldsymbol{x}$, but suppressing for clarity.
- Represents a **general methodology**, but we will specifically focus on learning DAGs.

PrO Posteriors for Dependent Data

Let $\boldsymbol{y}_{(-i)}$ denote the dataset $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{i-1}, \boldsymbol{y}_{i+1}, \ldots, \boldsymbol{y}_n\}$.

The *leave-one-out PrO posterior* is defined as

$$Q_{\mathrm{LOO-PrO}} = \arg\min_{Q} \underbrace{\sum_{i=1}^{n} S\left(p_Q(\cdot|\boldsymbol{y}_{(-i)}), \boldsymbol{y}_i\right)}_{\text{LOO predictive fit}} + \underbrace{\lambda_n \, \mathrm{KL}(Q||Q_0)}_{\text{regularisation}}$$

- Could also include covariates $\boldsymbol{x}$, but suppressing for clarity.
- Represents a **general methodology**, but we will specifically focus on learning DAGs.

PrO Posteriors for Dependent Data

Let $\boldsymbol{y}_{(-i)}$ denote the dataset $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{i-1}, \boldsymbol{y}_{i+1}, \ldots, \boldsymbol{y}_n\}$.

The *leave-one-out PrO posterior* is defined as

$$Q_{\mathrm{LOO-PrO}} = \arg\min_{Q} \underbrace{\sum_{i=1}^{n} S\left(p_Q(\cdot|\boldsymbol{y}_{(-i)}), \boldsymbol{y}_i\right)}_{\text{LOO predictive fit}} + \underbrace{\lambda_n \, \mathrm{KL}(Q||Q_0)}_{\text{regularisation}}$$

- Could also include covariates $\boldsymbol{x}$, but suppressing for clarity.
- Represents a **general methodology**, but we will specifically focus on learning DAGs.

Example: Well-Specified

Suppose there *is* a true DAG

$$G^\star : \ 1 \to 2 \to 3 \to 4$$

and we learn this with a sparsity prior ($\rho = 0.2$). Data are generated according to

$$\begin{aligned} Y_1 &= \varepsilon_1, \\ Y_2 &= 2.2 Y_1 + \varepsilon_2, \\ Y_3 &= -1.3 Y_2 + \varepsilon_3, \\ Y_4 &= 2 Y_3 + \varepsilon_4, \qquad\qquad \epsilon_i \sim \mathcal{N}(0, 0.5^2) \end{aligned}$$

The Markov equivalence class (MEC) of $G^\star$ contains the four DAGs

$$[G^\star] = \{1 \to 2 \to 3 \to 4, \quad 1 \leftarrow 2 \to 3 \to 4, \\ 1 \leftarrow 2 \leftarrow 3 \to 4, \quad 1 \leftarrow 2 \leftarrow 3 \leftarrow 4\}.$$

Example: Well-Specified

Suppose there *is* a true DAG

$$G^\star : \ 1 \to 2 \to 3 \to 4$$

and we learn this with a sparsity prior ($\rho = 0.2$). Data are generated according to

$$\begin{aligned} Y_1 &= \varepsilon_1, \\ Y_2 &= 2.2 Y_1 + \varepsilon_2, \\ Y_3 &= -1.3 Y_2 + \varepsilon_3, \\ Y_4 &= 2 Y_3 + \varepsilon_4, \qquad\qquad \epsilon_i \sim \mathcal{N}(0, 0.5^2) \end{aligned}$$

The Markov equivalence class (MEC) of $G^\star$ contains the four DAGs

$$\begin{aligned} [G^\star] = \{&1 \to 2 \to 3 \to 4, \quad 1 \leftarrow 2 \to 3 \to 4, \\ &1 \leftarrow 2 \leftarrow 3 \to 4, \quad 1 \leftarrow 2 \leftarrow 3 \leftarrow 4\}. \end{aligned}$$

