Usage of parameter estimation tool for offset mixture of Gaussian distributions and Laplace distributions

Masato Fujita *

October 15, 2013

Contents

1 Introduction

The enclosed software **OffLatDistEst.jar** implements the algorithm introduced in Reference [1]. This paper summarizes the usage of this software.

We first define our terminology. We will consider the sum of a discrete random variable with a random variable following the finite mixture of an arbitrary number of 1-dimensional Gaussian distributions and Laplace distributions with

*Electronic Navigation Research Institute

zero mean. We call it an *offset mixture distribution* (of Gaussian distributions and Laplace distributions) in this paper. The discrete random variable corresponds to the offset of the aircraft. On the other hand, the finite mixture distribution represents the lateral navigation performance of the aircraft. More precisely, it is the total system error (TSE) defined in Reference [2].

The probability density function of an offset mixture distribution is given by the following equation.

$$p(x|\boldsymbol{\omega}, \boldsymbol{\pi}, \boldsymbol{\sigma}, \boldsymbol{\lambda}) = \sum_{l=1}^{L} \omega_l \left(\sum_{k=1}^{m} \pi_k \mathcal{N}(x - o_l, \sigma_k) + \sum_{k=1}^{n} \pi_{m+k} \mathcal{D}(x - o_l, \lambda_k) \right) \quad (1)$$

Here, $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_K)$ are finite non-negative numbers satisfying $\sum_{k=1}^{K} \pi_k = 1$, where $K = m + n$. $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_m)$ and $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_n)$ denote positive numbers. $\boldsymbol{o} = (o_1, o_2, \ldots, o_L)$ are real numbers. $\boldsymbol{\omega} = (\omega_1, \omega_2, \ldots, \omega_L)$ are non-negative numbers satisfying $\sum_{l=1}^{L} \omega_l = 1$. We call ω_l an *offset mixing coefficient* in this paper. The notation $\mathcal{N}(x, \sigma)$ denotes the probability density function of a 1-dimensional Gaussian distribution with zero mean, and $\mathcal{D}(x, \lambda)$ denotes the probability density function of a Laplace distribution with zero mean.

The functions of the software are as follows:

1. Generate pseudo-random samples following the given offset mixture distribution model. (Sample generation function)
2. Estimate the parameter values of distribution model from the given data set. (Model Parameter estimation function) The EM algorithm and the variatioal Bayesain method are both implemented.
3. Estimate the applied offset when a offset mixture distribution model is given. (Offset estimation function)

We discussed about offset mixture distributions, but this software can handle non-offset mixture distributions, namely normal mixture distributions, as well. Section 2 treats the non-offset case, and Section 3 describes the offset case.

The source codes enclosed with the software also contain some other useful classes. The program can be used as a library as well. See the eclosed Javadoc for the structure of these codes.

2 Non-offset case

We discuss about the non-offset case in this section. The probability distribution model considered in this section is given by the following equation. It is a spacial case of the equation (2). It is $L = 1$ and $\boldsymbol{\omega} = \{0\}$ in this case.

$$p(x|\boldsymbol{\pi}, \boldsymbol{\sigma}, \boldsymbol{\lambda}) = \sum_{k=1}^{m} \pi_k \mathcal{N}(x, \sigma_k) + \sum_{k=1}^{n} \pi_{m+k} \mathcal{D}(x, \lambda_k) \quad (2)$$

2.1 Sample generation function

The software can generate psudo-random numbers. The usage of this fuction is as folllows:

Usage

java -jar OffLatDistEst.jar u nde rg <Path to distribution definition file> <Number of generated random numbers> <Path to output file>

The magic word 'rg' means 'random generation.'

The $1,000$ samples following the distrbution defined in the file of 'tmp/NDE.properties' are generated in the folllowing example. The results are recorded in the file of 'tmp/NDEdata.csv.'

Example

java -jar OffLatDistEst.jar u nde rg tmp/NDE.properties 1000 tmp/NDEdata.csv

The format employed in the distribution definition file is the '.properties' format intoroduced in Reference [3]. Each line in a .properties file normally stores a single property. Several formats are possible for each line, including key=value, key = value, key:value, and key value. The description $m = 1$ means the value of the variable 'm' is equal to one.

