MCMC

A PREPRINT

Ben Boys

August 5, 2019

Abstract

We use mathematical models to predict the behaviour of physical systems, whether it be structural beams, bridges, cells or circuits. This project aims to use a Bayesian approach to break down the error between measured data and an Euler-Bernoulli beam model into individual components of normally distributed measurement error, and a mismatch error between the model and reality.

1 The FE Method

We assume that the mathematical representation of the process, e.g. the deflection of a beam under loading, or the temperature distribution of a jet engine, is modelled by $u(x)$. This is governed by partial differential equation $\boldsymbol{\mathcal{L}_\theta} u(x) = -f(x)$ where $\boldsymbol{\mathcal{L}_\theta}$ is some linear differential operator with respect to the model parameters $\boldsymbol{\theta}$ (e.g. material properties such as thermal conductivity or bulk material stiffness). In their paper, 'The Statistical Finite Element Method', Girolami et al. induce uncertainty in the Finite Element model due to incomplete knowledge of the right hand side forcing term $f(x)$, and uncertain parameters $\boldsymbol{\theta}$. These uncertainties are formally incorporated into the probabilistic representation of the FE method by defining a probabilistic model of the randomised forcing term and making $\boldsymbol{\mathcal{L}_\theta}$ a randomised operator with respect to the uncertain parameters $\boldsymbol{\theta}$. However, for simplicity we employ the traditional representation of the finite element model, given certain parameters $\boldsymbol{\theta}$ and right hand side forcing term $f(x)$. Given a vector of spacial coordinates $\boldsymbol{x}_u$, $\boldsymbol{u}_h \in \mathbb{R}^{N\times 1}$ is the FE approximation to the function $u(x)$ at spacial coordinates $\boldsymbol{x}_u$

$$\boldsymbol{u}_h = \boldsymbol{A}^{-1}\boldsymbol{b} \tag{1}$$

where $\boldsymbol{A}$ is, for example, the stiffness matrix and $\boldsymbol{b}$ is the right-hand-side vector.

2 Problem Specifics

TO BE WRITTEN Deflections of a beam, Boundary conditions - simply supported, Load applied at centre, Dimensions, Material parameters

3 Finite Element Models Conditioned on Data

Consider a vector of measured deflections denoted as $\boldsymbol{y} \in \mathbb{R}^{N\times 1}$. This N dimensional vector represents N measurements taken simultaneously from N sensors. The important point to note here is that a statistical model of the data posits a true underlying generating process.

In their seminal work Kennedy and O'Hagan (2001) employed an additive functional error model to represent the model-reality mismatch, such that the 'true' partially known generating process values at the observation points are $\boldsymbol{\eta} \in \mathbb{R}^{N\times 1}$. As $\boldsymbol{\eta}$ is unobserved, its value is considered as a random function. This forms an additive regression model with random mismatch error $\boldsymbol{\delta}_h \in \mathbb{R}^{N\times 1}$ such that $\boldsymbol{\eta} = \boldsymbol{u}_h + \boldsymbol{\delta}$ with $\boldsymbol{\delta} \sim \mathcal{N}(0, \boldsymbol{K}_\lambda)$. Here, $\boldsymbol{K}_\lambda \in \mathbb{R}^{N\times N}$ is a matrix of covariance function evaluations $k_\delta(x, x')$, with length-scale λ, evaluated at the points $\boldsymbol{x}_y$.

$$K_{ij} = \exp(-\lambda |x_i - x_j|^2) \tag{2}$$

The final component in the data model is the observation error, which in this presentation is also assumed Gaussian. In which case, the data is now defined as $\boldsymbol{y} = \boldsymbol{\eta} + \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma_\epsilon^2 \boldsymbol{I})$. The conditional probability of the data under random vectors of the measure of the FE model-data mismatch, gives $p(\boldsymbol{y}|\boldsymbol{u}_h, \boldsymbol{\theta}, \boldsymbol{\delta}, \lambda, \sigma) = \mathcal{N}(\boldsymbol{u}_h + \boldsymbol{\delta}, \boldsymbol{I}\sigma_\epsilon^2)$. The marginal probability of the data, $p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\delta}, \lambda, \sigma)$, where the random vector $\boldsymbol{\delta}$ can be integrated out, and known parameters $\boldsymbol{\theta}$ are omitted in the notation for brevity, follows as,

$$p(\boldsymbol{y}|\lambda, \sigma) = \mathcal{N}(\boldsymbol{u}_h, \boldsymbol{K}_\lambda + \sigma_\epsilon^2 \boldsymbol{I}) \tag{3}$$

4 Using Monte Carlo procedures

We would like to infer information about the 'true' values of the parameters λ and σ. We could calculate the posterior distribution $p(\lambda, \sigma|\boldsymbol{y})$ analytically, but we will be using Monte Carlo procedures to generate a target distribution $p(\lambda, \sigma|\boldsymbol{y})$. In particular we will be using the Metropolis algorithm. We will use vector notation for brevity $\boldsymbol{w}_s = [\sigma, \lambda]^T$.

