Bayesian smoothing with extended second order random walk model: An detailed overview and comparison

Ziang Zhang

Supervisors: Patrick Brown
James Stafford

Department of Statistics, University of Toronto

November 2021

Outline

1 Introduction
- Why Bayesian smoothing spline?
- Exact method with ARIMA prior

2 Extended RW2 method
- SDE-based prior
- Finite element method
- Extended RW2

3 Computations

4 Examples
- Simulation Studies
- C02 Concentration Data

5 Conclusions

Smoothing Spline Problem

Consider a data set $\{y_i, x_i, i \in [n]\}$, and a nonparametric model $y_i = g(x_i) + \epsilon_i$ where $\epsilon_i \overset{iid}{\sim} N(0, \sigma_\epsilon^2)$ and $x_i \in [a, b]$, then the (traditional) smoothing spline aims to solve the following problem:

$$\arg\min_{g \in C^2} \left\{ \sum_i \left(\frac{y_i - g(x_i)}{\sigma_\epsilon} \right)^2 + \lambda \int_a^b g''(x)^2 dx \right\} \qquad (1)$$

The sum of square term on the left can be replaced by negative log likelihood, which is also called *penalized likelihood* method.

Question: How to incorporate the uncertainty from estimating σ_ξ and λ into the inferences?

One Solution: Bayesian hierarchical model, which provides model-based estimation and uncertainty quantification for all the parameters.

Smoothing Spline Problem

Consider a data set $\{y_i, x_i, i \in [n]\}$, and a nonparametric model $y_i = g(x_i) + \epsilon_i$ where $\epsilon_i \overset{iid}{\sim} N(0, \sigma_\epsilon^2)$ and $x_i \in [a, b]$, then the (traditional) smoothing spline aims to solve the following problem:

$$\arg\min_{g \in C^2} \left\{ \sum_i \left(\frac{y_i - g(x_i)}{\sigma_\epsilon} \right)^2 + \lambda \int_a^b g''(x)^2 dx \right\} \tag{1}$$

The sum of square term on the left can be replaced by negative log likelihood, which is also called *penalized likelihood* method.

Question: How to incorporate the uncertainty from estimating σ_ξ and λ into the inferences?

One Solution: Bayesian hierarchical model, which provides model-based estimation and uncertainty quantification for all the parameters.

Smoothing Spline Problem

Consider a data set $\{y_i, x_i, i \in [n]\}$, and a nonparametric model $y_i = g(x_i) + \epsilon_i$ where $\epsilon_i \overset{iid}{\sim} N(0, \sigma_\epsilon^2)$ and $x_i \in [a, b]$, then the (traditional) smoothing spline aims to solve the following problem:

$$\arg\min_{g \in C^2} \left\{ \sum_i \left(\frac{y_i - g(x_i)}{\sigma_\epsilon} \right)^2 + \lambda \int_a^b g''(x)^2 dx \right\} \tag{1}$$

The sum of square term on the left can be replaced by negative log likelihood, which is also called *penalized likelihood* method.

Question: How to incorporate the uncertainty from estimating σ_ξ and λ into the inferences?

One Solution: Bayesian hierarchical model, which provides model-based estimation and uncertainty quantification for all the parameters.

Smoothing Spline Problem

Consider a data set $\{y_i, x_i, i \in [n]\}$, and a nonparametric model $y_i = g(x_i) + \epsilon_i$ where $\epsilon_i \overset{iid}{\sim} N(0, \sigma_\epsilon^2)$ and $x_i \in [a, b]$, then the (traditional) smoothing spline aims to solve the following problem:

$$\arg\min_{g \in C^2} \left\{ \sum_i \left(\frac{y_i - g(x_i)}{\sigma_\epsilon} \right)^2 + \lambda \int_a^b g''(x)^2 dx \right\} \tag{1}$$

The sum of square term on the left can be replaced by negative log likelihood, which is also called *penalized likelihood* method.

Question: How to incorporate the uncertainty from estimating σ_ξ and λ into the inferences?

One Solution: Bayesian hierarchical model, which provides model-based estimation and uncertainty quantification for all the parameters.

Smoothing Spline Problem

Consider a data set $\{y_i, x_i, i \in [n]\}$, and a nonparametric model $y_i = g(x_i) + \epsilon_i$ where $\epsilon_i \overset{iid}{\sim} N(0, \sigma_\epsilon^2)$ and $x_i \in [a, b]$, then the (traditional) smoothing spline aims to solve the following problem:

$$\arg\min_{g \in C^2} \left\{ \sum_i \left(\frac{y_i - g(x_i)}{\sigma_\epsilon} \right)^2 + \lambda \int_a^b g''(x)^2 dx \right\} \tag{1}$$

The sum of square term on the left can be replaced by negative log likelihood, which is also called *penalized likelihood* method.

Question: How to incorporate the uncertainty from estimating σ_ξ and λ into the inferences?

One Solution: Bayesian hierarchical model, which provides model-based estimation and uncertainty quantification for all the parameters.

Vectorized expression of smoothing spline:

Using the property of natural cubic spline, the term $\int_a^b g''(x)^2 dx$ for any natural cubic spline $g(.)$ can be written as $\boldsymbol{g}^T K \boldsymbol{g}$. Therefore, the equation 1 can be written in the following vector form:

$$\frac{1}{\sigma_\epsilon^2}(\boldsymbol{y}-\boldsymbol{g})^T(\boldsymbol{y}-\boldsymbol{g})+\lambda \boldsymbol{g}^T K \boldsymbol{g}. \qquad (2)$$

Consider without the loss of generality that covariates are equally spaced, then the matrix K can be factorized as the following:

$$K = D^T R^{-1} D. \qquad (3)$$

The $(n-2)\times n$ matrix D is a second-order differencing matrix, and the $(n-2)\times(n-2)$ matrix R^{-1} can be shown to correspond to the precision matrix of a MA(1) process (Brown and De Jong, 2001).

