To-do list

1. Section 4.X 2
2. Section X 6
3. Running example: smoothing in one dimension. Run three methods of estimation, compare parameter estimates, MSE of prediction. Runtime? Highlight difficulties of MCMC. 8
4. Section X 8

Contents

4 Regression modelling using I-priors **1**

4.1 Various regression models 1
4.2 Estimation 1
4.2.1 The intercept and the prior mean 3
4.2.2 Direct optimisation 4
4.2.3 Expectation-maximisation algorithm 6
4.2.4 Markov chain Monte Carlo methods 7
4.2.5 Comparison of estimation methods 8
4.3 Computational considerations 8
4.4 Post-estimation 8
4.5 Examples 12
4.6 Conclusion 12
4.7 Deriving the posterior distribution for w 13
4.8 A recap on the exponential family EM algorithm 15
4.9 Deriving the posterior predictive distribution 17

Bibliography **20**

Haziq Jamil
Department of Statistics
London School of Economics and Political Science
March 20, 2018

Chapter 4

Regression modelling using I-priors

In the previous chapter, we defined an I-prior for the normal regression model (1.1) subject to (1.2) and f belonging to a reproducing kernel Hilbert or Krein space of functions. We also saw how new function spaces can be constructed via the polynomial and ANOVA RKKS. In this chapter, we shall describe various regression models, and connect them to an appropriate RKKS, so that an I-prior may be defined on it. Methods for estimating I-prior models will also be described. Finally, several examples of I-prior modelling are presented.

4.1 Various regression models

4.2 Estimation

Depending on the type of regression model, an appropriate function space $\mathcal{F}$ needs to be chosen, and the reproducing kernel for the function space identified. Using an I-prior on the regression function $f \in \mathcal{F}$ with prior mean $f_0 \in \mathcal{F}$, the interest is then to obtain a posterior estimate of f. We denote the dependence of the kernel h on the parameters η

by h_η. The regression model to be estimated is always of the form

$$\begin{gathered} y_i = \alpha + f_0(x_i) + \overbrace{\sum_{k=1}^n h_\eta(x_i, x_k) w_k}^{f(x_i)} + \epsilon_i \\ (\epsilon_1, \dots, \epsilon_n)^\top \sim \mathrm{N}_n(\mathbf{0}, \mathbf{\Psi}^{-1}) \\ (w_1, \dots, w_n)^\top \sim \mathrm{N}_n(\mathbf{0}, \mathbf{\Psi}), \end{gathered} \tag{4.1}$$

although the indices may need to be adjusted for the individual model at hand, especially when dealing with ANOVA RKKSs. The parameters of the I-prior model are collectively denoted by $\theta = \{\alpha, \eta, \mathbf{\Psi}\}$. Given θ and a prior choice for f_0, the posterior regression function is determined solely by the posterior distribution of the w_i's. Under all of these normality assumptions, using standard multivariate normal results one finds that the posterior of $\mathbf{w} := (w_1, \dots, w_n)^\top$ is $\mathbf{w}|\mathbf{y} \sim \mathrm{N}_n(\tilde{\mathbf{w}}, \tilde{\mathbf{V}}_w)$, where

$$\tilde{\mathbf{w}} = \mathbf{\Psi}\mathbf{H}_\eta\mathbf{V}_y^{-1}(\mathbf{y} - \alpha\mathbf{1}_n - \mathbf{f}_0) \quad \text{and} \quad \tilde{\mathbf{V}}_w = \big(\mathbf{H}_\eta\mathbf{\Psi}\mathbf{H}_\eta + \mathbf{\Psi}^{-1}\big)^{-1} = \mathbf{V}_y^{-1}, \tag{4.2}$$

where $\mathbf{f}_0 = \big(f_0(x_1), \dots, f_0(x_n)\big)^\top$, $\mathbf{H}_\eta$ is the kernel matrix with (i,j) entries equal to $h_\eta(x_i, x_j)$, and $\mathbf{V}_y$ is the variance of the marginal distribution for $\mathbf{y} = (y_1, \dots, y_n)$. See Appendix 4.7 for a derivation.

In each modelling scenario, there are a number of kernel parameters η that need to be estimated from the data. Assuming that the covariate space is $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_p$, and there is an ANOVA like decomposition of the function space $\mathcal{F}$ into its constituents spaces $\mathcal{F}_1, \dots, \mathcal{F}_p$, then at the very least, there are p scale parameters $\lambda_1, \dots, \lambda_p$ for each of the RKHSs. Depending on the RKHS used, there could be either kernel parameters that need to be optimised—the Hurst index for the fBm RKHS, the lengthscale for the SE RKHS, and the offset for the polynomial RKKS. Default settings for these parameters may be used, and if this is the case, only scale parameters need to be estimated, and the estimation procedure can be made more efficient as the kernel matrices need not be recomputed each time. This is explained in further detail in Section 4.X.

For simplicity, the following additional assumptions are imposed on the I-prior model (4.1):

A1. Set $\alpha = 0$ and replace the responses by their centred versions $y_i \mapsto \tilde{y}_i = y_i - \frac{1}{n}\sum_{i=1}^n$.

A2. Assume a zero prior mean $f_0(x) = 0$ for all $x \in \mathcal{X}$.

A3. Assume identical and independent errors, $\mathbf{\Psi} = \psi \mathbf{I}_n$.

Assumptions A1 and A2 are motivated by the discussion in Section 4.2.1. Although assumption A3 is not strictly necessary, it is often a reasonable one and one that simplifies the estimation procedure greatly.

The following subsections describe possible estimation procedures for the hyperparameters of the model. Implementation of these estimation procedures are done in R, mainly using the **iprior** package (Jamil and Bergsma, 2017).

4.2.1 The intercept and the prior mean

In most statistical models, an intercept is a necessary inclusion to aid interpretation. In the regression model stated in (1.1), a lack of an intercept would fail to account for the correct location of the regression function with respect to the y-axis. Further, when zero-mean functions are considered, the intercept serves as being the 'grand mean' value of the responses.

There are two ways of including an intercept in the I-prior model. One is by including the tensor sum of the RKHS of constant functions to $\mathcal{F}$, and the other is to simply treat the intercept as a parameter of the model to be estimated. In the polynomial and ANOVA RKKSs, we saw that an intercept is naturally induced by the inclusion of a RKHS of constant functions in their construction. In any of the other RKHSs described in Chapter 2, an intercept would need to be added separately.

