Inference in the linear model

Georgios Arampatzis

March 19, 2019

1 The model

In this tutorial we will describe how to run the Matlab code for the inference of the parameters in the linear model. We consider the model,

$$y = f(x; a, b) = ax + b\,. \tag{1}$$

We are given N_d data points $\boldsymbol{d} = \{y_i, x_i\}_{i=1}^{N_d}$. We consider the statistical model,

$$y = ax + b + \varepsilon, \tag{2}$$

where ε is a random variable that follows normal distribution with mean zeros and standard deviation σ. We assume that the data y_i, $i = 1, \ldots, N_d$ are i.i.d samples from the model in eq. (2).

Set

$$\vartheta = (a, b, \sigma)$$

the vector of parameters to be inferred. The likelihood of $\boldsymbol{d}$ under the assumption eq. (2) is given by,

$$p(\boldsymbol{d}|\vartheta) = \mathcal{N}\left(\boldsymbol{d}|\boldsymbol{\mu}, \Sigma\right)\,, \tag{3}$$

where

$$\boldsymbol{\mu} = \big(f(x_1; a, b), \ldots, f(x_{N_d}; a, b)\big)^\top \quad \text{and} \quad \Sigma = \sigma^2 I\,. \tag{4}$$

The log-likelihood is given by,

$$\log p(\boldsymbol{d}|\vartheta) = -\frac{1}{2} N_d \log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{N_d}(y_i - f(x_i;\vartheta))^2\,. \tag{5}$$

The derivatives and the Fisher information matrix has been implemented in the provided Matlab functions. The user has to implement the derivatives $\frac{\mathrm{d}}{\mathrm{d}\vartheta_j} f(x_i, \vartheta)$.

2 Running the code

In order to validate our implementation we will first create synthetic data with known parameters and then infer these parameters. From the root directory (the directory that contains the folders `data` and `engines`),

```
cd data/linear
make_data
```

This script will create the file `data.mat`. Using these data we will infer a, b and σ. Finally, from the root direcotry run

```
addpath(engine/postprocessing/).
```

2.1 Likelihood optimization

First we can optimize the likelihood function using the CMA-ES algorithm. From the root directory

```
cd engine/optimize/
edit run_linear_CMA.m
```

The important lines in this script are

```
addpath('../functions/linear/')

loglike_func = 'linear_loglike_cma';

d = load( [data_folder 'data.mat' ] );
```

and

```
opts.LBounds = [ -5 -5  0 ]';
opts.UBounds = [  5  5 10 ]';
```

The `addpath` command inserts the likelihood functions folder in the search path. In this case the log-likelihood is `linear_loglike_cma` and is located in the folder `../functions/linear/`. This function return the negative of the function `linear_loglike` that implements the log-likelihood as defined in eq. (5). The `opts.LBounds` and `opts.UBounds` variables set the search interval for the variables. Here, $a \in [-5, 5]$, $b \in [-5, 5]$ and $\sigma \in [0, 10]$. Finally, the command

```
start_parallel( 2, false );
```

will start a pool with two workers and run CMA in parallel. If you don't want parallel execution comment out this line. Running the script gives the results,

```
Maximum of log-likelihood at:
a     = 1.867522
b     = -1.741286
sigma = 1.770017
```

2.2 Sampling the posterior with BASIS

From the root directory

```
cd engine/sample/
edit run_linear.m
```

uncomment the line

```
method = 'BASIS'; cov_check = 'NONE';
```

and run the script. After the end of execution plot the results by running

```
plotmatrix_hist(BASIS.theta)
```

or

```
load ../../data/linear/BASIS_NONE_5000.mat
plotmatrix_hist(out_master.theta)
```

The results are shown in fig. 1.

2.3 Sampling the posterior with smTMCMC

First lets examine the function `linear_loglike_smMALA`. From root

```
cd engine/functions/linear/
edit linear_loglike_smMALA.m
```

The important lines are

```
% the model
Y = theta(1)*x + theta(2);

% the derivatives (row vectors)
dY1 = x(:)';
dY2 = ones(1,Nd);
```

Here the variable `dY1` is a row vector that contains the derivative $\frac{\mathrm{d}}{\mathrm{d}\vartheta_1} f(x_i, \vartheta)$ at the i-th position and the variable `dY2` is a row vector that contains the derivative $\frac{\mathrm{d}}{\mathrm{d}\vartheta_2} f(x_i, \vartheta)$ at the i-th position.