4.1 Metropolis Algorithm

The Metropolis Algorithm is as follows:

Algorithm 1 Metropolis Hastings Algorithm

procedure GETSAMPLES(S) ▷ Metropolis algorithm, S is number of iterations
  $s = 0$
  Choose $\boldsymbol{w}_s$
  while $s \leq S$ **do**
    $s = s + 1$
    Generate $\widetilde{\boldsymbol{w}}_s$ from $p(\widetilde{\boldsymbol{w}_s}|\boldsymbol{w}_{s-1})$
    Compute acceptance ratio r
    Generate u from $U(0, 1)$ ▷ Uniform distribution
    if $u \leq r$ **then**
      Accept sample, $\widetilde{\boldsymbol{w}}$
      $\boldsymbol{w}_s = \widetilde{\boldsymbol{w}_s}$
    else
      $\boldsymbol{w}_s = \boldsymbol{w}_{s-1}$
  return *accepted samples*

Generating $\widetilde{\boldsymbol{w}}_s$: $\widetilde{\boldsymbol{w}}_s$ is generated from a Gaussian proposal distribution, located at $\boldsymbol{w}_{s-1}$, which is the last accepted sample. $p(\widetilde{\boldsymbol{w}_s}|\boldsymbol{w}_{s-1}) = \mathcal{N}(\boldsymbol{w}_{s-1}, \Sigma)$. $\boldsymbol{\Sigma}$ is the covariance of the Gaussian proposal density. The Gaussian is a popular choice for the proposal density because it is symmetric: moving to $\widetilde{\boldsymbol{w}_s}$ from $\boldsymbol{w}_{s-1}$ is just as likely as moving from $\boldsymbol{w}_{s-1}$ to $\widetilde{\boldsymbol{w}_s}$, which simplifies the acceptance ratio:

$$r = \frac{p(\widetilde{\boldsymbol{w}}_s|\boldsymbol{y})}{p(\boldsymbol{w}_{s-1}|\boldsymbol{y})} \frac{p(\boldsymbol{w}_{s-1}|\widetilde{\boldsymbol{w}}_s, \boldsymbol{\Sigma})}{p(\widetilde{\boldsymbol{w}}_s|\boldsymbol{w}_{s-1}, \boldsymbol{\Sigma})} = \frac{p(\widetilde{\boldsymbol{w}}_s|\boldsymbol{y})}{p(\boldsymbol{w}_{s-1}|\boldsymbol{y})} \tag{4}$$

Using a nonsymmetric proposal density is known as the Metropolis-Hastings algorithm, whereas using a symmetric proposal density is known as the Metropolis algorithm. What's left is the first term, the ratio of the posterior density at the proposed sample to that of the old sample. We do not know the proposal distributions exactly because we cannot normalise them. However, because we are interested in a ratio, the normalisation constants cancel. So, applying Bayes rule, we can substitute the ratio of posteriors with the ratio of priors multiplied by the ratio of likelihoods. Putting independent priors on σ and λ, $\pi(\sigma)$ and $\pi(\lambda)$ leads us to the following expression:

$$r = \frac{p(\boldsymbol{y}|\sigma_s, \lambda_s)}{p(\boldsymbol{y}|\sigma_{s-1}, \lambda_{s-1})} \frac{\pi(\sigma_s)\pi(\lambda_s)}{\pi(\sigma_{s-1})\pi(\lambda_{s-1})} \tag{5}$$

The exact priors that we are putting on σ and λ, are Gamma distributions, as these parameters must be strictly positive.

$$\pi(\sigma|\alpha_\sigma, \beta_\sigma) = Ga(\alpha_\sigma, \beta_\sigma) \tag{6}$$

$$\pi(\lambda|\alpha_\lambda, \beta_\lambda) = Ga(\alpha_\lambda, \beta_\lambda) \tag{7}$$

where the gamma distribution pdf for some random variable, z is defined as

$$P(z|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} z^{\alpha-1} \exp(-\beta z) \tag{8}$$

The prior allows us to express a belief about the true values of σ, λ. σ is likely to be small but non zero, since we would expect a small percentage measurement error, and being a length scale, λ will be small if $\boldsymbol{\delta}$ is going to be a non-zero vector. We might then pick a gamma distribution with a high likelihood close to 0. This can be done by setting the shape parameter $\alpha_\sigma = 1.0$, $\alpha_\lambda = 1.0$. The rate parameters, β_σ and β_λ should depend on the relative sizes of σ and λ . A high rate parameter will give a smaller spread. A low value of λ is necessary for the $\boldsymbol{\delta}$ to have a visible effect on the data. Since lambda is going to be much smaller than σ, then is should have a lower spread about 0. Values of $\beta_\sigma = 1.0$, $\beta_\lambda = 100$ were used. The likelihood is calculated from equation (3).

4.2 Generating data

To test the model, and to see if the proposed method works, we will generate 'good' data using prescribed values of $\sigma = 0.6$ and $\lambda = 0.00823$ and run the Markov chain using the data to obtain a posterior over σ and λ. Generating 'good' data is necessary to test the proposed method. To make things simple, model displacements, errors and measured displacements have all been generated at the same 'sensor locations'. For speed, instead of using a Finite Element package, the Euler-Bernoulli beam model was used to generate the model data, $\boldsymbol{u}_h$. This example data is shown below.