These two methods convey the same mathematical model, and there is very little difference in the way of interpretation, although estimation is completely different. In the former method, the intercept-less RKHS/RKKS $\mathcal{F}$ with kernel h is made to include an intercept by modifying the kernel to be $h + 1$. The intercept will then be implicitly taken care of without having dealt with it explicitly. However, it can be obtained by realising that for $\alpha \in \mathcal{F}_0$ the RKHS of constant functions, then $\alpha = \sum_{i=1}^n w_i$.

On the other hand, consider the intercept as a parameter α to be estimated. Obtaining an estimate α using a likelihood-based argument is rather simple. From (4.1), $\mathrm{E}\, y_i = \alpha + f_0(x_i)$ for all $i = 1, \dots, n$, so the maximum likelihood estimate for $\mathrm{E}\, y$ is its sample mean $\bar{y} = \frac{1}{n}\sum_{i=1} y_i$, and hence the ML estimate for α is $\hat{\alpha} = \bar{y} - \frac{1}{n}\sum_{i=1}^n f_0(x_i)$. Alternatively, the estimation of α under a fully Bayesian treatment is possible by assuming an appropriate hyperprior on it, such as a conjugate normal prior $\mathrm{N}(a, A^{-1})$. If so, the conditional posterior of α given $\mathbf{w}$, θ and f_0 is also normal with mean $\tilde{a}$ and variance $\tilde{A}$, where

$$\tilde{A} = \sum_{i,j=1}^{n} \psi_{ij} + A \quad \text{and} \quad \tilde{a} = \tilde{A}^{-1} \left(\sum_{i=1}^{n} [(\mathbf{y} - \mathbf{f}_0 - \mathbf{H}_\eta \mathbf{w})\mathbf{\Psi}]_i + Aa \right).$$

This fact can be used, say, in conjunction with a Gibbs sampling procedure treating the rest of the unknowns as random. Note that the posterior mean for α is

$$\mathrm{E}[\alpha|\mathbf{y}] = \mathrm{E}_{\mathbf{w}} \big[\mathrm{E}[\alpha|\mathbf{y}, \mathbf{w}] \big] = \frac{\sum_{i,j=1}^{n} \psi_{ij}(y_i - f_0(x_i)) + Aa}{\sum_{i,j=1}^{n} \psi_{ij} + A},$$

which, in the iid errors case, is seen to be a weighted sum of the ML estimate $\hat{\alpha}$ and the prior mean a. Unless there is a strong reason to add prior information to the intercept, the ML estimate seems to be the simplest approach.

Now, a note on the prior mean f_0. For kernels with the property that $h(x, x^*) \to 0$ as $D(x, x^*) \to \infty$ for $x \in \mathcal{X}_{\text{train}}$ and $x^* \in \mathcal{X}_{\text{new}}$ such as the SE kernel, this means that predictions outside the training set will be zero and thus rely on the prior mean f_0. However, all of the other kernels in this thesis, namely the fBm, canonical, and polynomial kernels, do not have this property—they instead use information provided by the training data to extrapolate predictions far away from the data set. A prior mean of zero seems reasonable and safe in the absence of any prior information, so long as the global and local properties of the regression function are understood with respect to the kernel chosen. $f_0 = 0$ also implies a complete reliance on the data rather than subjective prior belief of a suitable choice for f.

Of course, should it be felt appropriate, a non-zero function f_0 may be imposed as the prior mean. If $f_0(x) = \mu_0 \in \mathbb{R}$ for all $x \in \mathcal{X}$, then this basically implies another intercept in the model, if it is not already present. Note that when treating μ_0 as a hyperparameter to be estimated, then this does not yield a fully identified model, and only $\alpha + \mu_0$ may be estimated.

4.2.2 Direct optimisation

Assuming A1 and A2, a direct optimisation of the parameters $\theta = \{\eta, \mathbf{\Psi}\}$ using the log-likelihood of θ is straightforward to implement. Denote $\mathbf{\Sigma}_\theta := \mathbf{H}_\eta \mathbf{\Psi} \mathbf{H}_\eta + \mathbf{\Psi}^{-1} = \mathbf{V}_y$.

From (4.1), the log-likelihood of θ is given by

$$\begin{aligned} L(\theta) &= \log \int p(\mathbf{y}|\mathbf{w})p(\mathbf{w})\,\mathrm{d}\mathbf{w} \\ &= -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{\Sigma}_\theta| - \frac{1}{2}\tilde{\mathbf{y}}^\top\mathbf{\Sigma}_\theta^{-1}\tilde{\mathbf{y}} \end{aligned} \tag{4.3}$$

This is typically done using conjugate gradients with a Cholesky decomposition on the covariance kernel to maintain stability, but the **iprior** package opts for an eigendecomposition of the kernel matrix $\mathbf{H}_\eta = \mathbf{V} \cdot \mathrm{diag}(u_1, \dots, u_n) \cdot \mathbf{V}^\top$ instead. Further, assuming A3 and since $\mathbf{H}_\eta$ is a symmetrix matrix, we have that $\mathbf{V}\mathbf{V}^\top = \mathbf{I}_n$, and thus

$$\mathbf{V}_y = \mathbf{V} \cdot \mathrm{diag}(\psi u_1^2 + \psi^{-1}, \dots, \psi u_n^2 + \psi^{-1}) \cdot \mathbf{V}^\top$$

for which the inverse and log-determinant is easily obtainable. This method is relatively robust to numerical instabilities and is better at ensuring positive definiteness of the covariance kernel. The eigendecomposition is performed using the **Eigen** C++ template library and linked to **iprior** using **Rcpp** (Eddelbuettel and Francois, 2011). The hyperparameters are transformed by the **iprior** package so that an unrestricted optimisation using the quasi-Newton L-BFGS algorithm provided by `optim()` in R. Note that minimisation is done on the deviance scale, i.e., minus twice the log-likelihood. The direct optimisation method can be prone to local optima, in which case repeating the optimisation at different starting points and choosing the one which yields the highest likelihood is one way around this.

Let $\mathbf{U}$ be the Fisher information matrix for $\theta \in \mathbb{R}^q$. Standard calculations show that under the marginal distribution $\tilde{\mathbf{y}} \sim \mathrm{N}_n\big(\mathbf{0}, \mathbf{\Sigma}_\theta\big)$, the (i,j)th coordinate of $\mathbf{U}$ is

$$u_{ij} = \frac{1}{2}\,\mathrm{tr}\left(\mathbf{\Sigma}_\theta^{-1}\frac{\partial\mathbf{\Sigma}_\theta}{\partial\theta_i}\mathbf{\Sigma}_\theta^{-1}\frac{\partial\mathbf{\Sigma}_\theta}{\partial\theta_j}\right)$$

where the derivative of a matrix with respect to a scalar is the element-wise derivative of the matrix. With $\hat{\theta}$ denoting the ML estimate for θ, under suitable conditions, $\sqrt{n}(\hat{\theta} - \theta)$ has an asymptotic multivariate normal distribution with mean zero and covariance matrix $\mathbf{U}^{-1}$. In particular, the standard errors for θ_k are the diagonal elements of $\mathbf{U}^{-1/2}$.