The variable m denotes the number of Gaussian components of mixture distribution. The variable n denotes the number of Laplace components defined in the mixture distribution. The variable 'pi_Ni' is the i-th mixing coefficient of the Gaussian components. Here, i denotes the positive integer. The variable 'pi_DEi' is the i-th mixing coefficient of the Laplace components. The variables 'sigma$i$' and 'lambda$i$' are the standard deviation of the i-th Gaussian component and the scale parameter of the i-th Laplace component, respectively. The following is an example of the distribution definition files. The following example means that $m = 1$, $n = 1$, $\pi_1 = 0.95$, $\pi_2 = 0.05$, $\sigma_1 = 1$ and $\lambda_1 = 10$ in the equation (2).

Example

m=1
n=1
pi_N1=0.95
sigma1=1
pi_DE1=0.05
lambda1=10

The output of the program in the above example situation is given in the following format. Each line stores a single sample. The first number in the line is the generated random sample and the second one shows that the number is generated from which component. The value 0 means the first component of the mixture. The value 1 means the second component. The first component

corresponds to the first Gaussian component and the second componet corresponds to the first Laplace component in this example. When two Gaussian distributions and one Laplace distribution are mixtured, the first component corresponds to the first Gaussian component and the second componet corresponds to the second Gaussian component, and the last one is the remaining Laplace component.

Example

```
-0.6748209837410951,0.0
-0.4277818997347839,0.0
-28.151872997991628,1.0
0.8039747832470926,0.0
-0.2489646290545688,0.0
```

2.2 Model parameter estimation function

The following command is used whenso as to estimate the model parameters from the data files

Usage

java -jar OffLatDistEst.jar u nde em/vb <Path to initial condition setting file> <Path to data file> <Path to output file>

The description 'em/vb' in the above usage means 'em' or 'vb.' The EM algorithm is used when 'em' is chosen and the variational Bayesian method is emplyed when 'vb' is chosen.

The initial condition is saved in 'tmp/NDEEM.properties,' and the data file is 'tmp/NDEdata.csv' in the following example. The EM algorithm is employed in this example.

Example

java -jar OffLatDistEst.jar u nde em tmp/NDEEM.properties tmp/NDEdata.csv tmp/NDEEM_out.properties

The input data file is assumed to follow the camma-separated csv format. The first datum in each line is a sample datum, and the remainings are ignored. Both input and output files employ the '.properties' format. The variables which should be defined depends on the employed estimation algorithm. We first consider the EM algorithm case, then treat the variational Bayesian method case afterward.

2.2.1 EM algorithm case

In the EM algorithm, the initial values of model parameters in the equation (2) should be given. The format of the initial condition setting file is the same as the format of the distribution definition file defined in Section 2.1. The format of the

output file is also almost same as the distribution definition file. A new variable 'logLiklihood' is defined in the output file other than the variables defined in the distribution definition file. It is the log likelihood value.

2.2.2 Variatioal Bayesian method case

When the variational Bayesian methodlogy is employed, the prior distribution of the model parameters should be defined.

We define the sequence of real numbers $\boldsymbol{\eta} = (\eta_1, \eta_2, \ldots, \eta_K)$ by the following equation. Set $K = m + n$.

$$\eta_k = \begin{cases} \dfrac{1}{\sigma_k^2} & (1 \leq k \leq m) \\ \dfrac{1}{\lambda_{k-m}} & (m+1 \leq k \leq K) \end{cases} \tag{3}$$

This paper introduces the variational Bayesian methodology for finding an approximation of the posterior distribution of $\boldsymbol{\pi}$ and $\boldsymbol{\eta}$. The prior distribution of $\boldsymbol{\pi}$ is a Dirichlet distribution $\mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha_0})$, where $\boldsymbol{\alpha_0} = (\alpha_{0,1}, \alpha_{0,2}, \ldots, \alpha_{0,K})$, and the prior distribution of η_k is a Gamma distribution $\mathrm{Gam}(\eta_k|a_{0,k}, b_{0,k})$. The posterior distributions of $\boldsymbol{\pi}$ and η_k are again Dirichlet and Gamma distributions in our variational Bayesian algorithm.

The initial condition file defines the parameters of the prior distributions. The variables m and n are the numbers of Gaussian and Laplace components, respectively. The 'alpha$i$' is the parameter of Dirichlet distribution. The 'ai' and 'bi' are the parameters of i-th Gamma distribution. The parameters are $m = 1$, $n = 1$, $\alpha_{0,1} = 0.95$, $\alpha_{0,2} = 0.05$, $a_{0,1} = 1$, $b_{0,1} = 1$, $a_{0,2} = 10$ and $b_{0,2} = 1$ in the following example.