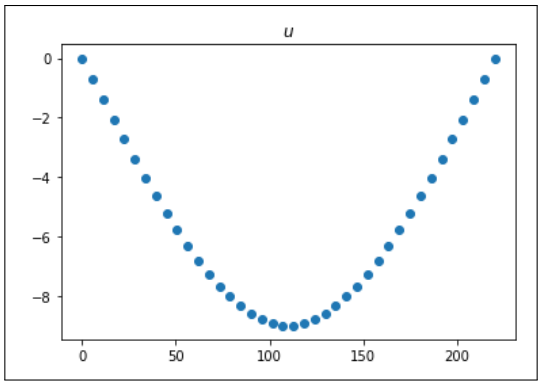


Figure 1: Model solution showing displacements, u_h at the 'sensor locations' on the y-axis and distance from support on x-axis.

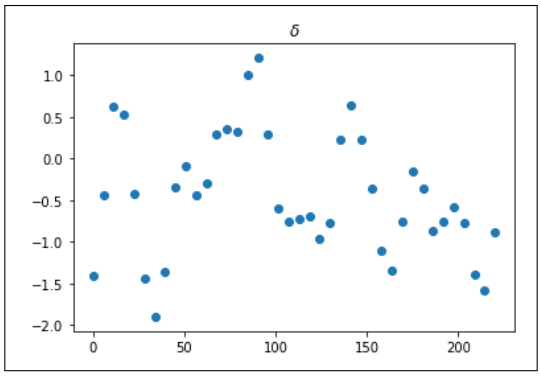


Figure 2: The model-reality mismatch error at the sensor locations, δ

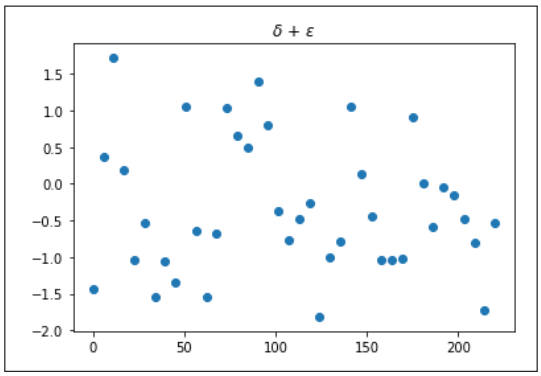


Figure 3: the total error with added measurement noise term $\delta + \epsilon$

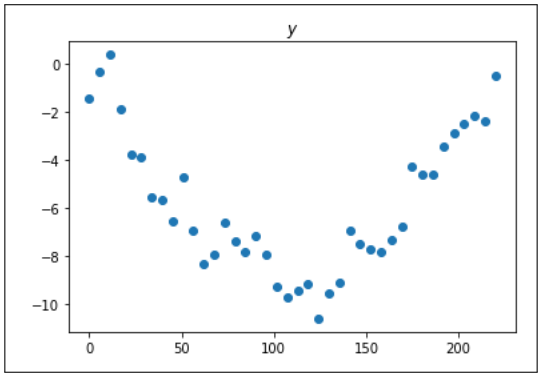


Figure 4: 'measured' displacements, y in mm

4.3 First results

For a first try, the proposal density covariance was chosen to be $\Sigma = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.2 \end{bmatrix}$ and the first sample location was chosen to be $\sigma = 1.0$, $\lambda = 1.0$.

The Metropolis algorithm was ran for a long period of time (100,000 iterations, taking 500s). The acceptance rate (after counting burn in samples as rejected) was low at 4.007%, giving 4007 samples. A burn was performed on the first 200 samples: whether this is enough for the Markov chain to converge should be explored later.

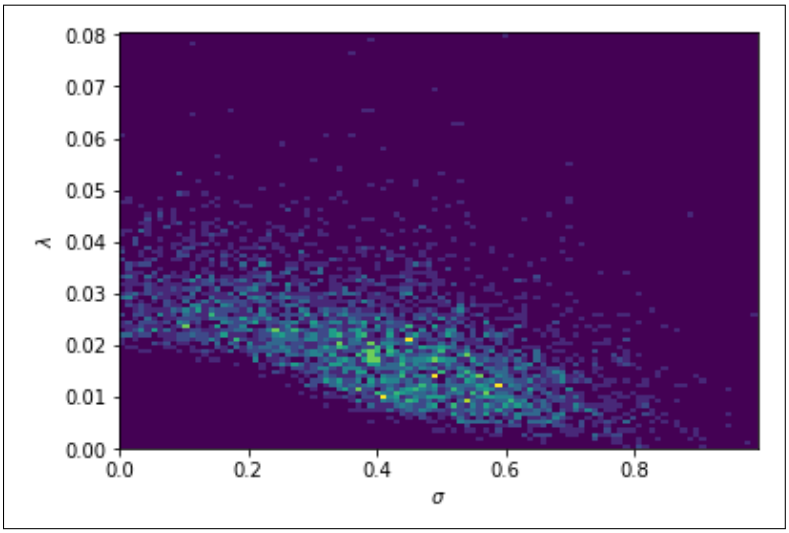


Figure 5: histogram of 4007 samples of lambda, sigma pairs

The histogram seems to show convergence, given that the start point was chosen at $\sigma_0 = 1.0$, $\lambda_0 = 1.0$. The posterior density picks out a high likelihood at the prescribed value of $\sigma = 0.6$ and $\lambda = 0.00823$, however, there is a sharp drop off of samples close to $\lambda = 0$. This is because if the proposal sample is negative or 0, then it is always rejected (given the gamma prior), so there will be a sharp drop off of samples close to 0, skewing the distribution. To solve this issue, the proposal density was made 200 times smaller in each direction, and a closer starter location to lambda was picked $\lambda_0 = 0.1$ so that the chain would converge quicker.