4.2.3 Expectation-maximisation algorithm

We describe an expectation-maximisation algorithm to estimate both the posterior regression function and the hyperparameters of (4.1) simultaneously. Assume A1 and A2 applies. Evidently, (4.1) lends itself to resembling a random-effects model. By treating the complete data as $\{\mathbf{y}, \mathbf{w}\}$ and the w_i's as "missing", the tth iteration of the E-step entails computing

$$\begin{aligned} Q(\theta) &= \mathrm{E}_{\mathbf{w}} \left[\log p(\mathbf{y}, \mathbf{w}|\theta) \,\middle|\, \mathbf{y}, \theta^{(t)} \right] \\ &= \mathrm{E}_{\mathbf{w}} \left[\text{const.} - \frac{1}{2}(\tilde{\mathbf{y}} - \mathbf{H}_\eta \mathbf{w})^\top \mathbf{\Psi} (\tilde{\mathbf{y}} - \mathbf{H}_\eta \mathbf{w}) - \frac{1}{2}\mathbf{w}^\top \mathbf{\Psi}^{-1} \mathbf{w} \,\middle|\, \mathbf{y}, \theta^{(t)} \right] \\ &= \text{const.} - \frac{1}{2}\tilde{\mathbf{y}}^\top \mathbf{\Psi} \tilde{\mathbf{y}} - \frac{1}{2} \operatorname{tr}\left(\overbrace{(\mathbf{H}_\eta \mathbf{\Psi} \mathbf{H}_\eta + \mathbf{\Psi}^{-1})}^{\mathbf{\Sigma}_\theta} \tilde{\mathbf{W}}^{(t)} \right) + \tilde{\mathbf{y}}^\top \mathbf{\Psi} \mathbf{H}_\eta \tilde{\mathbf{w}}^{(t)}, \end{aligned} \tag{4.4}$$

where $\tilde{\mathbf{w}}^{(t)} = \mathrm{E}[\mathbf{w}|\mathbf{y}, \theta^{(t)}]$ and $\tilde{\mathbf{W}}^{(t)} = \mathrm{E}[\mathbf{w}\mathbf{w}^\top|\mathbf{y}, \theta^{(t)}]$ are the first and second posterior moments of $\mathbf{w}$ calculated at the tth EM iteration. These can be computed directly from (4.2), substituting $\theta^{(t)}$ as appropriate. Note that (4.4) follows as a direct consequence of the results in Appendix 4.7.

Assume that A3 applies. The M-step then assigns $\theta^{(t+1)}$ the value of θ which maximises the Q function above. This boils down to solving the first order conditions

$$\frac{\partial Q}{\partial \eta} = -\frac{1}{2} \operatorname{tr}\left(\frac{\partial \mathbf{\Sigma}_\theta}{\partial \eta} \tilde{\mathbf{W}}^{(t)} \right) + \psi \cdot \tilde{\mathbf{y}}^\top \frac{\partial \mathbf{H}_\eta}{\partial \eta} \tilde{\mathbf{w}}^{(t)} \tag{4.5}$$

$$\frac{\partial Q}{\partial \psi} = -\frac{1}{2}\tilde{\mathbf{y}}^\top \tilde{\mathbf{y}} - \operatorname{tr}\left(\frac{\partial \mathbf{\Sigma}_\theta}{\partial \psi} \tilde{\mathbf{W}}^{(t)} \right) + \tilde{\mathbf{y}}^\top \mathbf{H}_\eta \tilde{\mathbf{w}}^{(t)} \tag{4.6}$$

equated to zero. As $\partial \mathbf{\Sigma}_\theta / \partial \psi = \mathbf{H}_\eta^2 - \psi^{-2}$, the solution to (4.6) is obtained as

$$\psi^{(t+1)} = \left\{ \frac{\operatorname{tr} \tilde{\mathbf{W}}^{(t)}}{\tilde{\mathbf{y}}^\top \tilde{\mathbf{y}} + \operatorname{tr}(\mathbf{H}_\eta^2 \tilde{\mathbf{W}}^{(t)}) - 2\tilde{\mathbf{y}}^\top \mathbf{H}_\eta \tilde{\mathbf{w}}^{(t)}} \right\}^{1/2}. \tag{4.7}$$

The solution to (4.5) can also be found in closed-form, but only in cases where the full-data likelihood in η emits itself as belonging to an exponential family likelihood. Such cases are described in further detail in Section X. In cases where closed-form solutions do exist for η, then it is just a matter of iterating the update equations until a suitable convergence criterion is met (e.g. no more sizeable increase in successive log-likelihood values). In cases where closed-form solutions do not exist for η, the Q function is again optimised with respect to η using the L-BFGS algorithm.

In our experience, EM algorithm is more stable than direct maximisation, in that it is less prone to the likelihood exploding should any of the parameters approach problematic boundary values. As such, the EM is especially suitable if there are many scale parameters to estimate. On the flip side, it is typically slow to converge. The **iprior** package provides a method to automatically switch to the direct optimisation method after running several EM iterations. This then combines the stability of the EM with the speed of direct optimisation. Section X also describes various strategies to run the EM algorithm efficiently.

As a final remark, it is well known that the EM algorithm increases the value of the log-likelihood at each iteration. It is also known that the EM sequence $\theta^{(t)}$ eventually convergences to some θ^*, but the fact that θ^* is the ML estimate is not guaranteed. The paper by Wu (1983) details the conditions necessary for the EM to produce ML estimates. If the EM tends to get stuck in some local maxima of the likelihood, then a general strategy is to restart the EM from multiple starting values.

4.2.4 Markov chain Monte Carlo methods

For completeness, it should be mentioned that a full Bayesian treatment of the model is possible, with additional priors on the set of hyperparameters. Markov chain Monte Carlo (MCMC) methods can then be employed to sample from the posteriors of the hyperparameters, with point estimates obtained using the posterior mean or mode, for instance. Additionally, the posterior distribution encapsulates the uncertainty about the parameter, for which inference can be made. Posterior sampling can be done using Gibbs-based methods in **WinBUGS** (Lunn et al., 2000) or **JAGS** (Plummer, 2003), and both have interfaces to R via **R2WinBUGS** (Sturtz et al., 2005) and **runjags** (Denwood, 2016) respectively. Hamiltonian Monte Carlo (HMC) sampling is also a possibility, and the Stan project (Carpenter et al., 2017) together with the package **rstan** (Stan Development Team, 2016) makes this possible in R. All of these MCMC packages require the user to code the model individually, and we are not aware of the existence of MCMC-based packages which are able to estimate GPR models. This makes it inconvenient for GPR and I-prior models, because in addition to the model itself, the kernel functions need to be coded as well and ensuring computational efficiency would be a difficult task. Note that this full Bayesian method is not implemented in **iprior**, but described here for completeness.