Vectorized expression of smoothing spline:

Using the property of natural cubic spline, the term $\int_a^b g''(x)^2 dx$ for any natural cubic spline $g(.)$ can be written as $\boldsymbol{g}^T K \boldsymbol{g}$. Therefore, the equation 1 can be written in the following vector form:

$$\frac{1}{\sigma_\epsilon^2}(\boldsymbol{y}-\boldsymbol{g})^T(\boldsymbol{y}-\boldsymbol{g})+\lambda \boldsymbol{g}^T K \boldsymbol{g}. \tag{2}$$

Consider without the loss of generality that covariates are equally spaced, then the matrix K can be factorized as the following:

$$K = D^T R^{-1} D. \tag{3}$$

The $(n-2)\times n$ matrix D is a second-order differencing matrix, and the $(n-2)\times(n-2)$ matrix R^{-1} can be shown to correspond to the precision matrix of a MA(1) process (Brown and De Jong, 2001).

Vectorized expression of smoothing spline:

Using the property of natural cubic spline, the term $\int_a^b g''(x)^2 dx$ for any natural cubic spline $g(.)$ can be written as $\boldsymbol{g}^T K \boldsymbol{g}$. Therefore, the equation 1 can be written in the following vector form:

$$\frac{1}{\sigma_\epsilon^2}(\boldsymbol{y}-\boldsymbol{g})^T(\boldsymbol{y}-\boldsymbol{g})+\lambda \boldsymbol{g}^T K \boldsymbol{g}. \tag{2}$$

Consider without the loss of generality that covariates are equally spaced, then the matrix K can be factorized as the following:

$$K = D^T R^{-1} D. \tag{3}$$

The $(n-2)\times n$ matrix D is a second-order differencing matrix, and the $(n-2)\times(n-2)$ matrix R^{-1} can be shown to correspond to the precision matrix of a MA(1) process (Brown and De Jong, 2001).

Vectorized expression of smoothing spline:

Using the property of natural cubic spline, the term $\int_a^b g''(x)^2 dx$ for any natural cubic spline $g(.)$ can be written as $\mathbf{g}^T K \mathbf{g}$. Therefore, the equation 1 can be written in the following vector form:

$$\frac{1}{\sigma_\epsilon^2}(\mathbf{y}-\mathbf{g})^T(\mathbf{y}-\mathbf{g})+\lambda \mathbf{g}^T K \mathbf{g}. \tag{2}$$

Consider without the loss of generality that covariates are equally spaced, then the matrix K can be factorized as the following:

$$K = D^T R^{-1} D. \tag{3}$$

The $(n-2)\times n$ matrix D is a second-order differencing matrix, and the $(n-2)\times(n-2)$ matrix R^{-1} can be shown to correspond to the precision matrix of a MA(1) process (Brown and De Jong, 2001).

Vectorized expression of smoothing spline:

Using the property of natural cubic spline, the term $\int_a^b g''(x)^2 dx$ for any natural cubic spline $g(.)$ can be written as $\boldsymbol{g}^T K \boldsymbol{g}$. Therefore, the equation 1 can be written in the following vector form:

$$\frac{1}{\sigma_\epsilon^2}(\boldsymbol{y}-\boldsymbol{g})^T(\boldsymbol{y}-\boldsymbol{g})+\lambda \boldsymbol{g}^T K \boldsymbol{g}. \tag{2}$$

Consider without the loss of generality that covariates are equally spaced, then the matrix K can be factorized as the following:

$$K = D^T R^{-1} D. \tag{3}$$

The $(n-2)\times n$ matrix D is a second-order differencing matrix, and the $(n-2)\times(n-2)$ matrix R^{-1} can be shown to correspond to the precision matrix of a MA(1) process (Brown and De Jong, 2001).

Specifically when covariates are unit spaced, we have the following expressions for D and R:

$$D = \begin{bmatrix} 1 & -2 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 & 0 \\ & \vdots & & & \ddots & & \\ 0 & \vdots & & & \ddots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & -2 & 1 \end{bmatrix}, \quad R = \begin{bmatrix} \frac{2}{3} & \frac{1}{6} & 0 & 0 & 0 & \cdots & 0 \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & \cdots & 0 \\ & & & & & \ddots & \\ 0 & 0 & 0 & \cdots & 0 & \frac{1}{6} & \frac{2}{3} \end{bmatrix} \tag{4}$$

Problem: This ARIMA(0,2,1) prior interpretation of smoothing spline will only be valid when all locations are equally spaced. Otherwise the locations should be refined into finer equally spaced resolution first.

Problem: R^{-1} will be a dense matrix, and hence the precision matrix of $\boldsymbol{g}$ will be a dense matrix as well. Computation will be hard and not compatible with inference method such as Integrated Nested Laplace Approximation(INLA) (Rue et al., 2009).

Specifically when covariates are unit spaced, we have the following expressions for D and R:

$$D = \begin{bmatrix} 1 & -2 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 & 0 \\ & \vdots & & & \ddots & & \\ 0 & \vdots & & & \ddots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & -2 & 1 \end{bmatrix}, \quad R = \begin{bmatrix} \frac{2}{3} & \frac{1}{6} & 0 & 0 & 0 & \cdots & 0 \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & \cdots & 0 \\ & & & & & \ddots & \\ 0 & 0 & 0 & \cdots & 0 & \frac{1}{6} & \frac{2}{3} \end{bmatrix} \tag{4}$$

Problem: This ARIMA(0,2,1) prior interpretation of smoothing spline will only be valid when all locations are equally spaced. Otherwise the locations should be refined into finer equally spaced resolution first.

Problem: R^{-1} will be a dense matrix, and hence the precision matrix of $\boldsymbol{g}$ will be a dense matrix as well. Computation will be hard and not compatible with inference method such as Integrated Nested Laplace Approximation(INLA) (Rue et al., 2009).

Specifically when covariates are unit spaced, we have the following expressions for D and R:

$$D = \begin{bmatrix} 1 & -2 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 & 0 \\ \vdots & & & & \ddots & & \\ 0 & \vdots & & & \ddots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & -2 & 1 \end{bmatrix}, \quad R = \begin{bmatrix} \frac{2}{3} & \frac{1}{6} & 0 & 0 & 0 & \cdots & 0 \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & \cdots & 0 \\ & & & & & \ddots & \\ 0 & 0 & 0 & \cdots & 0 & \frac{1}{6} & \frac{2}{3} \end{bmatrix} \tag{4}$$

Problem: This ARIMA(0,2,1) prior interpretation of smoothing spline will only be valid when all locations are equally spaced. Otherwise the locations should be refined into finer equally spaced resolution first.