Another point to note is the shape of the distribution. The dependence of λ on σ may be because, for example, in the hypothetical case that none of the error can be explained by σ (i.e. say $\sigma = 0$), then more of the error has to be explained by λ. Therefore the covariance matrix K, generated by λ must look more like IID (independent, identically distributed), so λ must get larger, increasing the length scale above the dimension of the beam.

As well as decreasing the size of the proposal distribution $\Sigma = \begin{bmatrix} 0.001 & 0 \\ 0 & 0.001 \end{bmatrix}$, the number of iterations was doubled to 200,000. As expected, the acceptance rate was higher at a reasonable 19.778% giving 39,556 samples. The histogram looks much the same as before with a high, but by no means highest, amount of samples close to the 'true' values of $\sigma = 0.6$ and $\lambda = 0.00823$

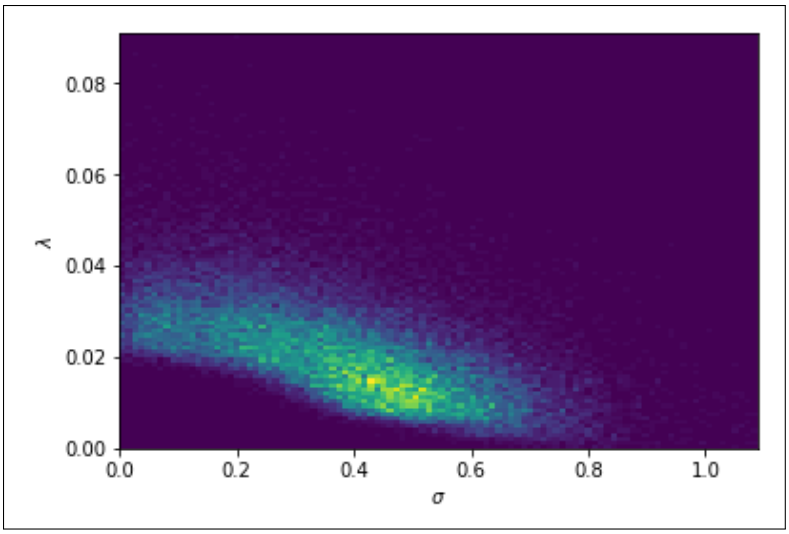
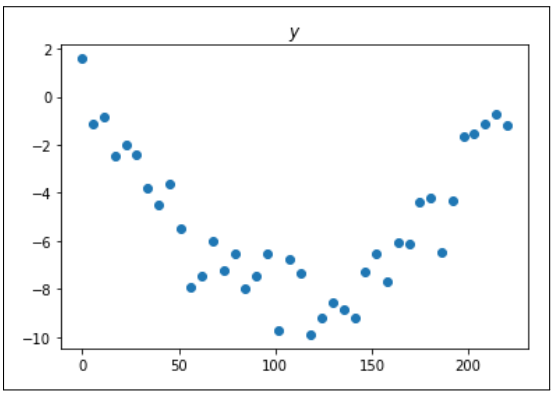
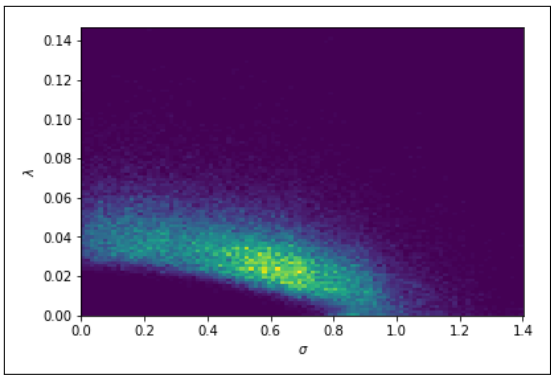


Figure 6: histogram of 39556 samples of λ, σ pairs

The data used to produce these distributions is convenient. Let us try using different values of lambda and sigma with the same proposal densities and priors.