4.2.5 Comparison of estimation methods

Running example: smoothing in one dimension. Run three methods of estimation, compare parameter estimates, MSE of prediction. Runtime? Highlight difficulties of MCMC.

4.3 Computational considerations

4.4 Post-estimation

One of the perks of a (semi-)Bayesian approach to regression modelling is that we are able to use Bayesian post-estimation machinery involving the relevant posterior distributions. With the normal I-prior model, there is the added benefit that posterior distributions are easily obtained in closed form. The plots that are shown in this subsection is a continuation from the example in Section X.

Recall that for the regression function as specified in (4.1), its posterior regression function is found to be $f(x) = \sum_{i=1}^n h_{\hat\eta}(x, x_i)\tilde w_i$, where $\hat\eta$ is the ML estimate for the kernel parameters, and the $\tilde w_i$'s are multivariate-normally distributed with mean and variance according to (4.2). Denote by $\mathbf{h}_{\hat\eta}(x)$ the n-vector with entries equal to $h_{\hat\eta}(x, x_i)$. Therefore, the posterior density for the regression function is

$$p\big(f(x)|\mathbf{y}\big) \sim \mathrm{N}\left(\mathbf{h}_{\hat\eta}(x)\hat{\mathbf{w}}, \mathbf{h}_{\hat\eta}(x)^\top\big(\mathbf{H}_{\hat\eta}\hat{\mathbf{\Psi}}\mathbf{H}_{\hat\eta} + \hat{\mathbf{\Psi}}^{-1}\big)^{-1}\mathbf{h}_{\hat\eta}(x)\right) \tag{4.8}$$

for any x in the domain of the regression function. Here, the hats on the parameters indicate the use of the optimised model parameters, i.e. the ML or MAP estimates.

Prediction of a new data point is also of interest. A priori, assume that $y_{\text{new}} = \hat\alpha + f(x_{\text{new}}) + \epsilon_{\text{new}}$, where $\epsilon_{\text{new}} \sim \mathrm{N}(0, \psi_{\text{new}}^{-1})$, and $f \sim$ I-prior. Denote the covariance between ϵ_{new} and $\boldsymbol\epsilon = (\epsilon_1, \dots, \epsilon_n)^\top$ by $\boldsymbol\sigma_{\text{new}}^\top \in \mathbb{R}^n$. Under an iid model (assumption A3), then $\psi_{\text{new}} = \psi = \operatorname{Var}\epsilon_i$ for any $i \in \{1, \dots, n\}$, and $\boldsymbol\sigma_{\text{new}}^\top = \mathbf{0}$, but otherwise, these extra parameters need to be dealt with somehow, either by specifying them a priori or estimating them again, which seems excessive. In any case, using a linearity argument, figure/04-post_reg_prior_post.pdf

Figure 4.1: Prior (top) and posterior (bottom) sample path realisations of regression functions drawn from their respective distributions when $\mathcal{F}$ is a fBm-0.5 RKHS. At the very top of the figure, a smoothed density estimate of the x's is overlaid. In regions with few data points (near the centre), there is little Fisher information, and hence a conservative prior closer to zero, the prior mean, for this region.

the posterior distribution for y_{new} is normal, with mean and variance given by

$$\mathrm{E}[y_{\text{new}}|\mathbf{y}] = \hat{\alpha} + \mathrm{E}\left[f(x_{\text{new}})|\mathbf{y}\right] + \text{mean correction term} \tag{4.9}$$

and

$$\mathrm{Var}[y_{\text{new}}|\mathbf{y}] = \mathrm{Var}\left[f(x_{\text{new}})|\mathbf{y}\right] + \psi_{\text{new}}^{-1} + \text{variance correction term.} \tag{4.10}$$

A derivation is presented in Appendix 4.9. Note that the mean and variance correction term vanishes under an iid assumption A3. The posterior distribution for y_{new} can be used in several ways. Among them, is to construct a $100(1-\alpha/2)\%$ credibility interval for the (mean) predicted value y_{new} using

$$\mathrm{E}[y_{\text{new}}|\mathbf{y}] \pm \Phi^{-1}(1-\alpha/2) \cdot \mathrm{Var}[y_{\text{new}}|\mathbf{y}]^{\frac{1}{2}},$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function. One could also perform a posterior predictive density check of the data $\mathbf{y}$, by repeatedly sampling n points from its posterior distribution. This provides a visual check of whether there are any systematic deviances between what the model predicts, and what is observed from the data.

Lastly, we discuss model comparison. Recall that the marginal distribution for $\mathbf{y}$ after integrating out the I-prior for f in model (4.1) is a normal distribution. Suppose that we are interested in comparing two candidate models M_1 and M_2, each with the parameter set θ_1 and θ_2. Commonly, we would like to test whether or not particular terms in the ANOVA RKKS are significant contributors in explaining the relationship between the responses and predictors. A log-likelihood comparison is possible using an asymptotic chi-squared distribution, with degrees of freedom equal to the difference between the number of parameters in θ_2 and θ_1. This is assuming model M_1 is nested within M_2, which is the case for ANOVA-type constructions. Note that if two models have the same number of parameters, then the model with the higher likelihood is preferred.

As a remark, this method of comparing marginal likelihoods can be seen as Bayesian model selection using *empirical Bayes factors*, where the Bayes factor of comparing model M_1 to model M_2 is defined as

$$\mathrm{BF}(M_1, M_2) = \frac{\int p(\mathbf{y}|\theta_1, \mathbf{f}) p(\mathbf{f}) \, \mathrm{d}\mathbf{f}}{\int p(\mathbf{y}|\theta_2, \mathbf{f}) p(\mathbf{f}) \, \mathrm{d}\mathbf{f}}$$

The word 'empirical' stems from the fact that the parameters are estimated via an figure/04-post_reg_cred.pdf

Figure 4.2: The fitted regression line in red is the posterior mean estimate of the regression function (shifted by the intercept), which also gives the posterior mean estimate for the responses y. The shaded region is the 95% credibility interval for predictions using this regression function.

figure/04-post_reg_ppc.pdf

Figure 4.3: Posterior predictive density checks of the responses: repeated sampling from the posterior density of the y_i's and plotting their densities allows us to compare model predictions against observed samples.

empirical Bayes approach (maximum marginal likelihood). This approach is fine when the number of comparisons to be made is small, but can be computationally unfeasible when many marginal likelihoods need to be pairwise compared. In Chapter 6, we explore a fully Bayesian approach to explore the entire model space for the special case of linear models.