Example: Well-Specified

Suppose there *is* a true DAG

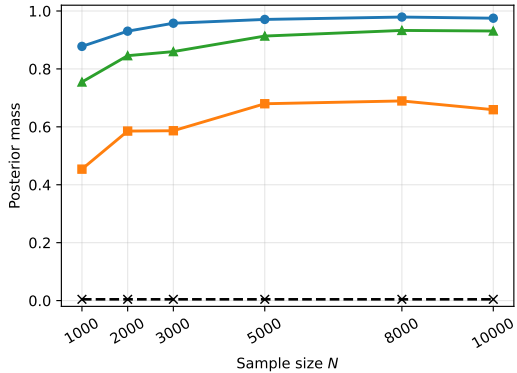
$$G^\star : \ 1 \to 2 \to 3 \to 4$$

and we learn this with a sparsity prior ($\rho = 0.2$). Data are generated according to

$$\begin{aligned} Y_1 &= \varepsilon_1, \\ Y_2 &= 2.2 Y_1 + \varepsilon_2, \\ Y_3 &= -1.3 Y_2 + \varepsilon_3, \\ Y_4 &= 2 Y_3 + \varepsilon_4, \qquad \epsilon_i \sim \mathcal{N}(0, 0.5^2) \end{aligned}$$

The Markov equivalence class (MEC) of $G^\star$ contains the four DAGs

$$\begin{aligned} [G^\star] = \{&1 \to 2 \to 3 \to 4, \quad 1 \leftarrow 2 \to 3 \to 4, \\ &1 \leftarrow 2 \leftarrow 3 \to 4, \quad 1 \leftarrow 2 \leftarrow 3 \leftarrow 4\}. \end{aligned}$$



Posterior mass on true MEC for Bayes, LOO-PrO, and posterior mass on the supergraph MEC in LOO-PrO

Example: Misspecified Regression Function

Now consider the **nonlinear** system

$$
\begin{aligned}
Y_1 &= \varepsilon_1, \\
Y_2 &= 1.2[(1-\lambda)Z_1 + \lambda s_\gamma(Z_1)] + \varepsilon_2 \\
Y_3 &= -1.3[(1-\lambda)Z_2 + \lambda s_\gamma(Z_2)] + \varepsilon_3 \\
Y_4 &= 1.1[(1-\lambda)Z_3 + \lambda s_\gamma(Z_3)] + \varepsilon_4
\end{aligned}
$$

where

$$
Z_j := \frac{Y_j}{\text{std}(Y_j)}, \qquad s_\gamma(z) = \frac{2}{1+\exp(-\gamma z)} - 1
$$

is a nonlinear function (we fix $\gamma = 5$).

Example: Misspecified Regression Function

Now consider the **nonlinear** system

$$
\begin{aligned}
Y_1 &= \varepsilon_1, \\
Y_2 &= 1.2[(1-\lambda)Z_1 + \lambda s_\gamma(Z_1)] + \varepsilon_2 \\
Y_3 &= -1.3[(1-\lambda)Z_2 + \lambda s_\gamma(Z_2)] + \varepsilon_3 \\
Y_4 &= 1.1[(1-\lambda)Z_3 + \lambda s_\gamma(Z_3)] + \varepsilon_4
\end{aligned}
$$

where

$$
Z_j := \frac{Y_j}{\text{std}(Y_j)}, \qquad s_\gamma(z) = \frac{2}{1+\exp(-\gamma z)} - 1
$$

is a nonlinear function (we fix $\gamma = 5$).