Problem: R^{-1} will be a dense matrix, and hence the precision matrix of $\boldsymbol{g}$ will be a dense matrix as well. Computation will be hard and not compatible with inference method such as Integrated Nested Laplace Approximation(INLA) (Rue et al., 2009).

Specifically when covariates are unit spaced, we have the following expressions for D and R:

$$D = \begin{bmatrix} 1 & -2 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 & 0 \\ & \vdots & & & \ddots & & \\ 0 & \vdots & & & & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & -2 & 1 \end{bmatrix}, \quad R = \begin{bmatrix} \frac{2}{3} & \frac{1}{6} & 0 & 0 & 0 & \cdots & 0 \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & \cdots & 0 \\ & & & & & \ddots & \\ 0 & 0 & 0 & \cdots & 0 & \frac{1}{6} & \frac{2}{3} \end{bmatrix} \tag{4}$$

Problem: This ARIMA(0,2,1) prior interpretation of smoothing spline will only be valid when all locations are equally spaced. Otherwise the locations should be refined into finer equally spaced resolution first.

Problem: R^{-1} will be a dense matrix, and hence the precision matrix of $\boldsymbol{g}$ will be a dense matrix as well. Computation will be hard and not compatible with inference method such as Integrated Nested Laplace Approximation(INLA) (Rue et al., 2009).

1. From result of Wahba (1978), there is a well known connection between smoothing spline and folded Wiener process prior:
2. Let $W(t)$ denote the standard Wiener's process (Brownian motion), a SDE based prior is assigned to $g(t)$ in the following way ($\sigma_s = 1/\sqrt{\lambda}$):

$$\frac{d^2 g(t)}{dt^2} = \sigma_s \frac{dW(t)}{dt}.$$

3. The derivative of $W(t)$ does not exist in ordinary definition, but can be defined as a generalized function, the *white noise* process.
4. If $g(0)$ and $g'(0)$ are given diffuse Gaussian prior, the limiting posterior mean of $\boldsymbol{g}$ will be the minimizer of the smoothing spline problem (Wahba, 1978).

1. From result of Wahba (1978), there is a well known connection between smoothing spline and folded Wiener process prior:
2. Let $W(t)$ denote the standard Wiener's process (Brownian motion), a SDE based prior is assigned to $g(t)$ in the following way ($\sigma_s = 1/\sqrt{\lambda}$):

$$\frac{d^2 g(t)}{dt^2} = \sigma_s \frac{dW(t)}{dt}.$$

3. The derivative of $W(t)$ does not exist in ordinary definition, but can be defined as a generalized function, the *white noise* process.
4. If $g(0)$ and $g'(0)$ are given diffuse Gaussian prior, the limiting posterior mean of $\boldsymbol{g}$ will be the minimizer of the smoothing spline problem (Wahba, 1978).

1. From result of Wahba (1978), there is a well known connection between smoothing spline and folded Wiener process prior:
2. Let $W(t)$ denote the standard Wiener's process (Brownian motion), a SDE based prior is assigned to $g(t)$ in the following way ($\sigma_s = 1/\sqrt{\lambda}$):

$$\frac{d^2 g(t)}{dt^2} = \sigma_s \frac{dW(t)}{dt}.$$

3. The derivative of $W(t)$ does not exist in ordinary definition, but can be defined as a generalized function, the *white noise* process.
4. If $g(0)$ and $g'(0)$ are given diffuse Gaussian prior, the limiting posterior mean of g will be the minimizer of the smoothing spline problem (Wahba, 1978).

1. From result of Wahba (1978), there is a well known connection between smoothing spline and folded Wiener process prior:
2. Let $W(t)$ denote the standard Wiener's process (Brownian motion), a SDE based prior is assigned to $g(t)$ in the following way ($\sigma_s = 1/\sqrt{\lambda}$):

$$\frac{d^2 g(t)}{dt^2} = \sigma_s \frac{dW(t)}{dt}.$$

3. The derivative of $W(t)$ does not exist in ordinary definition, but can be defined as a generalized function, the *white noise* process.
4. If $g(0)$ and $g'(0)$ are given diffuse Gaussian prior, the limiting posterior mean of $\boldsymbol{g}$ will be the minimizer of the smoothing spline problem (Wahba, 1978).

Definition (Finite Element Method)

Let $\mathbb{B}_p := \{\varphi_i, i \in [p]\}$ denote the set of p pre-specified basis functions, and let $\mathbb{T}_q := \{\phi_i, i \in [q]\}$ denote the set of q pre-specified test functions. The finite element approximation $\tilde{g}(.)$ to the true function $g(.)$ is defined as:

$$\tilde{g}(.) = \sum_{i=1}^{p} w_i \varphi_i(.), \tag{5}$$

where $\boldsymbol{w} := (w_1, ..., w_p)^T \in \mathbb{R}^p$ is a set of weights that satisfies:

$$\langle \frac{d^2\tilde{g}(t)}{dt^2}, \phi_i(t) \rangle \overset{d}{=} \langle \sigma_s \frac{dW(t)}{dt}, \phi_i(t) \rangle, \tag{6}$$

for any test function $\phi_i \in \mathbb{T}_q$.

Definition (Finite Element Method)

Let $\mathbb{B}_p := \{\varphi_i, i \in [p]\}$ denote the set of p pre-specified basis functions, and let $\mathbb{T}_q := \{\phi_i, i \in [q]\}$ denote the set of q pre-specified test functions. The finite element approximation $\tilde{g}(.)$ to the true function $g(.)$ is defined as:

$$\tilde{g}(.) = \sum_{i=1}^{p} w_i \varphi_i(.), \tag{5}$$

where $\boldsymbol{w} := (w_1, ..., w_p)^T \in \mathbb{R}^p$ is a set of weights that satisfies:

$$\langle \frac{d^2 \tilde{g}(t)}{dt^2}, \phi_i(t) \rangle \stackrel{d}{=} \langle \sigma_s \frac{dW(t)}{dt}, \phi_i(t) \rangle, \tag{6}$$

for any test function $\phi_i \in \mathbb{T}_q$.