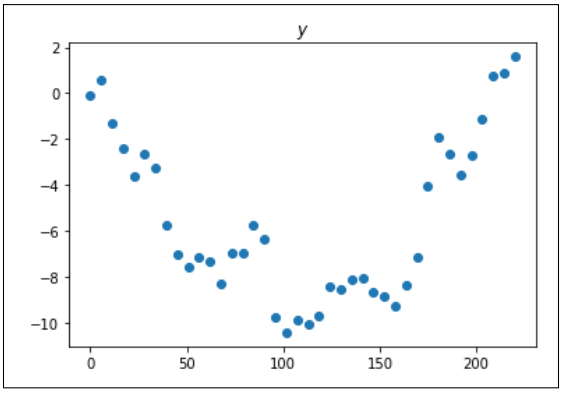


(a) 'Measured' displacements in mm

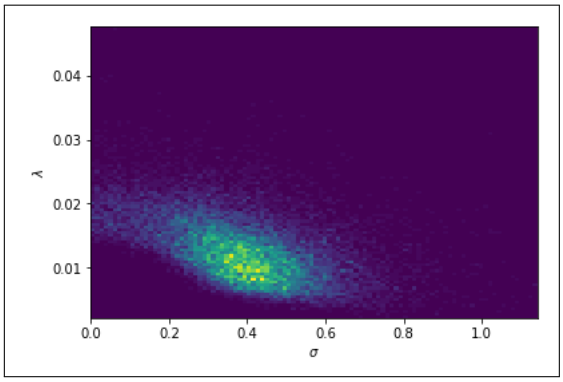


(b) Histogram of samples

Figure 7: Samples generated with values $\sigma = 0.6$, $\lambda = 0.04$

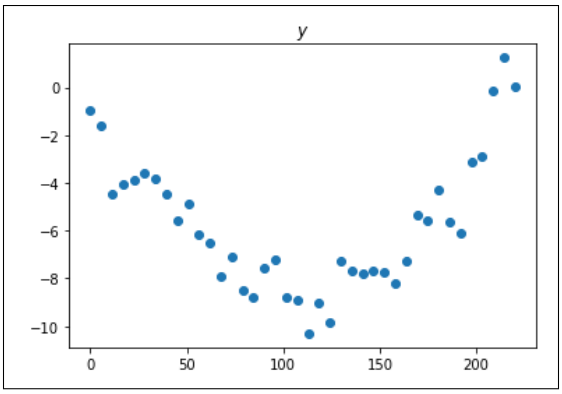


(a) 'Measured' displacements in mm

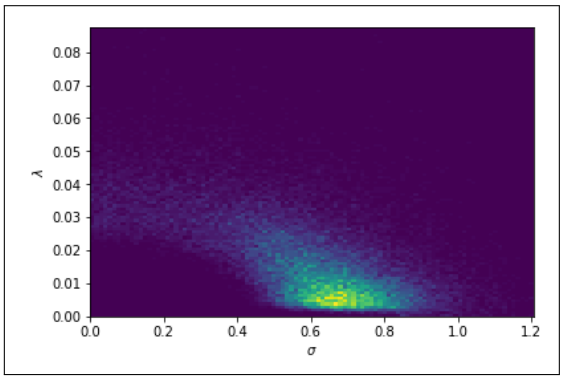


(b) Histogram of accepted samples, acceptance rate 10.8415%

Figure 9: Samples generated with values $\sigma = 0.4$, $\lambda = 0.01$

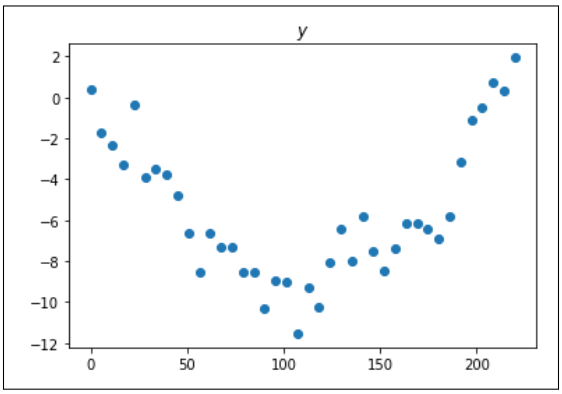


(a) 'Measured' displacements in mm

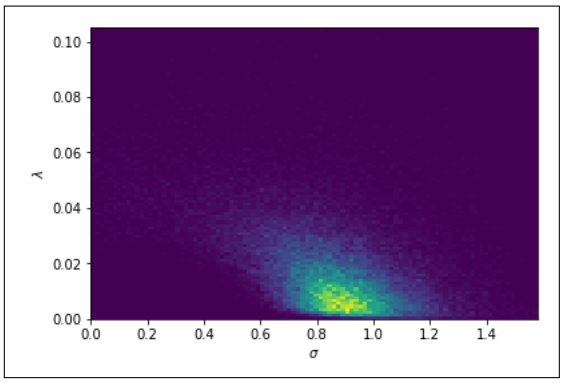


(b) Histogram of samples

Figure 8: Samples generated with values $\sigma = 0.4$, $\lambda = 0.04$



(a) 'Measured' displacements in mm, generated with values $\sigma = 0.8$, $\lambda = 0.01$



(b) Histogram of accepted samples

Figure 10: Samples generated with values $\sigma = 0.8$, $\lambda = 0.01$

4.4 Multiple sets of observations

In the case where there are multiple sets of observations $\boldsymbol{y}_i$ at the coordinates x_y, where the sensors are located, for $i = 1, ..., L$, it can be shown that the joint distribution of all $\boldsymbol{y}_i$ conditional on $\boldsymbol{u}_h$ is given in product form by assuming independence between the sets of observations. Let $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_L] \in \mathbb{R}^{N \times L}$, the joint distribution of $\boldsymbol{y}_i$ conditional on $\boldsymbol{u}_h$ given all sets of observations is