Example: Misspecified Regression Function

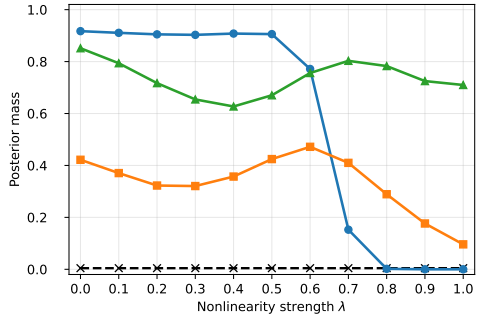
Now consider the **nonlinear** system

$$
\begin{aligned}
Y_1 &= \varepsilon_1, \\
Y_2 &= 1.2[(1-\lambda)Z_1 + \lambda s_\gamma(Z_1)] + \varepsilon_2 \\
Y_3 &= -1.3[(1-\lambda)Z_2 + \lambda s_\gamma(Z_2)] + \varepsilon_3 \\
Y_4 &= 1.1[(1-\lambda)Z_3 + \lambda s_\gamma(Z_3)] + \varepsilon_4
\end{aligned}
$$

where

$$
Z_j := \frac{Y_j}{\text{std}(Y_j)}, \qquad s_\gamma(z) = \frac{2}{1+\exp(-\gamma z)} - 1
$$

is a nonlinear function (we fix $\gamma = 5$).



Posterior mass on true MEC for Bayes, LOO-PrO, and posterior mass on the supergraph MEC in LOO-PrO

Example: Latent Confounding

Now consider the five-node structural equation model

$$Y_0 = \varepsilon_0,$$

$$Y_1 = 1.3Y_0 + \varepsilon_1, \qquad Y_3 = 1.3Y_0 + \varepsilon_3,$$

$$Y_2 = 0.8Y_1 + 0.7Y_3 + \varepsilon_2, \qquad Y_4 = 0.6Y_3 + \varepsilon_4,$$

where **only the variables**

$$(Y_1, Y_2, Y_3, Y_4)$$

are observed.

Thus Y_0 is a latent confounder of Y_1 and Y_3.

Note: No "true" DAG at the level of the observed variables (Y_1, Y_2, Y_3, Y_4).

Example: Latent Confounding

Now consider the five-node structural equation model

$$Y_0 = \varepsilon_0,$$

$$Y_1 = 1.3Y_0 + \varepsilon_1, \qquad Y_3 = 1.3Y_0 + \varepsilon_3,$$

$$Y_2 = 0.8Y_1 + 0.7Y_3 + \varepsilon_2, \qquad Y_4 = 0.6Y_3 + \varepsilon_4,$$

where **only the variables**

$$(Y_1, Y_2, Y_3, Y_4)$$

are observed.

Thus Y_0 is a latent confounder of Y_1 and Y_3.

Note: No "true" DAG at the level of the observed variables (Y_1, Y_2, Y_3, Y_4).

Example: Latent Confounding

Now consider the five-node structural equation model

$$Y_0 = \varepsilon_0,$$

$$Y_1 = 1.3Y_0 + \varepsilon_1, \qquad Y_3 = 1.3Y_0 + \varepsilon_3,$$

$$Y_2 = 0.8Y_1 + 0.7Y_3 + \varepsilon_2, \qquad Y_4 = 0.6Y_3 + \varepsilon_4,$$

where **only the variables**

$$(Y_1, Y_2, Y_3, Y_4)$$

are observed.

Thus Y_0 is a latent confounder of Y_1 and Y_3.

Note: No "true" DAG at the level of the observed variables (Y_1, Y_2, Y_3, Y_4).

Example: Latent Confounding

Now consider the five-node structural equation model

$$Y_0 = \varepsilon_0,$$

$$Y_1 = 1.3Y_0 + \varepsilon_1, \qquad Y_3 = 1.3Y_0 + \varepsilon_3,$$

$$Y_2 = 0.8Y_1 + 0.7Y_3 + \varepsilon_2, \qquad Y_4 = 0.6Y_3 + \varepsilon_4,$$

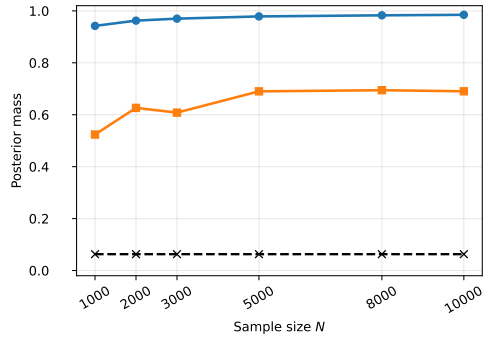
where **only the variables**

$$(Y_1, Y_2, Y_3, Y_4)$$

are observed.

