

OpenBDLM: An open-source Matlab software for
Structural Health Monitoring using Bayesian
dynamic linear models

Ianis Gaudot, Luong H. Nguyen, James-A. Goulet
Polytechnique Montreal

July 6, 2018

Contents

1	What OpenBanana does ?	7
2	Install OpenBanana	9
3	Data processing	11
3.1	Purpose	11
3.2	Input data format	11
3.3	Output data format	11
3.4	Merging timestamps vector	11
3.5	Data processing functions	11
4	Model building	13
4.1	Purpose	14
4.2	Model class	14
4.3	Dependencies	14
4.3.1	Observed covariate	14
4.3.2	Hidden covariate	14
4.4	Model components	14
4.4.1	Local level	14
4.4.2	Local trend	14
4.4.3	Local acceleration	14
4.4.4	Local level compatible trend	14
4.4.5	Local level compatible acceleration	14
4.4.6	Local trend compatible acceleration	14
4.4.7	Periodic	14
4.4.8	Dynamic regression	14
4.4.9	Static kernel regression	14
4.4.10	Dynamic kernel regression	14
4.4.11	Residual - first order autoregressive	14
4.5	Model building functions	14
5	Model parameters learning	15
5.1	Purpose	15
5.2	Posterior	15

5.2.1	Prior	15
5.2.2	Likelihood	15
5.3	Model parameters bounds and transformed spaces	15
5.3.1	Logarithmic transformation	15
5.3.2	Sigmoid transformation	15
5.4	Gradient-based optimization techniques	15
5.4.1	Newton-Raphson approach	15
5.4.2	Stochastic Gradient Ascent approach	15
5.5	Constrain model parameters between each others	15
5.6	Model parameters learning functions	15
6	Hidden states estimation	17
6.1	Purpose	17
6.2	Kalman equations	17
6.3	UD computations	17
6.4	Filtering	17
6.5	Smoothing	17
6.6	Hidden states estimation functions	17
7	Data simulation	19
7.1	Purpose	19
7.2	Data simulation functions	19
8	Model validation	21
8.1	Purpose	21
8.2	Prediction capacity	21
8.3	Posterior model parameters covariance matrix analysis	21
8.4	Posterior state covariance matrix analysis	21
8.5	Residual component analysis	21
8.6	Model validation functions	21
9	Visualization tools	23
9.1	Purpose	23
9.2	Data availability plots	23
9.3	Hidden states plots	23
9.4	Export figures options	23
9.5	Visualization tools functions	23
10	Version control	25
10.1	Purpose	25
10.2	Version control functions	25
11	Examples	27
11.1	Example 1	27
11.2	Example 2	27
11.3	Example 3	27

<i>CONTENTS</i>	5
11.4 Example 4	27
12 List of functions	29
13 Older versions	31
14 Last changes	33
14.1 Data loading	33
14.2 Data simulation	33
14.3 Create a configuration file	34
14.4 Export figures	34
14.5 Batch processing	34

Chapter 1

What OpenBDLM does ?

Chapter 2

Install OpenBDLM

Chapter 3

Data processing

3.1 Purpose

3.2 Input data format

3.3 Output data format

3.4 Merging timestamps vector

3.5 Data processing functions

Chapter 4

Model building

4.1 Purpose

4.2 Model class

4.3 Dependencies

4.3.1 Observed covariate

4.3.2 Hidden covariate

4.4 Model components

4.4.1 Local level

4.4.2 Local trend

4.4.3 Local acceleration

4.4.4 Local level compatible trend

4.4.5 Local level compatible acceleration

4.4.6 Local trend compatible acceleration

4.4.7 Periodic

4.4.8 Dynamic regression

4.4.9 Static kernel regression

4.4.10 Dynamic kernel regression

4.4.11 Residual - first order autoregressive

4.5 Model building functions

Chapter 5

Model parameters learning

5.1 Purpose

5.2 Posterior

5.2.1 Prior

5.2.2 Likelihood

5.3 Model parameters bounds and transformed spaces

5.3.1 Logarithmic transformation

5.3.2 Sigmoid transformation

5.4 Gradient-based optimization techniques

5.4.1 Newton-Raphson approach

5.4.2 Stochastic Gradient Ascent approach

5.5 Constrain model parameters between each others

5.6 Model parameters learning functions

Chapter 6

Hidden states estimation

6.1 Purpose

6.2 Kalman equations

6.3 UD computations

6.4 Filtering

6.5 Smoothing

6.6 Hidden states estimation functions

Chapter 7

Data simulation

7.1 Purpose

7.2 Data simulation functions

Chapter 8

Model validation

8.1 Purpose

8.2 Prediction capacity

8.3 Posterior model parameters covariance matrix analysis

8.4 Posterior state covariance matrix analysis

8.5 Residual component analysis

8.6 Model validation functions

Chapter 9

Visualization tools

9.1 Purpose

9.2 Data availability plots

9.3 Hidden states plots

9.4 Export figures options

9.5 Visualization tools functions

Chapter 10

Version control

10.1 Purpose

10.2 Version control functions

Chapter 11

Examples

11.1 Example 1

Simulated data with: one time series, one model class, $\{[12\ 31\ 41]\}$.

11.2 Example 2

Simulated data with: two time series with dependencies, one model class, $\{[11\ 41], [11\ 31\ 41]\}$.

11.3 Example 3

Simulated data with one time series, two model classes, $\{[21\ 31\ 41]\}$ and $\{[12\ 31\ 41]\}$.

11.4 Example 4

Real data with one time series, one model class, $\{[12\ 52\ 41]\}$.

Chapter 12

List of functions

Chapter 13

Older versions

Chapter 14

Last changes

This new version now includes the data loader and the data simulation tools. It also includes additionnal tools to export the figures in several format (.tex, .png, .pdf). This version also supports batch processing (loop over input files, see the script demo.1.m).

The model parameters and hidden state estimation functions are (almost) identical compared to the previous GitHub version.

WARNING: the 'data' structure completely changed (cell array instead of simple array) then previous configuration file are NOT readable with this version. You have to initialize a new project using the interactive tool, and then copy/paste manually your previously learned initial hidden states and/or model parameters values.

Note that the software will only work properly if you run it from the same folder that contains the file BDLM.m. As usual, always put all the folders in your Matlab path before running the program.

14.1 Data loading

Data loader is accessible via interactive tool (option '0'). See the DataLoader.m function to change data loading option (handle quantity of allowed missing data, handle tolerance between timestamp, etc...).

14.2 Data simulation

Once a project has been created, data simulation is now possible through option '16'. A new project is created.

14.3 Create a configuration file

Option '0' builds a configuration file by default. At any time, option '17' export current model in configuration file format.

14.4 Export figures

The program can now save figures in tex, png and pdf format. See the `plotEstimation.m` function to change export options. When pdf export is done, the program builds a single pdf file containing all the figures.

14.5 Batch processing

Batch processing allows you to plan a succession of tasks in advance, and let them run during the night (or whenever you want). All the planned command have to be written in ASCII input files.

See `demo_1.m` for an example of batch processing. `demo_1.m` sequentially creates 3 simulated dataset.

WARNING: the beta version of this tool uses global variables (named 'isAnswersFromFile' 'AnswersFromFile' 'AnswersIndex'), so always clean your workspace by typing 'clear -global isAnswersFromFile AnswersFromFile AnswersIndex' before and after using batch processing.