4.5 Examples

4.6 Conclusion

The steps for I-prior modelling are basically three-fold:

1. Select an appropriate function space; equivalently, the kernels for which a specific effect is desired on the covariates. Several modelling examples are described in Section 4.1.
2. Estimate the hyperparameters (these included the RKHS scale parameter(s), error precision, and any other kernel parameters such as the Hurst index of fBm) of the I-prior model and obtain the posterior regression function.
3. Post-estimation procedures include
   - Posterior predictive checks;
   - Model comparison via log-likelihood ratio tests/empirical Bayes factors; and
   - Prediction of new data point.

The main sticking point with the estimation procedure is the involvement of the $n \times n$ kernel matrix, for which its inverse is needed. This requires $O(n^2)$ storage and $O(n^3)$ computational time. The Nyström method of approximating the kernel matrix reduces complexity to $O(nm)$ storage and approximately $O(nm^2)$, and is highly advantageous if $m \ll n$. The computational issue faced by I-priors are mirrored in Gaussian process regression, so the methods to overcome these computational challenges in GPR can be explored further. However, most efficient computational solutions exploit the nature of the SE kernel structure, which is the most common kernel used in GPR.

Several avenues have been discussed to make the estimation procedure more efficient, but improvements can be had. One promising avenue to achieve efficient estimation for I-prior models is by using variational methods. A sparse variational approximation (typically by using inducing points) or stochastic variational inference can greatly reduce computational storage and speed requirements. A recent paper by Cheng and Boots (2017) suggested a variational algorithm with linear complexity for GPR-type models.

On the topic of accelerating the EM algorithm, besides the MOEM procedure, there are two other algorithms that could be explored. The first is called parameter-expansion EM algorithm (PXEM) by (Liu et al., 1998), which has been shown to be promising for random-effects type models. It involves correcting the M-step by a 'covariance adjustment', so that extra information can be capitalised on to improve convergence rates. The second is a quasi-Newton acceleration of the EM algorithm as proposed by Lange (1995). A slight change to the EM gradient algorithm in the M-step steers the EM algorithm to the Newton-Raphson algorithm, thus exploiting the benefits of the EM algorithm in the early stages (monotonic increase in likelihood) and avoiding the pitfalls of Newton-Raphson (getting stuck in local optima). The PXEM and quasi-Newton EM algorithms require an in-depth reassessment of the EM algorithm to specifically tailor them to I-prior models, which we leave as future work.

Miscellanea

Appendix

4.7 Deriving the posterior distribution for w

In the following derivation, we implicitly assume the dependence on $\mathbf{f}_0$ and θ. The distribution of $\mathbf{y}|\mathbf{w}$ is $\mathrm{N}_n(\boldsymbol{\alpha}+\mathbf{f}_0+\mathbf{H}_\eta\mathbf{w}, \boldsymbol{\Psi}^{-1})$, where $\boldsymbol{\alpha} = \alpha\mathbf{1}_n$, while the prior distribution for $\mathbf{w}$ is $\mathrm{N}_n(\mathbf{0}, \boldsymbol{\Psi})$. Since $p(\mathbf{w}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{w})p(\mathbf{w})$, we have that

$$
\begin{aligned}
\log p(\mathbf{w}|\mathbf{y}) &= \log p(\mathbf{y}|\mathbf{w}) + \log p(\mathbf{w}) \\
&= \text{const.} + \cancel{\frac{1}{2}\log|\boldsymbol{\Psi}|} - \frac{1}{2}(\mathbf{y} - \boldsymbol{\alpha} - \mathbf{f}_0 - \mathbf{H}_\eta\mathbf{w})^\top \boldsymbol{\Psi}(\mathbf{y} - \boldsymbol{\alpha} - \mathbf{f}_0 - \mathbf{H}_\eta\mathbf{w}) \\
&\quad - \cancel{\frac{1}{2}\log|\boldsymbol{\Psi}|} - \frac{1}{2}\mathbf{w}^\top\boldsymbol{\Psi}^{-1}\mathbf{w} \\
&= \text{const.} - \frac{1}{2}\mathbf{w}^\top(\mathbf{H}_\eta\boldsymbol{\Psi}\mathbf{H}_\eta + \boldsymbol{\Psi}^{-1})\mathbf{w} + (\mathbf{y} - \boldsymbol{\alpha} - \mathbf{f}_0)^\top\boldsymbol{\Psi}\mathbf{H}_\eta\mathbf{w}.
\end{aligned}
$$

Setting $\mathbf{A} = \mathbf{H}_\eta \boldsymbol{\Psi} \mathbf{H}_\eta + \boldsymbol{\Psi}^{-1}$, $\mathbf{a}^\top = (\mathbf{y} - \boldsymbol{\alpha} - \mathbf{f}_0)^\top \boldsymbol{\Psi} \mathbf{H}_\eta$, and using the fact that

$$\mathbf{w}^\top \mathbf{A} \mathbf{w} - 2\mathbf{a}^\top \mathbf{w} = (\mathbf{w} - \mathbf{A}^{-1}\mathbf{a})^\top \mathbf{A} (\mathbf{w} - \mathbf{A}^{-1}\mathbf{a}),$$

we have that $\mathbf{w}|\mathbf{y}$ is normally distributed with the required mean and variance.

Alternatively, one could have shown this using standard results of multivariate normal distributions. Noting that the covariance between $\mathbf{y}$ and $\mathbf{w}$ is

$$\begin{aligned} \operatorname{Cov}(\mathbf{y}, \mathbf{w}) &= \operatorname{Cov}(\boldsymbol{\alpha} + \mathbf{f}_0 + \mathbf{H}_\eta \mathbf{w} + \boldsymbol{\epsilon}, \mathbf{w}) \\ &= \mathbf{H}_\eta \operatorname{Cov}(\mathbf{w}, \mathbf{w}) \\ &= \mathbf{H}_\eta \boldsymbol{\Psi} \end{aligned}$$

and that $\operatorname{Cov}(\mathbf{w}, \mathbf{y}) = \boldsymbol{\Psi} \mathbf{H}_\eta = \mathbf{H}_\eta \boldsymbol{\Psi} = \operatorname{Cov}(\mathbf{y}, \mathbf{w})$ by symmetry, the joint distribution $(\mathbf{y}, \mathbf{w})$ is

$$\begin{pmatrix} \mathbf{y} \\ \mathbf{w} \end{pmatrix} \sim \mathrm{N}_{n+n} \left(\begin{pmatrix} \boldsymbol{\alpha} + \mathbf{f}_0 \\ \mathbf{0} \end{pmatrix}, \begin{pmatrix} \mathbf{V}_y & \mathbf{H}_\eta \boldsymbol{\Psi} \\ \mathbf{H}_\eta \boldsymbol{\Psi} & \boldsymbol{\Psi} \end{pmatrix} \right).$$