Example

```
m=1
n=1
alpha1=0.95
a1=1
b1=1
alpha2=0.05
a2=10
b2=1
```

The format of the output file is similiar to that of the initial condition file. The output file gives the parameters of the posterior distributions. It also contains the model parameters of the mixture distribution defined in the equation (2). These are estimated by means of the MAP estimation. The format for defining the model parameters of the mixture distribution is the same as the format of the distribution definition file defined in Section 2.1. A new variable 'lowerbound' is the value defined in Reference [1].

3 Offset case

We consideer the case where offset is allowed. The distribution model considered in this section is given in the equation (1).

3.1 Sample generation function

The software can generate psudo-random samples. The usage of this fuction is as folllows:

Usage

java -jar OffLatDistEst.jar u **onde** rg <Path to distribution definition file> <Number of generated random numbers> <Path to output file>

The second argument was 'nde' in the non-offset case. However, the magic word 'nde' is replaced by 'onde' in this usage.

The $1,000$ samples following the distrbution defined in the file 'tmp/ONDE.properties' are generated in the folllowing example. The outputs are recorded in the file 'tmp/ONDEdata.csv.'

Example

java -jar OffLatDistEst.jar u onde rg tmp/ONDE.properties 1000 tmp/ONDEdata.csv

The format of the input file is almost the same as the case where offset is not allowed. The new variables 'L', 'offset$i$' and 'omega$i$' are introduced in this format. The variable 'L' corresponds to the number of possible offset which is denoted by L in the equation (1). The variables 'offset$i$' and 'omega$i$' are o_i and ω_i in the equation (1). The following example means that $m = 1$, $n = 1$, $\pi_1 = 0.95$, $\pi_2 = 0.05$, $\sigma_1 = 1$, $\lambda_1 = 10$ $L = 3$, $o_1 = 0$, $o_2 = 100$, $o_3 = 200$, $\omega_1 = 0.6$, $\omega_2 = 0.3$ and $\omega_3 = 0.1$ in the equation (1).

Example

m=1
n=1
pi_N1=0.95
sigma1=1
pi_DE1=0.05
lambda1=10
L=3
offset1=0
offset2=100
offset3=200
omega1=0.6
omega2=0.3
omega3=0.1

The output of the program in the above example situation is given in the following format. Each line stores a single sample just like the non-offset case. The first two numbers in the line are the same as the non-offset case. The last number is the applied offset.

Example

98.7234011320356,0.0,100.0
99.6129842355946,0.0,100.0
99.75406291903305,0.0,100.0
101.20332639212567,0.0,100.0
-1.8654843885172798,0.0,0.0
0.32696081002411587,0.0,0.0

3.2 Model parameter estimation function

The following command is used when the model parameters are estimated. from the data files

Usage

java -jar OffLatDistEst.jar u **onde** em/vb <Path to initial condition setting file> <Path to data file> <Path to output file>

The description 'em/vb' in the above usage means 'em' or 'vb.' The EM algorithm is used when 'em' is chosen and the variational Bayesian method is emplyed when 'vb' is chosen. Just like the case of random sample generation, the second argument becomes 'onde' and the remaining is the same as the non-offset case.

The initial condition is saved in 'tmp/ONDEEM.properties,' and the data file is 'tmp/ONDEdata.csv' in the following example. The EM algorithm is employed in this example.

Example

java -jar OffLatDistEst.jar u onde em tmp/ONDEEM.properties tmp/ONDEdata.csv tmp/ONDEEM_out.properties

3.2.1 EM algorithm case

The format of the initial condition setting file is the same as the format of the distribution definition file defined in Section 3.1. The format of the output file is also almost same as the distribution definition file with the variable 'logLiklihood.'

3.2.2 Variatioal Bayesian method case

We assume that the prior distribution of $\boldsymbol{\omega}$ in the equation (1) follows a Dirichlet distribution and the posterior distribution of $\boldsymbol{\omega}$ is again a Dirichlet distribution in our variational Bayesian method. The following is an example of the input file format. They are almost the same as the non-offffset case. The newly introduced variables are 'L,' 'offset$i$' and 'pi.' The notations 'L' and 'offset$i$' mean the same in the case of the random sample generation. The variables 'pi' are the parameters of Dirichlet distribution which is the prior distribution of $\boldsymbol{\omega}$.