Definition (Finite Element Method)

Let $\mathbb{B}_p := \{\varphi_i, i \in [p]\}$ denote the set of p pre-specified basis functions, and let $\mathbb{T}_q := \{\phi_i, i \in [q]\}$ denote the set of q pre-specified test functions. The finite element approximation $\tilde{g}(.)$ to the true function $g(.)$ is defined as:

$$\tilde{g}(.) = \sum_{i=1}^{p} w_i \varphi_i(.), \tag{5}$$

where $\boldsymbol{w} := (w_1, ..., w_p)^T \in \mathbb{R}^p$ is a set of weights that satisfies:

$$\langle \frac{d^2 \tilde{g}(t)}{dt^2}, \phi_i(t) \rangle \stackrel{d}{=} \langle \sigma_s \frac{dW(t)}{dt}, \phi_i(t) \rangle, \tag{6}$$

for any test function $\phi_i \in \mathbb{T}_q$.

Definition (Finite Element Method)

Let $\mathbb{B}_p := \{\varphi_i, i \in [p]\}$ denote the set of p pre-specified basis functions, and let $\mathbb{T}_q := \{\phi_i, i \in [q]\}$ denote the set of q pre-specified test functions. The finite element approximation $\tilde{g}(.)$ to the true function $g(.)$ is defined as:

$$\tilde{g}(.) = \sum_{i=1}^{p} w_i \varphi_i(.), \tag{5}$$

where $\boldsymbol{w} := (w_1, ..., w_p)^T \in \mathbb{R}^p$ is a set of weights that satisfies:

$$\langle \frac{d^2 \tilde{g}(t)}{dt^2}, \phi_i(t) \rangle \stackrel{d}{=} \langle \sigma_s \frac{dW(t)}{dt}, \phi_i(t) \rangle, \tag{6}$$

for any test function $\phi_i \in \mathbb{T}_q$.

- Lindgren and Rue (2008) applied the finite element method above to approximate the SDE-based prior, setting both $\mathbb{B}_p$ and $\mathbb{T}_q$ to be the set of n first order B-spline basis with knots at the covariate values.
- This results in the weights parameter $\boldsymbol{w}$ being jointly normal with precision matrix $H^T B^{-1} H$. The matrices H and B are $n \times n$ defined with $H_{ij} = [\langle \frac{d^2\phi_j(t)}{dt^2}, \phi_i(t) \rangle]$ and $B_{ij} = [\langle \phi_i, \phi_j \rangle]$.
- The matrices H and B equal to the matrices D and R in the ARIMA representation, except at the boundaries. They will be exactly equal if we remove ϕ_1 and ϕ_n from the set of test functions an reapply the finite element method.
- Lindgren and Rue (2008) then approximate the tri-diagonal matrix B with its diagonal approximation A, which distributes the off-diagonal values in B back to its main diagonal.

- Lindgren and Rue (2008) applied the finite element method above to approximate the SDE-based prior, setting both $\mathbb{B}_p$ and $\mathbb{T}_q$ to be the set of n first order B-spline basis with knots at the covariate values.
- This results in the weights parameter $\boldsymbol{w}$ being jointly normal with precision matrix $H^T B^{-1} H$. The matrices H and B are $n \times n$ defined with $H_{ij} = [\langle \frac{d^2 \phi_j(t)}{dt^2}, \phi_i(t) \rangle]$ and $B_{ij} = [\langle \phi_i, \phi_j \rangle]$.
- The matrices H and B equal to the matrices D and R in the ARIMA representation, except at the boundaries. They will be exactly equal if we remove ϕ_1 and ϕ_n from the set of test functions an reapply the finite element method.
- Lindgren and Rue (2008) then approximate the tri-diagonal matrix B with its diagonal approximation A, which distributes the off-diagonal values in B back to its main diagonal.

- Lindgren and Rue (2008) applied the finite element method above to approximate the SDE-based prior, setting both $\mathbb{B}_p$ and $\mathbb{T}_q$ to be the set of n first order B-spline basis with knots at the covariate values.
- This results in the weights parameter $\boldsymbol{w}$ being jointly normal with precision matrix $H^T B^{-1} H$. The matrices H and B are $n \times n$ defined with $H_{ij} = [\langle \frac{d^2\phi_j(t)}{dt^2}, \phi_i(t) \rangle]$ and $B_{ij} = [\langle \phi_i, \phi_j \rangle]$.
- The matrices H and B equal to the matrices D and R in the ARIMA representation, except at the boundaries. They will be exactly equal if we remove ϕ_1 and ϕ_n from the set of test functions an reapply the finite element method.
- Lindgren and Rue (2008) then approximate the tri-diagonal matrix B with its diagonal approximation A, which distributes the off-diagonal values in B back to its main diagonal.

- Lindgren and Rue (2008) applied the finite element method above to approximate the SDE-based prior, setting both $\mathbb{B}_p$ and $\mathbb{T}_q$ to be the set of n first order B-spline basis with knots at the covariate values.
- This results in the weights parameter $\boldsymbol{w}$ being jointly normal with precision matrix $H^T B^{-1} H$. The matrices H and B are $n \times n$ defined with $H_{ij} = [\langle \frac{d^2 \phi_j(t)}{dt^2}, \phi_i(t) \rangle]$ and $B_{ij} = [\langle \phi_i, \phi_j \rangle]$.
- The matrices H and B equal to the matrices D and R in the ARIMA representation, except at the boundaries. They will be exactly equal if we remove ϕ_1 and ϕ_n from the set of test functions an reapply the finite element method.
- Lindgren and Rue (2008) then approximate the tri-diagonal matrix B with its diagonal approximation A, which distributes the off-diagonal values in B back to its main diagonal.

- The resulting *approximated* finite element representation is called the extended RW2 model, which generalizes the traditional RW2 model defined for regularly spaced locations (Rue and Held, 2005).
- Also, because of the diagonal approximation it utilized, the precision matrix will be sparse and hence compatible with inference method such as INLA.

- The resulting *approximated* finite element representation is called the extended RW2 model, which generalizes the traditional RW2 model defined for regularly spaced locations (Rue and Held, 2005).
- Also, because of the diagonal approximation it utilized, the precision matrix will be sparse and hence compatible with inference method such as INLA.