In order to run, from the root directory

```
cd engine/sample/
edit run_linear.m
```

uncomment the line

```
method = 'smMALA'; cov_check = 'EIG';
```

and execute the script. After the end of execution plot the results by running

```
plotmatrix_hist(BASIS.theta)
```

or

```
load ../../data/linear/smMALA_EIG_5000.mat
plotmatrix_hist(out_master.theta)
```

The results are shown in fig. 2.

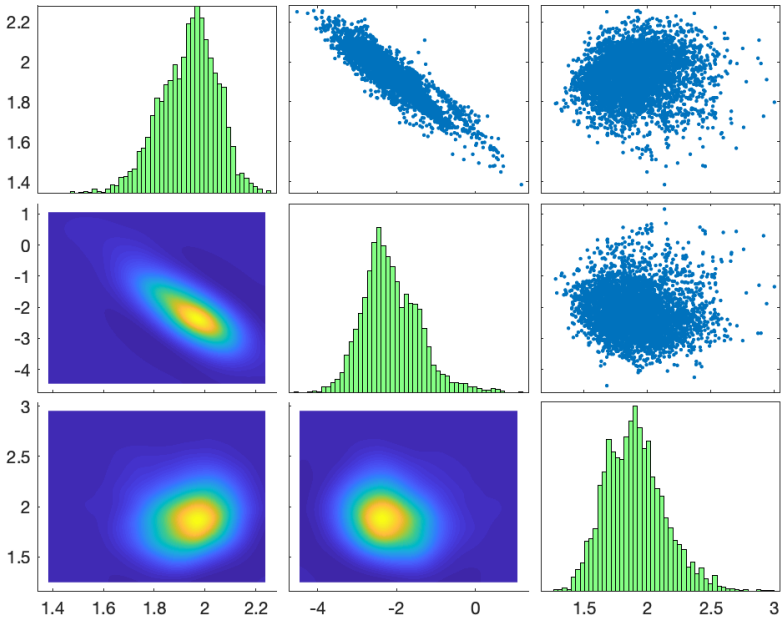

Figure 1: Results from BASIS.

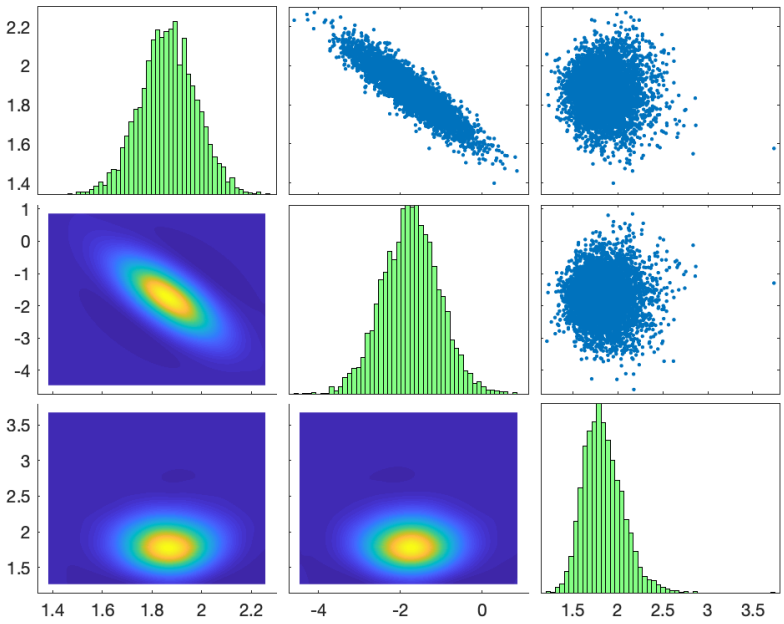

Figure 2: Results from smTMCMC.