4.5 1 dimensional search on lambda

4.6 Changing of variables

The low sample rate is likely to be a problem with how close to 0 the lambda is. The likelihood of samples close to zero is very low since if the proposal sample is negative or 0, then it is always rejected, so likelihood will have a sharp drop off close to 0, skewing the distribution. To solve this issue, we could attempt make a change of variables

$$\lambda = \exp(\zeta) \tag{9}$$

For a continuous random variable, the relationship between the cdf

$$F_x(x) = Pr(X \leq x) = \int_{-\infty}^{x} f_x(t)dt \tag{10}$$

and its pdf $f_x(t)$ is

$$f_x(t) = \frac{dF_x(t)}{dx} \tag{11}$$

The cdf is useful when characterising the probability distribution of a transformation of a random variable. That is from X define new random variables, e.g. $Y = \exp(X)$ or $Y = X^2$.

Let X have pdf f_X and cdf F_X. Define the new random variable $Y = r(X)$ and then follow these steps to derive the pdf f_Y.

1. For each y find the set $A_y = \{x : r(x) \leq y\}$
2. Find the cdf of Y.

$$\begin{aligned} F_Y(y) &= P(Y \leq y) = P(r(X) \leq y) \\ &= P(X \in A_y) \\ &= \int_{-\infty}^{\infty} \mathbb{1}_{A_y}(x) f_x(x) dx \end{aligned} \tag{12}$$

3. Let $F_y^{'}(y)$ be the derivative of $F_y(y)$ at y (when it exists). Set $f_Y(y) = F_Y^{'}(y)$

When r is strictly increasing or strictly decreasing we can derive a formula for f_Y. In this case r has an inverse, let $s = r^{-1}$ Then

$$f_Y(y) = f_X(s(y)) \left| \frac{ds(y)}{dy} \right| \tag{13}$$

Making the suggested substitution, $\zeta = \ln \lambda$, such that

$$s(\zeta) = exp(\zeta) \tag{14}$$

since $r(\lambda) = \ln |\lambda|$ is strictly increasing, then we can derive a formula for f_ζ.

$$\begin{aligned} f_\zeta(\zeta) &= f_\lambda(s(\zeta)) \left| \frac{ds(\zeta)}{d\zeta} \right| \\ &= \frac{\beta^\alpha}{\Gamma(\alpha)} \exp(\zeta)^{\alpha-1} \exp\left(-\beta \exp(\zeta)\right) \left| \frac{d(\exp(\zeta))}{d\zeta} \right| \\ &= \frac{\beta^\alpha}{\Gamma(\alpha)} \exp\left(\zeta(\alpha - 1) - \beta \exp\left(\zeta\right) + 1\right) \\ &= \frac{\beta^\alpha}{\Gamma(\alpha)} \exp\left(\zeta\alpha - \beta \exp\left(\zeta\right)\right) \end{aligned} \tag{15}$$

This distribution, with $\alpha = 1.0$ and $\beta = 100.0$ looks like

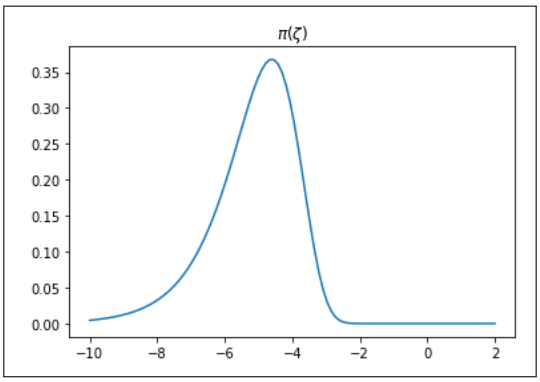


Figure 11: Prior on ζ with $\alpha = 1.0$ and $\beta = 100.0$

with $\alpha = 1.0$ and $\beta = 300.0$ looks like

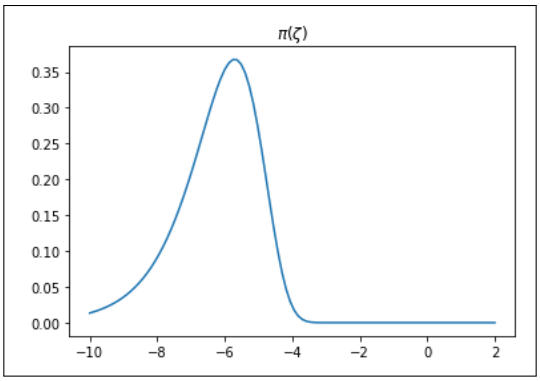


Figure 12: Prior on ζ with $\alpha = 1.0$ and $\beta = 300.0$

The result of this change of variables drastically increases the acceptance rate of our sampler to between $80\% - 90\%$, which is a good thing. Generating data with values of $\lambda = 0.00823$ (i.e. $\zeta = -4.8$) and $\sigma = 0.6$ and the prior shown in Fig. 11 gives the following results.