Thus Y_0 is a latent confounder of Y_1 and Y_3.

Note: No "true" DAG at the level of the observed variables (Y_1, Y_2, Y_3, Y_4).



Posterior mass assigned to the *most probable* MEC, according to Bayes and LOO-PrO.

Example: Protein Kinase Signalling Network

The flow cytometry data of Sachs et al. [2005] were used to learn a protein kinase signalling network, with:

- an informative prior; at most two edges can differ from G_{ref}
- interventions explicitly included as covariates $\boldsymbol{x}$ (so can "look inside the MEC")

Method	Rank	DAG mass	Local modification of G_{ref}
Bayes	1	1.0000	add $\text{Erk} \to \text{Akt}$; add $\text{Mek} \to \text{Akt}$
LOO-PrO	1	0.9130	add $\text{Erk} \to \text{Akt}$; add $\text{Mek} \to \text{Akt}$
LOO-PrO	2	0.0212	add $\text{Mek} \to \text{Akt}$; add $\text{Mek} \to \text{P38}$
LOO-PrO	3	0.0203	add $\text{Erk} \to \text{Akt}$; add $\text{Mek} \to \text{P38}$
LOO-PrO	4	0.0129	add $\text{Jnk} \to \text{P38}$; add $\text{Mek} \to \text{P38}$
LOO-PrO	5	0.0073	add $\text{Erk} \to \text{Akt}$; delete $\text{PKC} \to \text{P38}$

Q: Does LOO-PrO reflect uncertainty due to misspecification, or just limited data?

Example: Protein Kinase Signalling Network

The flow cytometry data of Sachs et al. [2005] were used to learn a protein kinase signalling network, with:

- an informative prior; at most two edges can differ from G_{ref}
- interventions explicitly included as covariates $\boldsymbol{x}$ (so can "look inside the MEC")

Method	Rank	DAG mass	Local modification of G_{ref}
Bayes	1	1.0000	add $\text{Erk} \rightarrow \text{Akt}$; add $\text{Mek} \rightarrow \text{Akt}$
LOO-PrO	1	0.9130	add $\text{Erk} \rightarrow \text{Akt}$; add $\text{Mek} \rightarrow \text{Akt}$
LOO-PrO	2	0.0212	add $\text{Mek} \rightarrow \text{Akt}$; add $\text{Mek} \rightarrow \text{P38}$
LOO-PrO	3	0.0203	add $\text{Erk} \rightarrow \text{Akt}$; add $\text{Mek} \rightarrow \text{P38}$
LOO-PrO	4	0.0129	add $\text{Jnk} \rightarrow \text{P38}$; add $\text{Mek} \rightarrow \text{P38}$
LOO-PrO	5	0.0073	add $\text{Erk} \rightarrow \text{Akt}$; delete $\text{PKC} \rightarrow \text{P38}$

Q: Does LOO-PrO reflect uncertainty due to misspecification, or just limited data?