We implemented and compared the two Bayesian smoothing methods using the following procedures:

- Re-parametrizing the smoothing parameter σ_s^2 as $\theta = -2\log\sigma_s$, and for each value of θ let Q_θ denotes the precision matrix corresponding to the evaluation vector $\boldsymbol{g}$.
- The conditional posterior $\pi(\boldsymbol{g}|\boldsymbol{y},\theta)$ then is approximated by its Gaussian approximation:

$$\tilde{\pi}_G(\boldsymbol{g}|\boldsymbol{y},\theta) \propto \exp\left\{-\frac{1}{2}\left(\boldsymbol{g}-\hat{\boldsymbol{g}}_\theta\right)^T H_\theta(\hat{\boldsymbol{g}}_\theta)\left(\boldsymbol{g}-\hat{\boldsymbol{g}}_\theta\right)\right\}, \quad (7)$$

the quantity $\hat{\boldsymbol{g}}_\theta$ denotes $\operatorname{argmax}_g \log\pi(\boldsymbol{g}|\theta,\boldsymbol{y})$ and $H_\theta(\boldsymbol{g})$ denotes $-\frac{d^2}{dgdg^T}\log\pi(\boldsymbol{g}|\theta,\boldsymbol{y})$.

We implemented and compared the two Bayesian smoothing methods using the following procedures:

- Re-parametrizing the smoothing parameter σ_s^2 as $\theta = -2\log \sigma_s$, and for each value of θ let Q_θ denotes the precision matrix corresponding to the evaluation vector $\boldsymbol{g}$.
- The conditional posterior $\pi(\boldsymbol{g}|\boldsymbol{y}, \theta)$ then is approximated by its Gaussian approximation:

$$\tilde{\pi}_G(\boldsymbol{g}|\boldsymbol{y}, \theta) \propto \exp\left\{-\frac{1}{2}\left(\boldsymbol{g}-\hat{\boldsymbol{g}}_\theta\right)^T H_\theta(\hat{\boldsymbol{g}}_\theta)\left(\boldsymbol{g}-\hat{\boldsymbol{g}}_\theta\right)\right\}, \qquad (7)$$

the quantity $\hat{\boldsymbol{g}}_\theta$ denotes $\operatorname{argmax}_g \log \pi(\boldsymbol{g}|\theta, \boldsymbol{y})$ and $H_\theta(\boldsymbol{g})$ denotes $-\frac{d^2}{dgdg^T} \log \pi(\boldsymbol{g}|\theta, \boldsymbol{y})$.

We implemented and compared the two Bayesian smoothing methods using the following procedures:

- Re-parametrizing the smoothing parameter σ_s^2 as $\theta = -2\log\sigma_s$, and for each value of θ let Q_θ denotes the precision matrix corresponding to the evaluation vector $\boldsymbol{g}$.
- The conditional posterior $\pi(\boldsymbol{g}|\boldsymbol{y},\theta)$ then is approximated by its Gaussian approximation:

$$\tilde{\pi}_G(\boldsymbol{g}|\boldsymbol{y},\theta) \propto \exp\left\{-\frac{1}{2}\left(\boldsymbol{g}-\hat{\boldsymbol{g}}_\theta\right)^T H_\theta(\hat{\boldsymbol{g}}_\theta)\left(\boldsymbol{g}-\hat{\boldsymbol{g}}_\theta\right)\right\}, \quad (7)$$

the quantity $\hat{\boldsymbol{g}}_\theta$ denotes $\text{argmax}_g \log\pi(\boldsymbol{g}|\theta,\boldsymbol{y})$ and $H_\theta(\boldsymbol{g})$ denotes $-\frac{d^2}{d\boldsymbol{g}d\boldsymbol{g}^T}\log\pi(\boldsymbol{g}|\theta,\boldsymbol{y})$.

- Then, we will follow the procedures as in Tierney and Kadane (1986), to obtain the Laplace approximation of the posterior of the smoothing parameter θ:

$$\tilde{\pi}_{\text{LA}}(\theta|\boldsymbol{y}) \propto \pi(\theta)\left\{\frac{|Q_\theta|}{|H_\theta(\hat{\boldsymbol{g}}_\theta)|}\right\}^{1/2} \exp\left\{-\frac{1}{2}\hat{\boldsymbol{g}}_\theta^T Q_\theta \hat{\boldsymbol{g}}_\theta + l(\boldsymbol{y};\hat{\boldsymbol{g}}_\theta)\right\}. \tag{8}$$

- For the posterior of $\boldsymbol{g}$, we will use the following approximation:

$$\tilde{\pi}(\boldsymbol{g}|\boldsymbol{y}) = \sum_{k=1}^{K} \tilde{\pi}_G(\boldsymbol{g}|\boldsymbol{y},\theta_k)\tilde{\pi}_{\text{LA}}(\theta_k|\boldsymbol{y})\delta_k, \tag{9}$$

where $\{\theta_k, \delta_k\}_{k=1}^K$ is a set of K nodes and weights selected using Adaptive Gauss-Hermite Quadrature (AGHQ) rule (Stringer, 2021).
- The computation of the AGHQ rule requires optimization of $\tilde{\pi}_{\text{LA}}(\theta|\boldsymbol{y})$, which will be done through the TMB package (Kristensen et al., 2015) with automatic differentiation.

- Then, we will follow the procedures as in Tierney and Kadane (1986), to obtain the Laplace approximation of the posterior of the smoothing parameter θ:

$$\tilde{\pi}_{\text{LA}}(\theta|\boldsymbol{y}) \propto \pi(\theta)\left\{\frac{|Q_\theta|}{|H_\theta(\hat{\boldsymbol{g}}_\theta)|}\right\}^{1/2} \exp\left\{-\frac{1}{2}\hat{\boldsymbol{g}}_\theta^T Q_\theta \hat{\boldsymbol{g}}_\theta + l(\boldsymbol{y};\hat{\boldsymbol{g}}_\theta)\right\}. \tag{8}$$

- For the posterior of $\boldsymbol{g}$, we will use the following approximation:

$$\tilde{\pi}(\boldsymbol{g}|\boldsymbol{y}) = \sum_{k=1}^{K} \tilde{\pi}_G(\boldsymbol{g}|\boldsymbol{y},\theta_k)\tilde{\pi}_{\text{LA}}(\theta_k|\boldsymbol{y})\delta_k, \tag{9}$$

where $\{\theta_k,\delta_k\}_{k=1}^K$ is a set of K nodes and weights selected using Adaptive Gauss-Hermite Quadrature (AGHQ) rule (Stringer, 2021).
- The computation of the AGHQ rule requires optimization of $\tilde{\pi}_{\text{LA}}(\theta|\boldsymbol{y})$, which will be done through the TMB package (Kristensen et al., 2015) with automatic differentiation.