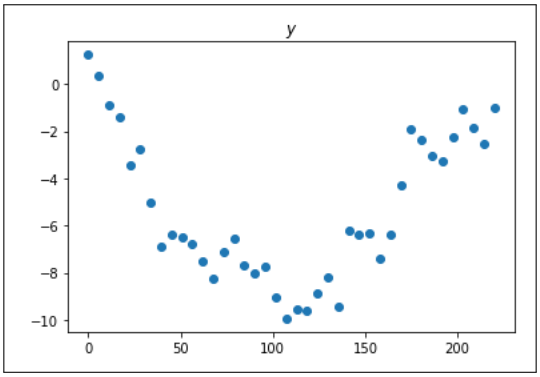


Figure 13: 'Measured' displacements in mm, generated with values $\sigma = 0.8$, $\zeta = -4.8$

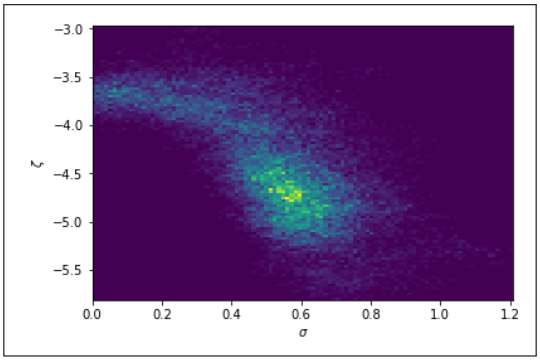


Figure 14: Histogram of accepted samples, acceptance rate 86.9% with the prior shown in Fig. 15

with the same data and a different prior, shown in Fig. 12 gives the following results

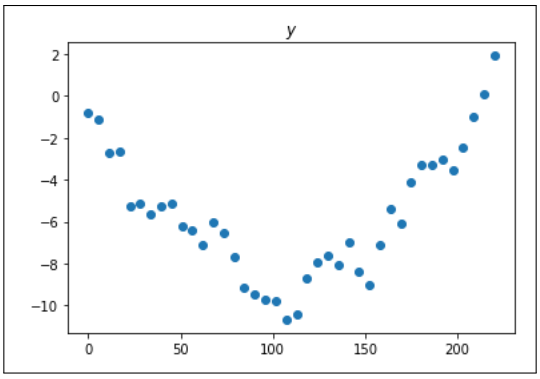


Figure 15: 'Measured' displacements in mm, generated with values $\sigma = 0.8$, $\zeta = -4.8$

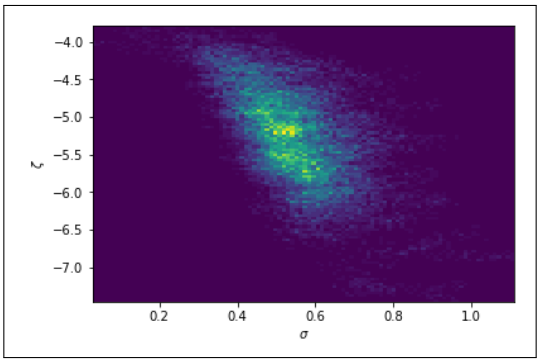


Figure 16: Histogram of accepted samples, with the prior shown in Fig. 16

This shows that the target distribution is strongly dependent on the parameters put on the prior. Therefore, the prior should be selected carefully. NOTE: here would be a good place to fix sigma and sample from a 1D distribution on lambda. Also add more sensors to see if the posterior concentrates more on the 'true' values.

5 Appendix

5.1 Symmetric bi-modality

Since the traditional notation of $VAR(\epsilon) = \sigma^2$ was used, and $\sigma = \sqrt{VAR}$ then σ can be positive or negative. Putting a gaussian prior $\pi(\sigma) = \mathcal{N}(0, 5.0)$ on σ and using a proposal density $\Sigma = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$ yields the following results

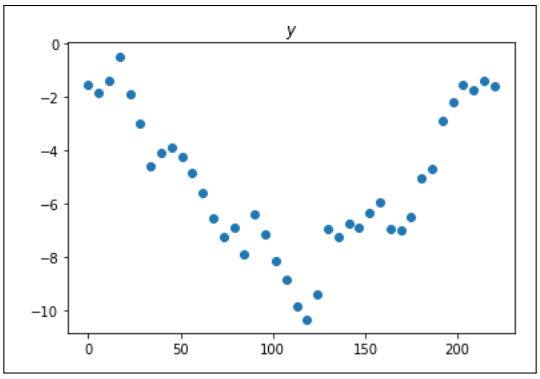


Figure 17: 'Measured' displacements in mm, generated with values $\sigma = 0.3$, $\zeta = -4.8$

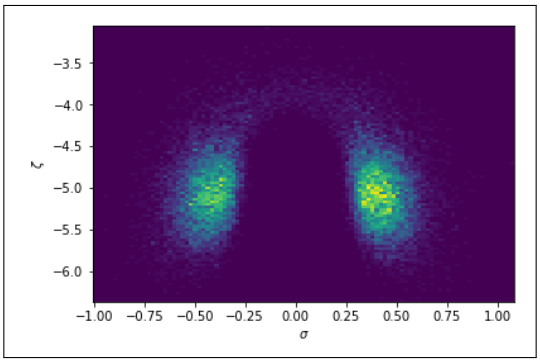


Figure 18: Symmetric bi-modality

5.2 Demonstration of Metropolis sampler using R parameter to test convergence, and adaptive sampling

A multivariate Gaussian distribution with high, positive covariance $\Sigma = \begin{bmatrix} 1.0 & 0.95 \\ 0.95 & 1.0 \end{bmatrix}$ and mean of $[2.0, 2.0]$ was sampled from using my Metropolis sampler with adaptive proposals, tests for convergence and burn in, with the following results: 10000 samples were taken at a high acceptance rate of 79.32%. The adaptive proposals increased the acceptance rate by more than 40%. The Markov Chain reached convergence (defined as R <1.01 for both parameters, X1 and X2) at $3.0e3$ samples. These samples were burned.