Example

```
m=1
n=1
alpha1=8
a1=1
b1=1
alpha2=2
a2=10
b2=50
L=3
offset1=0
offset2=100
offset3=200
p1=5
p2=3
p3=2
```

The output file contains the parameters of the posterior distribution and the parameters of MAP estimation just like in the case of Section 2.2. The following is a example of the output file.

Example

#BaysienUpdate generated automatically in Tue Oct 15 13:09:51 JST 2013
#Tue Oct 15 13:09:51 JST 2013
lowerbound=-2548.1402570141263
lambda1=9.072036792841228
p3=93.5326860016927
p2=295.46685995538036
p1=620.0004540429268
b2=521.9377939187281
b1=469.2083825928237
L=3
sigma1=0.9936404536431874
alpha2=50.532592276366294
offset3=200.0
alpha1=958.4674077236335
n=1
offset2=100.0
m=1
offset1=0.0
pi_DE1=0.04918827435587518
pi_N1=0.9508117256441249
omega3=0.09198080119452555
omega2=0.29271059637711766
omega1=0.6153086024283567
a2=58.532592276366294
a1=476.23370386181676

3.3 Offset estimation function

This software estimates the applied offset using a distibution model (typeically, it is the result of the parameter estimation by means of the EM algorithm) or a posterior distribution (typically, it is the result of the parameter estimation by means of the variatioal Bayesia method). The following is the usage of this function.

Usage

java -jar OffLatDistEst.jar u onde oe d/b <Path to definition file of distribution model or posterior distribution> <Path to input data file> <Path to output file>

The algorithm employed in this ssoftware is found i [1]. The description 'd/b' means 'd' or 'b.' The distribution model is used when it is 'd' and the posterior distribution model is used otherwise. The followings are examples of usage.

Example

```
java -jar OffLatDistEst.jar u onde oe d tmp/ONDEEM_out.properties
tmp/ONDEdata2.csv tmp/ONDEoe1.csv
java -jar OffLatDistEst.jar u onde oe b tmp/ONDEVB_out.properties
tmp/ONDEdata2.csv tmp/ONDEoe2.csv
```

The following is the example of an input data file. It is generated by means of the random sample generation function.

Example

```
101.06098250370194,0.0,100.0
99.1370098918537,0.0,100.0
1.161038718131618,0.0,0.0
99.19302094520688,0.0,100.0
1.1862779484618524,0.0,0.0
1.7136672096592742,0.0,0.0
198.14413995307189,0.0,200.0
```

The following is the example of the output file. The first three columns are the same as the input file. The last three columns show the probability that the selected offset is o_i. For instance, as to the first data 101.06098250370194, the probability that the applied offset is $o_1 = 0$ is $3.161188293167 \times 10^{-7}$, the probability that the applied offset is $o_2 = 100$ is 0.9999996236513, and the probability that the applied offset is $o_3 = 200$ is $6.022983599509 \times 10^{-8}$.

Example

```
101.06098250370194,0.0,100.0,3.161188293167E-7,0.9999996236513,6.022983599509E-8
99.1370098918537,0.0,100.0,3.2393437356634E-7,0.9999996359809,4.00847172340E-8
1.161038718131618,0.0,0.0,0.999999897027,1.0297201515039E-7,4.368167090427E-13
99.19302094520688,0.0,100.0,3.071036901020E-7,0.9999996544137,3.848255633989E-8
1.1862779484618524,0.0,0.0,0.9999998936267,1.063727515537E-7,4.512206125519E-13
1.7136672096592742,0.0,0.0,0.9999997587778,2.4122108828109E-7,1.0213179144518E-12
198.14413995307189,0.0,200.0,5.84209355022E-11,2.074112221570E-6,0.9999979258293
```

4 License

Copyright (c) 2013 Masato Fujita

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in

all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

5 Package enclosed to the software

- COLT
  `http://acs.lbl.gov/software/colt/`
- Apache Commons Logging
  `http://commons.apache.org/proper/commons-logging/`
- Apache Commons Math
  `http://commons.apache.org/proper/commons-math/`

References

[1] Fujita, M. (2013). Estimation of Navigation Performance and Offset by the EM Algorithm and the Variational Bayesian Methods, *Advances and applications in statistics*, 35(1), 1–27.

[2] International Civil Aviation Organization(ICAO) (2008). *Performance-based Navigation (PBN) manual*. 3rd eds.: International Civil Aviation Organization.

[3] `http://docs.oracle.com/cd/E23095_01/Platform.93/ATGProgGuide/html/s0204propertiesfileformat01.html`