- Then, we will follow the procedures as in Tierney and Kadane (1986), to obtain the Laplace approximation of the posterior of the smoothing parameter θ:

$$\tilde{\pi}_{\text{LA}}(\theta|\boldsymbol{y}) \propto \pi(\theta)\left\{\frac{|Q_\theta|}{|H_\theta(\hat{\boldsymbol{g}}_\theta)|}\right\}^{1/2} \exp\left\{-\frac{1}{2}\hat{\boldsymbol{g}}_\theta^T Q_\theta \hat{\boldsymbol{g}}_\theta + l(\boldsymbol{y};\hat{\boldsymbol{g}}_\theta)\right\}. \tag{8}$$

- For the posterior of $\boldsymbol{g}$, we will use the following approximation:

$$\tilde{\pi}(\boldsymbol{g}|\boldsymbol{y}) = \sum_{k=1}^{K} \tilde{\pi}_G(\boldsymbol{g}|\boldsymbol{y},\theta_k)\tilde{\pi}_{\text{LA}}(\theta_k|\boldsymbol{y})\delta_k, \tag{9}$$

where $\{\theta_k,\delta_k\}_{k=1}^K$ is a set of K nodes and weights selected using Adaptive Gauss-Hermite Quadrature (AGHQ) rule (Stringer, 2021).

- The computation of the AGHQ rule requires optimization of $\tilde{\pi}_{\text{LA}}(\theta|\boldsymbol{y})$, which will be done through the TMB package (Kristensen et al., 2015) with automatic differentiation.

- Then, we will follow the procedures as in Tierney and Kadane (1986), to obtain the Laplace approximation of the posterior of the smoothing parameter θ:

$$\tilde{\pi}_{\text{LA}}(\theta|\boldsymbol{y}) \propto \pi(\theta)\left\{\frac{|Q_\theta|}{|H_\theta(\hat{\boldsymbol{g}}_\theta)|}\right\}^{1/2} \exp\left\{-\frac{1}{2}\hat{\boldsymbol{g}}_\theta^T Q_\theta \hat{\boldsymbol{g}}_\theta + l(\boldsymbol{y};\hat{\boldsymbol{g}}_\theta)\right\}. \tag{8}$$

- For the posterior of $\boldsymbol{g}$, we will use the following approximation:

$$\tilde{\pi}(\boldsymbol{g}|\boldsymbol{y}) = \sum_{k=1}^{K} \tilde{\pi}_G(\boldsymbol{g}|\boldsymbol{y},\theta_k)\tilde{\pi}_{\text{LA}}(\theta_k|\boldsymbol{y})\delta_k, \tag{9}$$

  where $\{\theta_k,\delta_k\}_{k=1}^K$ is a set of K nodes and weights selected using Adaptive Gauss-Hermite Quadrature (AGHQ) rule (Stringer, 2021).
- The computation of the AGHQ rule requires optimization of $\tilde{\pi}_{\text{LA}}(\theta|\boldsymbol{y})$, which will be done through the TMB package (Kristensen et al., 2015) with automatic differentiation.

- Assume the following true data generating process:

$$y_i = g(x_i) + \epsilon_i,$$

where $\epsilon_i \sim N(0,3)$ and $g(x) = 5\sin(0.1x)$, observed at $x \in [0, 100]$.

- There are 10 equally spaced unique covariate values each with 10 repeated measurements.
- For both methods, we utilized the same penalized complexity prior (Simpson et al., 2017) for σ_s^2 and σ_ϵ^2, such that $P(\sigma_s > 2) = P(\sigma_\epsilon > 2) = 0.5$.
- We infer the values of g at a high resolution grid $\{z_i\}_{i=1}^{200}$ of equally spaced set of locations in $[0, 100]$ with spacing 0.5. We assume the function $g(.)$ can be well approximated by the step function $\tilde{g}(.) = \sum_{i=1}^{200} \mathbb{I}(z_i \leq . < z_{i+1}) g(z_i)$ where $z_{201} := +\infty$.

- Assume the following true data generating process:

$$y_i = g(x_i) + \epsilon_i,$$

where $\epsilon_i \sim N(0,3)$ and $g(x) = 5\sin(0.1x)$, observed at $x \in [0,100]$.

- There are 10 equally spaced unique covariate values each with 10 repeated measurements.
- For both methods, we utilized the same penalized complexity prior (Simpson et al., 2017) for σ_s^2 and σ_ϵ^2, such that $P(\sigma_s > 2) = P(\sigma_\epsilon > 2) = 0.5$.
- We infer the values of g at a high resolution grid $\{z_i\}_{i=1}^{200}$ of equally spaced set of locations in $[0,100]$ with spacing 0.5. We assume the function $g(.)$ can be well approximated by the step function $\tilde{g}(.) = \sum_{i=1}^{200} \mathbb{I}(z_i \leq . < z_{i+1}) g(z_i)$ where $z_{201} := +\infty$.

- Assume the following true data generating process:

$$y_i = g(x_i) + \epsilon_i,$$

where $\epsilon_i \sim N(0, 3)$ and $g(x) = 5\sin(0.1x)$, observed at $x \in [0, 100]$.

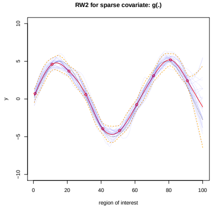
- There are 10 equally spaced unique covariate values each with 10 repeated measurements.
- For both methods, we utilized the same penalized complexity prior (Simpson et al., 2017) for σ_s^2 and σ_ϵ^2, such that $P(\sigma_s > 2) = P(\sigma_\epsilon > 2) = 0.5$.
- We infer the values of g at a high resolution grid $\{z_i\}_{i=1}^{200}$ of equally spaced set of locations in $[0, 100]$ with spacing 0.5. We assume the function $g(.)$ can be well approximated by the step function $\tilde{g}(.) = \sum_{i=1}^{200} \mathbb{I}(z_i \leq . < z_{i+1}) g(z_i)$ where $z_{201} := +\infty$.