Example: Protein Kinase Signalling Network

The flow cytometry data of Sachs et al. [2005] were used to learn a protein kinase signalling network, with:

- an informative prior; at most two edges can differ from G_{ref}
- interventions explicitly included as covariates $\boldsymbol{x}$ (so can "look inside the MEC")

Method	Rank	DAG mass	Local modification of G_{ref}
Bayes	1	1.0000	add Erk $\rightarrow$ Akt; add Mek $\rightarrow$ Akt
LOO-PrO	1	0.9130	add Erk $\rightarrow$ Akt; add Mek $\rightarrow$ Akt
LOO-PrO	2	0.0212	add Mek $\rightarrow$ Akt; add Mek $\rightarrow$ P38
LOO-PrO	3	0.0203	add Erk $\rightarrow$ Akt; add Mek $\rightarrow$ P38
LOO-PrO	4	0.0129	add Jnk $\rightarrow$ P38; add Mek $\rightarrow$ P38
LOO-PrO	5	0.0073	add Erk $\rightarrow$ Akt; delete PKC $\rightarrow$ P38

Q: Does LOO-PrO reflect uncertainty due to misspecification, or just limited data?

Example: Protein Kinase Signalling Network

The flow cytometry data of Sachs et al. [2005] were used to learn a protein kinase signalling network, with:

- an informative prior; at most two edges can differ from G_{ref}
- interventions explicitly included as covariates $\boldsymbol{x}$ (so can "look inside the MEC")

Method	Rank	DAG mass	Local modification of G_{ref}
Bayes	1	1.0000	add $\text{Erk} \to \text{Akt}$; add $\text{Mek} \to \text{Akt}$
LOO-PrO	1	0.9130	add $\text{Erk} \to \text{Akt}$; add $\text{Mek} \to \text{Akt}$
LOO-PrO	2	0.0212	add $\text{Mek} \to \text{Akt}$; add $\text{Mek} \to \text{P38}$
LOO-PrO	3	0.0203	add $\text{Erk} \to \text{Akt}$; add $\text{Mek} \to \text{P38}$
LOO-PrO	4	0.0129	add $\text{Jnk} \to \text{P38}$; add $\text{Mek} \to \text{P38}$
LOO-PrO	5	0.0073	add $\text{Erk} \to \text{Akt}$; delete $\text{PKC} \to \text{P38}$

Q: Does LOO-PrO reflect uncertainty due to misspecification, or just limited data?

Example: Protein Kinase Signalling Network

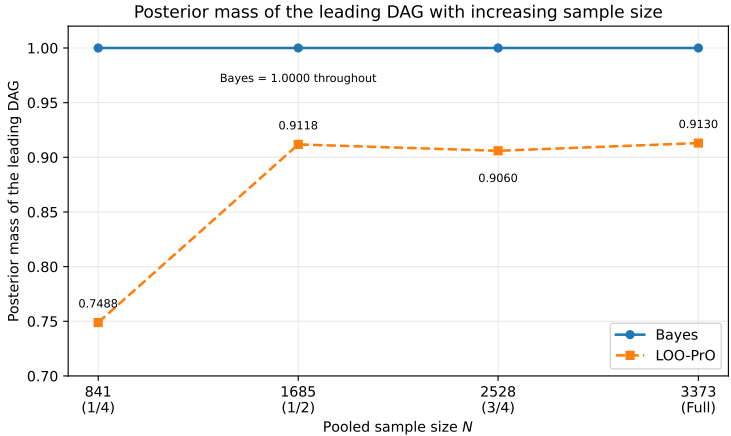
The flow cytometry data of Sachs et al. [2005] were used to learn a protein kinase signalling network, with:

- an informative prior; at most two edges can differ from G_{ref}
- interventions explicitly included as covariates $\boldsymbol{x}$ (so can "look inside the MEC")

Method	Rank	DAG mass	Local modification of G_{ref}
Bayes	1	1.0000	add $\text{Erk} \to \text{Akt}$; add $\text{Mek} \to \text{Akt}$
LOO-PrO	1	0.9130	add $\text{Erk} \to \text{Akt}$; add $\text{Mek} \to \text{Akt}$
LOO-PrO	2	0.0212	add $\text{Mek} \to \text{Akt}$; add $\text{Mek} \to \text{P38}$
LOO-PrO	3	0.0203	add $\text{Erk} \to \text{Akt}$; add $\text{Mek} \to \text{P38}$
LOO-PrO	4	0.0129	add $\text{Jnk} \to \text{P38}$; add $\text{Mek} \to \text{P38}$
LOO-PrO	5	0.0073	add $\text{Erk} \to \text{Akt}$; delete $\text{PKC} \to \text{P38}$

