Assignment 2 - Inverse Problems

Linear, Non-Gaussian Inversion of Magnetic Data

Data

The data file ”dataM.txt” contains an idealized magnetic field profile measured above a plate, magnetized in parallel "stripes" (see figure).

The file ”dataM.txt” is organized in 2 columns. The first column contains the horizontal coordinate x in cm of the measurement points along a profile perpendicular to the stripes. The second column contains the measured values of the vertical component of the magnetic field in nT. The uncertainties of the measured data are statistically independent and Gaussian with mean 0 nT and standard deviation 25 nT.

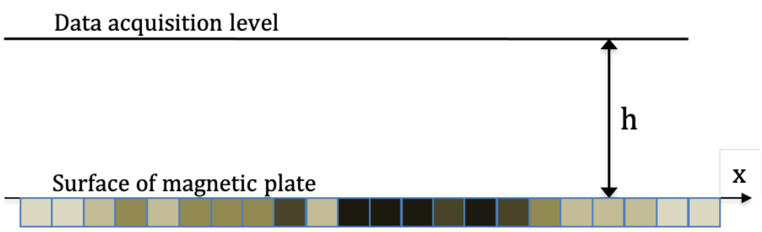

Figure 1: Measurement geometry. The figure shows a cross section of the plate, perpendicular to the magnetized stripes.

A priori information

To simplify the calculation, we assume that we have a discretized model of the magnetized plate with 200 equally wide, linear bands and a total width of 100 cm.

Let us assume that we have the following a priori knowledge about the magnetization structure of the plate: The magnetization is uniform within (physical) stripes, each consisting of one or several bands. The horizontal widths w of the physical stripes are distributed according to the exponential probability density

$$f(w) = \frac{1}{w_0} \exp\left(-\frac{w}{w_0}\right) \tag{1}$$

where w_0 , the mean stripe width, has the value $w_0 = 4$ cm.

Independently of the width of the stripes, the surface dipole magnetization m of each stripe follows a Gaussian probability density with mean 0 Am^{-1} and standard deviation $\sigma = 0.025$ Am^{-1}.

It can be shown that, to sample the prior (1), one can proceed by performing one of the following three perturbations in each iteration:

- changing the magnetization in one stripe,
- adding a new stripe boundary and assigning (new) magnetizations to the stripes on both sides of it, or
- removing one stripe boundary and assigning a (new) magnetization to the new compound stripe.

In each iteration it is first decided which kind of model perturbation should be performed next. Performing a "pure" stripe magnetization perturbation has the same probability (0.5) as performing a pure stripe boundary perturbation (removing or adding a boundary).

In case of a step involving a pure stripe magnetization perturbation, a stripe is selected uniformly at random and a (new) magnetization is chosen for that stripe according to a Gaussian magnetization prior with mean 0 Am^{-1} and standard deviation $\sigma = 0.025\ \mathrm{Am}^{-1}$.

In case of a stripe boundary perturbation step we exploit the fact that (approximately) exponentially distributed stripe thicknesses can be obtained by assuming that the probability that a stripe interface is present at a given position (sample point) is equal to (0.5 cm/w_0) = 0.125 and independent of the presence of other stripe interfaces.

A stripe boundary perturbation step therefore works as follows: First, we select one of the 200 discrete points of the current magnetization function, uniformly at random. We then randomly decide if there should exist a stripe boundary at that point or not. The probability for the point to be a stripe boundary is 0.125. In case this operation creates a new stripe boundary, we generate a magnetization for the stripes around the new stripe boundary according to the Gaussian magnetization prior. In case this operation removes a stripe boundary, we generate a magnetization for the new compound stripe (consisting of the stripes around the removed stripe boundary) according to the magnetization prior.

The direct problem

We wish to estimate a parameterized model for the magnetization of the plate, based on the magnetic observations. We assume that the plate magnetization only depends on the horizontal coordinate x.

We let the z-axis point vertically upward and the y-axis be parallel to the stripes. We further assume that the sources of the field is in the plate $z = 0\ cm$, and that they are vertical dipoles whose intensity (surface dipole magnetization) only varies with x. The observations (which are only z components) are made on a plane located at $z = h = 2\ cm$.

It can be shown that a datum d_j, measured at x_j, can be expressed through the integral

$$d_j = \int_{-\infty}^{\infty} g_j(x) m(x) dx,$$

where

$$g_j(x) = -\frac{\mu_0}{2\pi} \frac{\left(x_j - x\right)^2 - h^2}{\left[\left(x_j - x\right)^2 + h^2\right]^2}.$$

and $m(x)$ is the surface dipole magnetization as a function of x.

Consider a finite set of x-values: $x_1, x_2, \ldots, x_M$. Let us represent $m(x)$ by the vector:

$$\mathbf{m} = (m(x_1), m(x_2), \ldots, m(x_M)) \tag{2}$$

This leads to a discretized expression:

$$g_i(x_j) = -\frac{\mu_0}{2\pi} \frac{\left(x_i - x_j\right)^2 - h^2}{\left[\left(x_i - x_j\right)^2 + h^2\right]^2} \tag{3}$$

The inverse problem

The inverse problem consists of estimating $m(x)$ from the observed data. The problem is linear, but because of the non-Gaussian prior, it cannot be analyzed by means of techniques from linear algebra. We shall therefore follow a different path where we use the Metropolis Algorithm to sample solutions to the problem. The probability density of solutions is the so-called *a posteriori* probability density

$$\sigma(\mathbf{m}) = \frac{\rho(\mathbf{m}) L(\mathbf{m})}{\mu(\mathbf{m})},$$

where $\rho(\mathbf{m})$ is the *a priori*-probability density, $L(\mathbf{m})$ is the *Likelihood-function*, and $\mu(\mathbf{m})$ is the null-information probability density.

1. Explain why the the Likelihood-function is given by

   $$L(\mathbf{m}) = \text{constant } \cdot \exp\left(-\frac{1}{2}(\mathbf{d}_{obs} - \mathbf{G}\mathbf{m})^T \mathbf{C}_d^{-1} (\mathbf{d}_{obs} - \mathbf{G}\mathbf{m})\right),$$

   where $g_i(x_j)$ is the ij'th component of the matrix $\mathbf{G}$, and $\mathbf{C}_d$ is the data covariance matrix (which is also the covariance matrix for the noise).

2. Build an algorithm that samples the prior $\rho(\mathbf{m})$, as described above.
3. Explain how you choose the null-information probability density $\mu(\mathbf{m})$ for this problem.
4. Using the (Extended) Metropolis Algorithm, generate a large number of models $\mathbf{m}^{(n)}$, $n = 1 \ldots N$, distributed according to the *a posteriori* distribution
5. Find the uncertainties of the model parameters.