- Assume the following true data generating process:

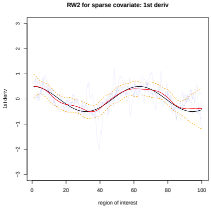
$$y_i = g(x_i) + \epsilon_i,$$

where $\epsilon_i \sim N(0, 3)$ and $g(x) = 5\sin(0.1x)$, observed at $x \in [0, 100]$.

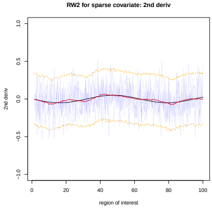
- There are 10 equally spaced unique covariate values each with 10 repeated measurements.
- For both methods, we utilized the same penalized complexity prior (Simpson et al., 2017) for σ_s^2 and σ_ϵ^2, such that $P(\sigma_s > 2) = P(\sigma_\epsilon > 2) = 0.5$.
- We infer the values of g at a high resolution grid $\{z_i\}_{i=1}^{200}$ of equally spaced set of locations in $[0, 100]$ with spacing 0.5. We assume the function $g(.)$ can be well approximated by the step function $\tilde{g}(.) = \sum_{i=1}^{200} \mathbb{I}(z_i \leq . < z_{i+1}) g(z_i)$ where $z_{201} := +\infty$.



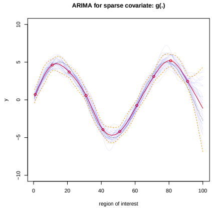
(a) g inferred using RW2



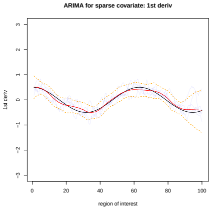
(b) g' inferred using RW2



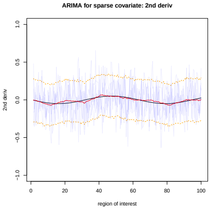
(c) g'' inferred using RW2



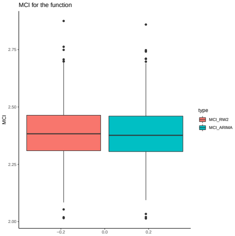
(d) g inferred using ARIMA



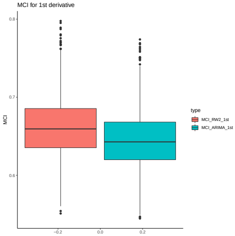
(e) g' inferred using ARIMA



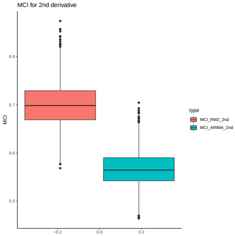
(f) g'' inferred using ARIMA



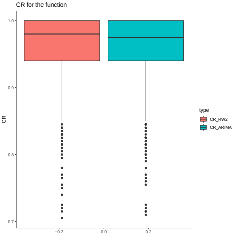
(g) MCI for g



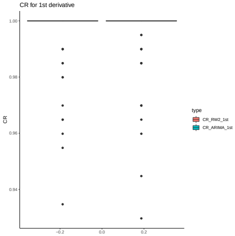
(h) MCI for g'



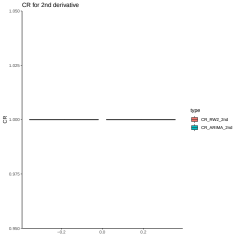
(i) MCI for g''



(j) CR for g



(k) CR for g'



(l) CR for g''

We illustrate the utility of the two Bayesian smoothing methods described above, using the atmospheric Carbon Dioxide (CO2) concentrations data from an observatory in Hawaii. This dataset contains the observation of CO2 concentrations from 1960 to 2021, with unequally spaced observation times.

- We consider the following model:

$$y_i = f_p(t_i) + f_{np}(t_i) + \epsilon_i,$$

where y_i denotes the observed CO2 concentration at year t_i, f_p and f_{np} are parametric effects and non-parametric random effects of time t.

- The parametric effect f_p is defined as:

$$f_p(t) = \beta_0 + \beta_1 \cos(2\pi t) + \beta_2 \sin(2\pi t) + \beta_3 \cos(4\pi t) + \beta_4 \sin(4\pi t),$$

which aims to capture the deterministic cycles of CO2 variation over time.

- The non-parametric effect function $f_{np}(t_i)$ will be inferred using the two Bayesian smoothing methods. For the priors, we use PC prior for the variance parameter σ_ϵ and the smoothing parameter σ_s, such that $P(\sigma_s > 0.01) = P(\sigma_\epsilon > 1) = 0.5$.
- The five fixed effects parameters $\beta_0, ..., \beta_4$ are given independent normal priors with zero mean and variance 10^6.

- We consider the following model:

$$y_i = f_p(t_i) + f_{np}(t_i) + \epsilon_i,$$

  where y_i denotes the observed CO2 concentration at year t_i, f_p and f_{np} are parametric effects and non-parametric random effects of time t.
- The parametric effect f_p is defined as:

$$f_p(t) = \beta_0 + \beta_1 \cos(2\pi t) + \beta_2 \sin(2\pi t) + \beta_3 \cos(4\pi t) + \beta_4 \sin(4\pi t),$$

  which aims to capture the deterministic cycles of CO2 variation over time.
- The non-parametric effect function $f_{np}(t_i)$ will be inferred using the two Bayesian smoothing methods. For the priors, we use PC prior for the variance parameter σ_ϵ and the smoothing parameter σ_s, such that $P(\sigma_s > 0.01) = P(\sigma_\epsilon > 1) = 0.5$.
- The five fixed effects parameters $\beta_0, ..., \beta_4$ are given independent normal priors with zero mean and variance 10^6.

- We consider the following model:

$$y_i = f_p(t_i) + f_{np}(t_i) + \epsilon_i,$$

where y_i denotes the observed CO2 concentration at year t_i, f_p and f_{np} are parametric effects and non-parametric random effects of time t.

- The parametric effect f_p is defined as:

$$f_p(t) = \beta_0 + \beta_1 \cos(2\pi t) + \beta_2 \sin(2\pi t) + \beta_3 \cos(4\pi t) + \beta_4 \sin(4\pi t),$$

which aims to capture the deterministic cycles of CO2 variation over time.