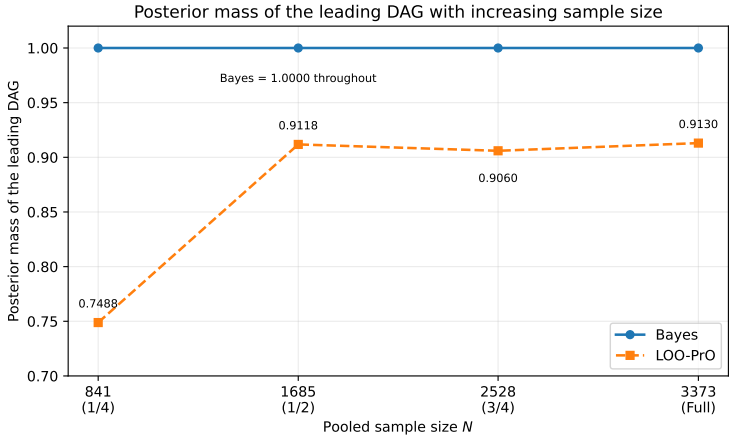
Q: Does LOO-PrO reflect uncertainty due to misspecification, or just limited data?

Example: Protein Kinase Signalling Network



$\therefore$ the LOO-PrO posterior has adapted to misspecification, while Bayes has not

Example: Protein Kinase Signalling Network



∴ the LOO-PrO posterior has adapted to misspecification, while Bayes has not

Wrapping Up - An Invitation to Get Involved!

- Post-Bayes seminar series (`https://postbayes.github.io/seminar/`)
- Advances in Post-Bayesian Methods, 15-16 May 2025, London, UK.
- Optimisation and Post-Bayesian Inference in Machine Learning, 5 May 2026, Tangier, Morocco.
- Post-Bayesian Statistics, 3 August 2026, USyd Business School, Australia.
- Rethinking the Role of Bayesianism in the Age of Modern AI, 26-30 October 2026, Edinburgh, UK.
- Prediction-Centric Uncertainty Quantification, 9-13 November 2026, London, UK.
- Post-Bayes on the Beach, 8 December 2026, Newcastle, UK.

Thank you for your attention!

Wrapping Up - An Invitation to Get Involved!

- Post-Bayes seminar series (`https://postbayes.github.io/seminar/`)
- ~~Advances in Post-Bayesian Methods, 15-16 May 2025, London, UK.~~
- ~~Optimisation and Post-Bayesian Inference in Machine Learning, 5 May 2026, Tangier, Morocco.~~
- ~~Post-Bayesian Statistics, 3 August 2026, USyd Business School, Australia.~~
- Rethinking the Role of Bayesianism in the Age of Modern AI, 26-30 October 2026, Edinburgh, UK.
- Prediction-Centric Uncertainty Quantification, 9-13 November 2026, London, UK.
- Post-Bayes on the Beach, 8 December 2026, Newcastle, UK.

Thank you for your attention!

Wrapping Up - An Invitation to Get Involved!

- Post-Bayes seminar series (`https://postbayes.github.io/seminar/`)
- ~~Advances in Post-Bayesian Methods, 15-16 May 2025, London, UK.~~
- ~~Optimisation and Post-Bayesian Inference in Machine Learning, 5 May 2026, Tangier, Morocco.~~
- ~~Post-Bayesian Statistics, 3 August 2026, USyd Business School, Australia.~~
- Rethinking the Role of Bayesianism in the Age of Modern AI, 26-30 October 2026, Edinburgh, UK.
- Prediction-Centric Uncertainty Quantification, 9-13 November 2026, London, UK.
- Post-Bayes on the Beach, 8 December 2026, Newcastle, UK.