Thus,

$$\begin{aligned} \mathrm{E}[\mathbf{w}|\mathbf{y}] &= \mathrm{E}\,\mathbf{w} + \operatorname{Cov}(\mathbf{w}, \mathbf{y})(\operatorname{Var} \mathbf{y})^{-1}(\mathbf{y} - \mathrm{E}\,\mathbf{y}) \\ &= \mathbf{H}_\eta \boldsymbol{\Psi} \mathbf{V}_y^{-1} (\mathbf{y} - \boldsymbol{\alpha} - \mathbf{f}_0), \end{aligned}$$

and

$$\begin{aligned} \operatorname{Var}[\mathbf{w}|\mathbf{y}] &= \operatorname{Var} \mathbf{w} - \operatorname{Cov}(\mathbf{w}, \mathbf{y})(\operatorname{Var} \mathbf{y})^{-1} \operatorname{Cov}(\mathbf{y}, \mathbf{w}) \\ &= \boldsymbol{\Psi} - \mathbf{H}_\eta \boldsymbol{\Psi} \mathbf{V}_y^{-1} \mathbf{H}_\eta \boldsymbol{\Psi} \\ &= \boldsymbol{\Psi} - \boldsymbol{\Psi} \mathbf{H}_\eta \left(\boldsymbol{\Psi}^{-1} + \mathbf{H}_\eta \boldsymbol{\Psi} \mathbf{H}_\eta \right)^{-1} \mathbf{H}_\eta \boldsymbol{\Psi} \\ &= \left(\boldsymbol{\Psi}^{-1} + \mathbf{H}_\eta \boldsymbol{\Psi} \mathbf{H}_\eta \right)^{-1} \\ &= \mathbf{V}_y^{-1} \end{aligned}$$

as a direct consequence of the Woodbury matrix identity.

4.8 A recap on the exponential family EM algorithm

Consider the density function $p(\cdot|\boldsymbol{\theta})$ of the complete data $\mathbf{z} = \{\mathbf{y}, \mathbf{w}\}$, which depends on parameters $\boldsymbol{\theta} = (\theta_1, \dots, \theta_s)^\top \in \Theta \subseteq \mathbb{R}^s$, belonging to an exponential family of distributions. This density takes the form $p(\mathbf{z}|\boldsymbol{\theta}) = B(\mathbf{z}) \exp\big(\langle \boldsymbol{\eta}(\boldsymbol{\theta}), \mathbf{T}(\mathbf{z})\rangle - A(\boldsymbol{\theta})\big)$, where $\boldsymbol{\eta} : \mathbb{R}^s \mapsto \mathbb{R}$ is a link function, $\mathbf{T}(\mathbf{z}) = \big(T_1(\mathbf{z}), \dots, T_s(\mathbf{z})\big)^\top \in \mathbb{R}^s$ are the sufficient statistics of the distribution, and $\langle \cdot, \cdot \rangle$ is the usual Euclidean dot product. It is often easier to work in the *natural parameterisation* of the exponential family distribution

$$p(\mathbf{z}|\boldsymbol{\eta}) = B(\mathbf{z}) \exp\big(\langle \boldsymbol{\eta}, \mathbf{T}(\mathbf{z})\rangle - A^*(\boldsymbol{\eta})\big) \tag{4.11}$$

by defining $\boldsymbol{\eta} := \big(\eta_1(\boldsymbol{\theta}), \dots, \eta_r(\boldsymbol{\theta})\big) \in \mathcal{E}$, and $\exp A^*(\boldsymbol{\eta}) = \int B(\mathbf{z}) \exp \langle \boldsymbol{\eta}, \mathbf{T}(\mathbf{z})\rangle \,\mathrm{d}\mathbf{z}$ to ensure the density function normalises to one. As an aside, the set $\mathcal{E} := \big\{\boldsymbol{\eta} = (\eta_1, \dots, \eta_s) \,|\, \int \exp A^*(\boldsymbol{\eta}) < \infty\big\}$ is called the *natural parameter space*. If $\dim \mathcal{E} = r < s = \dim \Theta$, then the the pdf belongs to the *curved exponential family* of distributions. If $\dim \mathcal{E} = r = s = \dim \Theta$, then the family is a *full exponential family*.

Assuming the latent $\mathbf{w}$ variables are observed and working with the natural parameterisation, then the complete maximum likelihood (ML) estimate for $\boldsymbol{\eta}$ is obtained by solving

$$\frac{\partial}{\partial \boldsymbol{\eta}} \log p(\mathbf{z}|\boldsymbol{\eta}) = \mathbf{T}(\mathbf{z}) - \frac{\partial}{\partial \boldsymbol{\eta}} A^*(\boldsymbol{\eta}) = 0. \tag{4.12}$$

Of course, the variable $\mathbf{w}$ are never observed, so the ML estimate for $\boldsymbol{\eta}$ can only be informed from what is observed. Let $p(\mathbf{y}|\boldsymbol{\eta}) = \int p(\mathbf{y}, \mathbf{w}|\boldsymbol{\eta}) \,\mathrm{d}\mathbf{w}$ represent the marginal density of the observations $\mathbf{y}$. Now, the ML estimate for $\boldsymbol{\eta}$ is obtained by solving