The mean of the MCMC samples was $[2.0231432798619857, 2.0095061284314486]$

The covariance of the MCMC samples was $\Sigma_s = \begin{bmatrix} 0.91526046 & 0.85018028 \\ 0.85018028 & 0.88587958 \end{bmatrix}$

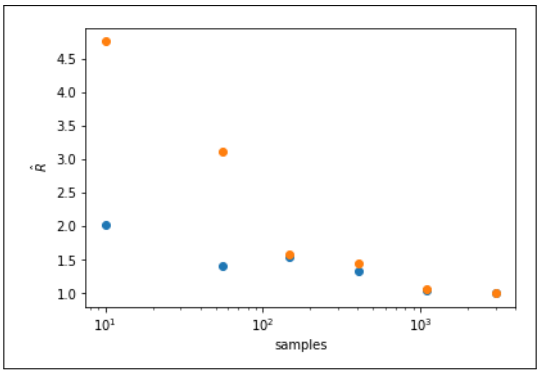


Figure 19: Convergence test, showing the burn in phase

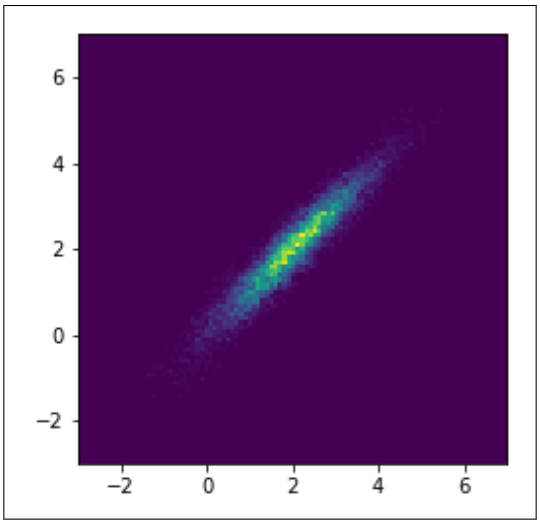


Figure 20: Histogram of samples

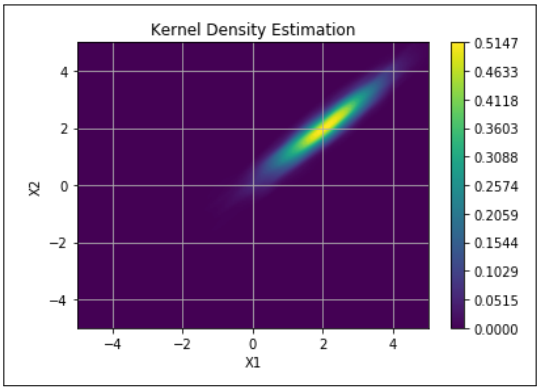


Figure 21: Kernel Density Estimate of the generated samples

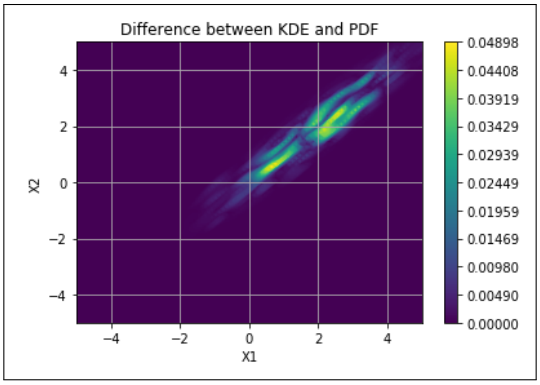


Figure 22: Difference between true pdf and KDE

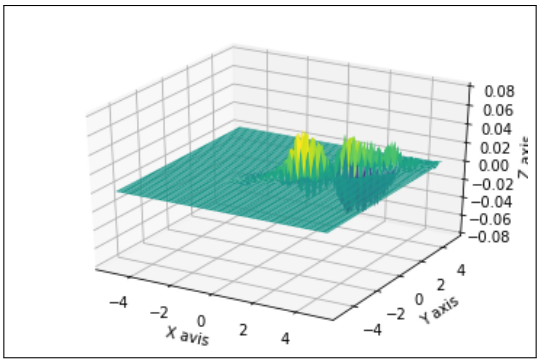


Figure 23: 3D plot of the difference between true pdf and KDE

References

[1] Girolami, M., Gregory, A. Yin, G., Cirak, F. (2019). The Statistical Finite Element Method