Thank you for your attention!

Wrapping Up - An Invitation to Get Involved!

- Post-Bayes seminar series (`https://postbayes.github.io/seminar/`)
- ~~Advances in Post-Bayesian Methods, 15-16 May 2025, London, UK.~~
- ~~Optimisation and Post-Bayesian Inference in Machine Learning, 5 May 2026, Tangier, Morocco.~~
- ~~Post-Bayesian Statistics, 3 August 2026, USyd Business School, Australia.~~
- Rethinking the Role of Bayesianism in the Age of Modern AI, 26-30 October 2026, Edinburgh, UK.
- Prediction-Centric Uncertainty Quantification, 9-13 November 2026, London, UK.
- Post-Bayes on the Beach, 8 December 2026, Newcastle, UK.

Thank you for your attention!

Wrapping Up - An Invitation to Get Involved!

- Post-Bayes seminar series (`https://postbayes.github.io/seminar/`)
- ~~Advances in Post-Bayesian Methods, 15-16 May 2025, London, UK.~~
- ~~Optimisation and Post-Bayesian Inference in Machine Learning, 5 May 2026, Tangier, Morocco.~~
- ~~Post-Bayesian Statistics, 3 August 2026, USyd Business School, Australia.~~
- Rethinking the Role of Bayesianism in the Age of Modern AI, 26-30 October 2026, Edinburgh, UK.
- Prediction-Centric Uncertainty Quantification, 9-13 November 2026, London, UK.
- Post-Bayes on the Beach, 8 December 2026, Newcastle, UK.

Thank you for your attention!

Wrapping Up - An Invitation to Get Involved!

- Post-Bayes seminar series (`https://postbayes.github.io/seminar/`)
- ~~Advances in Post-Bayesian Methods, 15-16 May 2025, London, UK.~~
- ~~Optimisation and Post-Bayesian Inference in Machine Learning, 5 May 2026, Tangier, Morocco.~~
- ~~Post-Bayesian Statistics, 3 August 2026, USyd Business School, Australia.~~
- Rethinking the Role of Bayesianism in the Age of Modern AI, 26-30 October 2026, Edinburgh, UK.
- Prediction-Centric Uncertainty Quantification, 9-13 November 2026, London, UK.
- Post-Bayes on the Beach, 8 December 2026, Newcastle, UK.

Thank you for your attention!

References

P. G. Bissiri, C. C. Holmes, and S. G. Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society, Series B*, 78(5):1103, 2016.

C. Jordan-Squire. *Convex Optimization over Probability Measures*. PhD thesis, University of Washington, 2015.

J. Knoblauch, J. Jewson, and T. Damoulas. An optimization-centric view on Bayes' rule: Reviewing and generalizing variational inference. *Journal of Machine Learning Research*, 23(132):1–109, 2022.

N. Laird. Nonparametric maximum likelihood estimation of a mixing distribution. *Journal of the American Statistical Association*, 73(364):805–811, 1978.

B. G. Lindsay. *Mixture Models: Theory, Geometry, and Applications*. Institute of Mathematical Statistics, 1995.

Y. McLatchie, B.-E. Cherief-Abdellatif, D. T. Frazier, and J. Knoblauch. Predictively oriented posteriors. *arXiv preprint arXiv:2510.01915*, 2025.

K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529, 2005.

Z. Shen, J. Knoblauch, S. Power, and C. J. Oates. Prediction-centric uncertainty quantification via MMD. In *AISTATS*, 2026.