$$\begin{aligned}
\frac{\partial}{\partial\boldsymbol{\eta}}\log p(\mathbf{y}|\boldsymbol{\eta}) &= \frac{1}{p(\mathbf{y}|\boldsymbol{\eta})}\cdot\frac{\partial}{\partial\boldsymbol{\eta}}p(\mathbf{y}|\boldsymbol{\eta}) \\
&= \frac{1}{p(\mathbf{y}|\boldsymbol{\eta})}\cdot\frac{\partial}{\partial\boldsymbol{\eta}}\left(\int p(\mathbf{y},\mathbf{w}|\boldsymbol{\eta})\,\mathrm{d}\mathbf{w}\right) \\
&= \frac{1}{p(\mathbf{y}|\boldsymbol{\eta})}\cdot\int\left(\frac{\partial}{\partial\boldsymbol{\eta}}p(\mathbf{y},\mathbf{w}|\boldsymbol{\eta})\right)\mathrm{d}\mathbf{w} \\
&= \frac{1}{p(\mathbf{y}|\boldsymbol{\eta})}\cdot\int\left(p(\mathbf{y},\mathbf{w}|\boldsymbol{\eta})\frac{\partial}{\partial\boldsymbol{\eta}}\log p(\mathbf{y},\mathbf{w}|\boldsymbol{\eta})\right)\mathrm{d}\mathbf{w} \\
&= \int\left(\mathbf{T}(\mathbf{y},\mathbf{w})-\frac{\partial}{\partial\boldsymbol{\eta}}A^*(\boldsymbol{\eta})\right)p(\mathbf{w}|\mathbf{y},\boldsymbol{\eta})\,\mathrm{d}\mathbf{w} \\
&= \mathrm{E}_{\mathbf{w}}\left[\mathbf{T}(\mathbf{y},\mathbf{w})|\mathbf{y}\right]-\frac{\partial}{\partial\boldsymbol{\eta}}A^*(\boldsymbol{\eta}) \qquad (4.13)
\end{aligned}$$

equated to zero. Note that we are allowed to change the order of integration and differentation provided the integrand is continuously differentiable. So the only difference between the first order condition of (4.12) and that of (4.13) is that the sufficient statistics involving the unknown $\mathbf{w}$ are replaced by their conditional or posterior expectations.

A useful identity to know is that $\frac{\partial}{\partial\boldsymbol{\eta}}A^*(\boldsymbol{\eta}) = \mathrm{E}_{\mathbf{z}}\,\mathbf{T}(\mathbf{z})$ (Casella and Berger, 2002, Theorem 3.4.2 & Exercise 3.32(a)), which can be expressed in terms of the original parameters $\boldsymbol{\theta}$. As a consequence, solving for the ML estimate for $\boldsymbol{\theta}$ from the FOC equations (4.13) is possible without having to deal with the derivative of A^* with respect to the natural parameters. Having said this, an analytical solution in $\boldsymbol{\theta}$ may not exist, because the relationship of $\boldsymbol{\theta}$ could be implicit in the set of equations $\mathrm{E}_{\mathbf{w}}\left[\mathbf{T}(\mathbf{w},\mathbf{y})|\mathbf{y},\boldsymbol{\theta}\right] = \mathrm{E}_{\mathbf{y},\mathbf{w}}\left[\mathbf{T}(\mathbf{y},\mathbf{w})|\boldsymbol{\theta}\right]$. One way around this is to employ an iterative procedure, as detailed in Algorithm 1.

Algorithm 1 Exponential family EM

```
1: initialise θ^(0) and t ← 0
2: while not converged do
3:     E-step: T̃^(t+1)(y, w) ← E_w [T(w, y)|y, θ^(t)]
4:     M-step: θ^(t+1) ← solution to T̃^(t+1)(y, w) = E_{y,w} [T(y, w)|θ]
5:     t ← t + 1
6: end while
```

To see how Algorithm 1 motivates the EM algorithm, consider the following argument. Recall that for the EM algorithm, the function $Q_t(\boldsymbol{\eta}) = \mathrm{E}_{\mathbf{w}}[\log p(\mathbf{y},\mathbf{w}|\boldsymbol{\eta})|\mathbf{y},\boldsymbol{\eta}^{(t)}]$ is maximised at each iteration t. For exponential families of the form (4.11), the Q_t

function turns out to be

$$Q_t(\boldsymbol{\eta}) = \mathrm{E}_{\mathbf{w}}\left[\langle \boldsymbol{\eta}, \mathbf{T}(\mathbf{z})\rangle | \mathbf{y}, \boldsymbol{\eta}^{(t)}\right] - A^*(\boldsymbol{\eta}) + \log B(\mathbf{z}),$$

and this is maximised at the value of $\boldsymbol{\eta}$ satisfying

$$\frac{\partial}{\partial \boldsymbol{\eta}} Q_t(\boldsymbol{\eta}) = \mathrm{E}_{\mathbf{w}}\left[\mathbf{T}(\mathbf{y}, \mathbf{w}) | \mathbf{y}, \boldsymbol{\eta}^{(t)}\right] - \frac{\partial}{\partial \boldsymbol{\eta}} A^*(\boldsymbol{\eta}) = 0,$$

a similar condition to (4.13) when obtaining ML estimate of $\boldsymbol{\eta}$. Thus, Q_t is maximised by the solution to line 4 in Algorithm (1).

4.9 Deriving the posterior predictive distribution

A priori, assume that $y_{\text{new}} \sim \mathrm{N}(\hat\alpha, v_{\text{new}})$, where $v_{\text{new}} = \mathbf{h}_{\hat\eta}(x_{\text{new}})^\top \hat{\boldsymbol{\Psi}} \mathbf{h}_{\hat\eta}(x_{\text{new}}) + \psi_{\text{new}}^{-1}$. Consider the joint distribution of $(y_{\text{new}}, \mathbf{y}^\top)^\top$, which is multivariate normal (since both y_{new} and $\mathbf{y}$ are. Write

$$\begin{pmatrix} y_{\text{new}} \\ \mathbf{y} \end{pmatrix} \sim \mathrm{N}_{n+1}\left(\begin{pmatrix} \hat\alpha \\ \hat\alpha \mathbf{1}_n \end{pmatrix}, \begin{pmatrix} v_{\text{new}} & \mathrm{Cov}(y_{\text{new}}, \mathbf{y}) \\ \mathrm{Cov}(y_{\text{new}}, \mathbf{y})^\top & \tilde{\mathbf{V}}_y \end{pmatrix} \right),$$

where

$$\begin{aligned} \mathrm{Cov}(y_{\text{new}}, \mathbf{y}) &= \mathrm{Cov}(f_{\text{new}} + \epsilon_{\text{new}}, \mathbf{f} + \boldsymbol{\epsilon}) \\ &= \mathrm{Cov}(f_{\text{new}}, \mathbf{f}) + \mathrm{Cov}(\epsilon_{\text{new}}, \boldsymbol{\epsilon}) \\ &= \mathrm{Cov}\left(\mathbf{h}_{\hat\eta}(x_{\text{new}})^\top \tilde{\mathbf{w}}, \mathbf{H}_{\tilde\eta} \tilde{\mathbf{w}}\right) + (\sigma_{\text{new},1}, \dots, \sigma_{\text{new},n}) \\ &= \mathbf{h}_{\hat\eta}(x_{\text{new}})^\top \hat{\boldsymbol{\Psi}} \mathbf{H}_{\tilde\eta} + \boldsymbol{\sigma}_{\text{new}}. \end{aligned}$$

The vector of covariances $\boldsymbol{\sigma}_{\text{new}}$ between observations $y_1, \dots, y_n$ and the predicted point y_{new} would need to be prescribed a priori (treated as extra parameters), or estimated again, which seems excessive. Assuming $\boldsymbol{\sigma}_{\text{new}} = \mathbf{0}$ would be acceptable, especially under an iid assumption the error precisions. In any case, using standard multivariate normal results, we get that $y_{\text{new}}|\mathbf{y}$ is also normally distributed with mean

$$\begin{aligned}
E[y_{\text{new}}|\mathbf{y}] &= \hat{\alpha} + (\mathbf{h}_{\hat{\eta}}(x_{\text{new}})^\top \hat{\mathbf{\Psi}} \mathbf{H}_{\tilde{\eta}} + \boldsymbol{\sigma}_{\text{new}}) \tilde{\mathbf{V}}_y^{-1} \tilde{\mathbf{y}} \\
&= \hat{\alpha} + \mathbf{h}_{\hat{\eta}}(x_{\text{new}})^\top \overbrace{\mathbf{h}_{\hat{\eta}}(x_{\text{new}})^\top \hat{\mathbf{\Psi}} \mathbf{H}_{\tilde{\eta}} \tilde{\mathbf{V}}_y^{-1} \tilde{\mathbf{y}}}^{\hat{\mathbf{w}}} + \boldsymbol{\sigma}_{\text{new}} \tilde{\mathbf{V}}_y^{-1} \tilde{\mathbf{y}} \\
&= \hat{\alpha} + \mathrm{E}\left[f(x_{\text{new}})|\mathbf{y}\right] + \text{mean correction term}
\end{aligned}$$

and variance

$$\begin{aligned}
\mathrm{Var}[y_{\text{new}}|\mathbf{y}] &= v_{\text{new}} - (\mathbf{h}_{\hat{\eta}}(x_{\text{new}})^\top \hat{\mathbf{\Psi}} \mathbf{H}_{\tilde{\eta}} + \boldsymbol{\sigma}_{\text{new}}) \tilde{\mathbf{V}}_y^{-1} (\mathbf{h}_{\hat{\eta}}(x_{\text{new}})^\top \hat{\mathbf{\Psi}} \mathbf{H}_{\tilde{\eta}} + \boldsymbol{\sigma}_{\text{new}})^\top \\
&= \mathbf{h}_{\hat{\eta}}(x_{\text{new}})^\top \hat{\mathbf{\Psi}} \hat{\mathbf{h}}_{\hat{\eta}}(x_{\text{new}}) + \psi_{\text{new}}^{-1} - \mathbf{h}_{\hat{\eta}}(x_{\text{new}})^\top \hat{\mathbf{\Psi}} \mathbf{H}_{\tilde{\eta}} \tilde{\mathbf{V}}_y^{-1} \mathbf{H}_{\tilde{\eta}} \hat{\mathbf{\Psi}} \mathbf{h}_{\hat{\eta}}(x_{\text{new}}) \\
&\quad + \text{variance correction term} \\
&= \mathbf{h}_{\hat{\eta}}(x_{\text{new}})^\top \big(\hat{\mathbf{\Psi}} - \hat{\mathbf{\Psi}} \mathbf{H}_{\tilde{\eta}} \tilde{\mathbf{V}}_y^{-1} \mathbf{H}_{\tilde{\eta}} \hat{\mathbf{\Psi}} \big) \mathbf{h}_{\hat{\eta}}(x_{\text{new}}) + \psi_{\text{new}}^{-1} \\
&\quad + \text{variance correction term} \\
&= \mathbf{h}_{\hat{\eta}}(x_{\text{new}})^\top \hat{\mathbf{V}}_y^{-1} \mathbf{h}_{\hat{\eta}}(x_{\text{new}}) + \psi_{\text{new}}^{-1} + \text{variance correction term} \\
&= \mathrm{Var}\left[f(x_{\text{new}})|\mathbf{y}\right] + \psi_{\text{new}}^{-1} + \text{variance correction term.}
\end{aligned}$$

Bibliography

Carpenter, B., A. Gelman, M. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. Brubaker, J. Guo, P. Li, and A. Riddell (2017). "Stan: A Probabilistic Programming Language". In: *Journal of Statistical Software, Articles* 76.1, pp. 1–32. DOI: 10.18637/jss.v076.i01.

Casella, G. and R. L. Berger (2002). *Statistical inference*. Vol. 2. Duxbury Pacific Grove, CA.

Cheng, C.-A. and B. Boots (2017). "Variational Inference for Gaussian Process Models with Linear Complexity". In: *Advances in Neural Information Processing Systems*, pp. 5190–5200.

Denwood, M. (2016). "**runjags**: An R Package Providing Interface Utilities, Model Templates, Parallel Computing Methods and Additional Distributions for MCMC Models in JAGS". In: *Journal of Statistical Software* 71.9, pp. 1–25. DOI: 10.18637/jss.v071.i09.

Eddelbuettel, D. and R. Francois (2011). "**Rcpp**: Seamless R and C++ Integration". In: *Journal of Statistical Software* 40.8, pp. 1–18. DOI: 10.18637/jss.v040.i08.

Jamil, H. and W. Bergsma (2017). "iprior: An R Package for Regression Modelling using I-priors". In: *Manuscript in submission.*

Lange, K. (1995). "A quasi-Newton acceleration of the EM algorithm". In: *Statistica sinica*, pp. 1–18.

Liu, C., D. B. Rubin, and Y. N. Wu (1998). "Parameter expansion to accelerate EM: The PX-EM algorithm". In: *Biometrika* 85.4, pp. 755–770.

Lunn, D. J., A. Thomas, N. Best, and D. Spiegelhalter (Oct. 2000). "WinBUGS - A Bayesian modelling framework: Concepts, structure, and extensibility". In: *Statistics and Computing* 10.4, pp. 325–337. DOI: 10.1023/A:1008929526011.

Plummer, M. (2003). "JAGS: A Program for Analysis of Bayesian Graphical Models Using Gibbs Sampling". In: *Proceedings of the 3rd International Workshop on Distributed Statistical Computing*. Vol. 124. Vienna, Austria, p. 125.

Stan Development Team (2016). **RStan**: *The R Interface to Stan*. R package version 2.14.1. URL: http://mc-stan.org/.

Sturtz, S., U. Ligges, and A. Gelman (2005). "**R2WinBUGS**: A Package for Running WinBUGS from R". In: *Journal of Statistical Software* 12.3, pp. 1–16. DOI: 10.18637/jss.v012.i03.

Wu, C. J. (1983). "On the convergence properties of the EM algorithm". In: *The Annals of statistics*, pp. 95–103.