- The non-parametric effect function $f_{np}(t_i)$ will be inferred using the two Bayesian smoothing methods. For the priors, we use PC prior for the variance parameter σ_ϵ and the smoothing parameter σ_s, such that $P(\sigma_s > 0.01) = P(\sigma_\epsilon > 1) = 0.5$.
- The five fixed effects parameters $\beta_0, ..., \beta_4$ are given independent normal priors with zero mean and variance 10^6.

- We consider the following model:

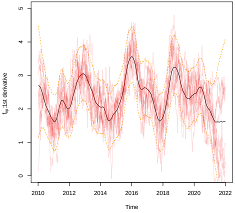
$$y_i = f_p(t_i) + f_{np}(t_i) + \epsilon_i,$$

  where y_i denotes the observed CO2 concentration at year t_i, f_p and f_{np} are parametric effects and non-parametric random effects of time t.
- The parametric effect f_p is defined as:

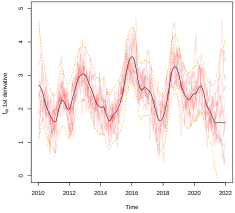
$$f_p(t) = \beta_0 + \beta_1 \cos(2\pi t) + \beta_2 \sin(2\pi t) + \beta_3 \cos(4\pi t) + \beta_4 \sin(4\pi t),$$

  which aims to capture the deterministic cycles of CO2 variation over time.
- The non-parametric effect function $f_{np}(t_i)$ will be inferred using the two Bayesian smoothing methods. For the priors, we use PC prior for the variance parameter σ_ϵ and the smoothing parameter σ_s, such that $P(\sigma_s > 0.01) = P(\sigma_\epsilon > 1) = 0.5$.
- The five fixed effects parameters $\beta_0, \ldots, \beta_4$ are given independent normal priors with zero mean and variance 10^6.

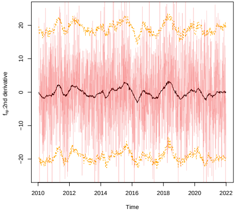
We want to consider both the quantity $f_{np}(t)$ and its first/second derivatives for $t \geq 2020$. To better infer the function f_{np}, we utilize a resolution grid with equal spacing being 1 week on the time domain, and predict the time domain from the observed years all the way to January 1st of year 2022.



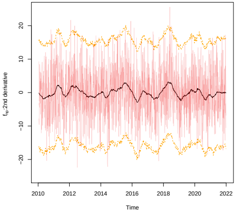
(a) f'_{np} using RW2



(b) f'_{np} using ARIMA



(c) f''_{np} using RW2



(d) f''_{np} using ARIMA

- We provide an overview of the extended second order random walk method (Lindgren and Rue, 2008), as well as its connection with the smoothing spline (Wahba, 1978) and the ARIMA prior (Brown and De Jong, 2001).
- The extended RW2 method gives similar result in terms of inference for g as the ARIMA method, but at a much smaller computational cost.
- Because of the diagonal approximation it used in the precision matrix, the method gives less smooth inference for higher order derivatives of g compared to ARIMA method.
- We illustrate that It is possible to implement the exact ARIMA method without diagonal approximation. But which method is better should depend on the question of interest.

- We provide an overview of the extended second order random walk method (Lindgren and Rue, 2008), as well as its connection with the smoothing spline (Wahba, 1978) and the ARIMA prior (Brown and De Jong, 2001).
- The extended RW2 method gives similar result in terms of inference for g as the ARIMA method, but at a much smaller computational cost.
- Because of the diagonal approximation it used in the precision matrix, the method gives less smooth inference for higher order derivatives of g compared to ARIMA method.
- We illustrate that It is possible to implement the exact ARIMA method without diagonal approximation. But which method is better should depend on the question of interest.

- We provide an overview of the extended second order random walk method (Lindgren and Rue, 2008), as well as its connection with the smoothing spline (Wahba, 1978) and the ARIMA prior (Brown and De Jong, 2001).
- The extended RW2 method gives similar result in terms of inference for g as the ARIMA method, but at a much smaller computational cost.
- Because of the diagonal approximation it used in the precision matrix, the method gives less smooth inference for higher order derivatives of g compared to ARIMA method.
- We illustrate that It is possible to implement the exact ARIMA method without diagonal approximation. But which method is better should depend on the question of interest.

- We provide an overview of the extended second order random walk method (Lindgren and Rue, 2008), as well as its connection with the smoothing spline (Wahba, 1978) and the ARIMA prior (Brown and De Jong, 2001).
- The extended RW2 method gives similar result in terms of inference for g as the ARIMA method, but at a much smaller computational cost.
- Because of the diagonal approximation it used in the precision matrix, the method gives less smooth inference for higher order derivatives of g compared to ARIMA method.
- We illustrate that It is possible to implement the exact ARIMA method without diagonal approximation. But which method is better should depend on the question of interest.

Brown, P. and P. De Jong (2001, March). Nonparametric smoothing using state space techniques. *Canadian Journal of Statistics 29*(1), 37–50.

Kristensen, K., A. Nielsen, C. W. Berg, H. Skaug, and B. Bell (2015). Tmb: automatic differentiation and laplace approximation. *arXiv preprint arXiv:1509.00660*.

Lindgren, F. and H. Rue (2008). On the second-order random walk model for irregular locations. *Scandinavian Journal of Statistics 35*(4), 691–700.

Rue, H. and L. Held (2005). *Gaussian Markov random fields: theory and applications*. CRC press.

Rue, H., S. Martino, and N. Chopin (2009). Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 71*(2), 319–392.

Simpson, D., H. Rue, A. Riebler, T. G. Martins, and S. H. Sørbye (2017). Penalising model component complexity: A principled, practical approach to constructing priors. *Statistical science 32*(1), 1–28.

Stringer, A. (2021). Implementing approximate bayesian inference using adaptive quadrature: the aghq package.

Tierney, L. and J. B. Kadane (1986). Accurate approximations for posterior moments and marginal densities. *Journal of the american statistical association 81*(393), 82–86.

Wahba, G. (1978). Improper priors, spline smoothing and the problem of guarding against model errors in regression. *Journal of the Royal Statistical Society. Series B (Methodological) 40*(3), 364–372.