Example: Covid Model

Susceptible, **I**nfected, **R**ecovered (S-I-R):

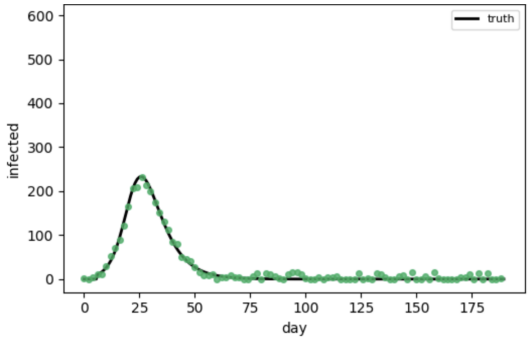
$$\frac{\mathrm{d}S}{\mathrm{d}x} = -\theta_1 SI \qquad S(\theta_3) = S_0$$
$$\frac{\mathrm{d}I}{\mathrm{d}x} = \theta_1 SI - \theta_2 I \qquad I(\theta_3) = I_0$$
$$\frac{\mathrm{d}R}{\mathrm{d}x} = \theta_2 I \qquad R(\theta_3) = R_0$$

Parameters:

$\theta_1 =$ contact rate

$\theta_2 =$ recovery rate

$\theta_3 =$ start time offset



Example: Covid Model

Susceptible, **I**nfected, **R**ecovered (S-I-R):

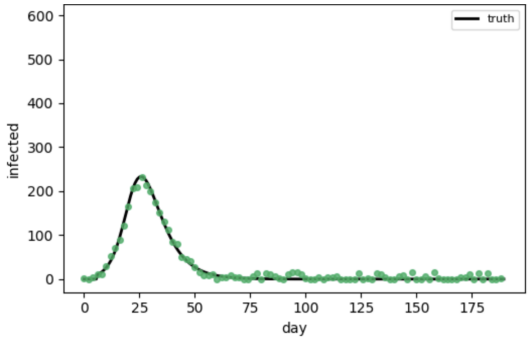
$$\frac{\mathrm{d}S}{\mathrm{d}x} = -\theta_1 SI \qquad S(\theta_3) = S_0$$
$$\frac{\mathrm{d}I}{\mathrm{d}x} = \theta_1 SI - \theta_2 I \qquad I(\theta_3) = I_0$$
$$\frac{\mathrm{d}R}{\mathrm{d}x} = \theta_2 I \qquad R(\theta_3) = R_0$$

Parameters:

$\theta_1 =$ contact rate

$\theta_2 =$ recovery rate

$\theta_3 =$ start time offset



Example: Covid Model

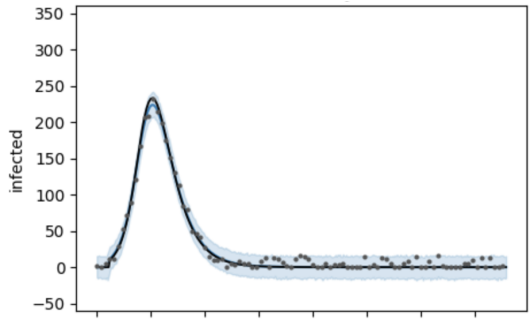
Statistical Model:

$$y_i | x_i \sim \mathcal{N}(I(x_i), \sigma^2) \quad (x_i = \text{day of } i\text{th datum})$$

Prior:

$$\log(\theta_1) \sim \mathcal{N}(\log(0.4), 0.6^2)$$
$$\log(\theta_2) \sim \mathcal{N}(\log(0.17), 0.6^2)$$
$$\theta_3 \sim \mathcal{N}(10, 45^2)$$

Posterior Predictive:



Example: Covid Model

Statistical Model:

$$y_i | x_i \sim \mathcal{N}(I(x_i), \sigma^2) \quad (x_i = \text{day of } i\text{th datum})$$

Prior:

$$\log(\theta_1) \sim \mathcal{N}(\log(0.4), 0.6^2)$$
$$\log(\theta_2) \sim \mathcal{N}(\log(0.17), 0.6^2)$$
$$\theta_3 \sim \mathcal{N}(10, 45^2)$$

Posterior Predictive:

