

OpSiMorE/FIEM

OpSiMorE is a web repository for the contributions of the OpSiMorE team.

Author: [Gersende Fort](#)

Affiliation: [CNRS](#) and [Institut de Mathématiques de Toulouse](#), France

Created in: July 2020

FIEM : Fast Incremental Expectation Maximization

Here are the codes associated to the paper "Fast Incremental Expectation Maximization for non-convex finite-sum optimization: non asymptotic convergence bounds" available from [this web page](#); author : G. Fort, P. Gach and E. Moulines

In this repository, the subdirectory *ToyExample* contains the MATLAB codes for the toy example (section 4); the subdirectory *MNIST* contains the MATLAB codes for the more challenging example using the MNIST data set.

How to use the matlab files available in the subdirectory *ToyExample* ?

- Statistical model : n observations (Y_1, \dots, Y_n) . For the value $\theta \in \mathbb{R}^q$ of the parameter: conditionally to (Z_1, \dots, Z_n) , the random variables Y_i are independent with distribution

$$\mathcal{N}_y(AZ_i, I);$$

the random variables (Z_1, \dots, Z_n) are independent with the same distribution

$$\mathcal{N}_p(X\theta, I).$$

The matrices X and A are known.

- Statistical analysis: the goal is the estimation of the parameter θ through the minimum of the criterion

$$-\frac{1}{n} \log g(Y_1, \dots, Y_n; \theta) + \frac{\nu}{2} \|\theta\|^2$$

where $g(y_1, \dots, y_n; \theta)$ denotes the distribution of the vector (Y_1, \dots, Y_n) . In this toy example, the solution exists, is unique and is even explicit (see section 4.1).

- Description of the file *SampleData.m*
 - The variables :
 - n : the size of the data set
 - dim_theta : the size of the vector θ
 - dim_Y : the size of each vector Y
 - dim_Z (or p in the paper): the size of each hidden variable Z
 - $theta_true$: the value of the parameter used to generate the observations as described by the above statistical model. The entries of $theta_true$ are sampled uniformly in $[-5, 5]$ and then 40% of these entries are set to zero (they are chosen at random).
 - The columns of the $dim_Y \times dim_Z$ matrix A are sampled from a stationary AR(1) with variance 1 and coefficient $\rho = 0.8$
 - The columns of the $dim_Z \times dim_theta$ matrix X are sampled from a stationary AR(1) with variance 1 and coefficient $\rho = 0.9$
 - The output of the function: the function creates a file *DataLong.mat* which contains the observations Y_1, \dots, Y_n , the matrices X and A , and the parameter $theta_true$.
- Description of the file *FIEM_Gamma.m*
 - The call
$$>> FIEM_Gamma$$
will launch the estimation of the parameter θ by the optimized FIEM algorithm described in Section 2.3.4 of the paper. The user is invited to choose different design parameters
 - Choices by the user
 - During the run of *FIEM_Gamma* graphical controls can be displayed (see the description below). The user is invited to choose if it accepts or not the display.
 - The data are loaded directly in the code. See the line
$$>> load Data.mat$$

Data.mat contains the matrices A and X which define the statistical model; and the n observations stored in a $dim_Y \times n$ matrix called *Ymatrix*.

- *upsilon*: the numerical value of the regularization parameter in the penalty term $\nu \|\theta\|^2 / 2$. The default value is 0.1
- *NbrMC*: independent runs of FIEM can be launched through a single call to *FIEM_Gamma*; they will be run with the same values of the design parameters and will only differ by the random mechanism for the selection of the examples. The user is invited to indicate this number. The default value is 1e3.
- *Kmax*: the total length of a FIEM path. The default value is $20n$ where n is the number of observations, i.e. 20 epochs.
- The learning rate is constant over the iterations of the path: the two strategies provided in the paper (see Proposition 5 and Proposition 6) are possible, the user can choose between resp. *rate* $n^{(2/3)}$ and *rate* $n^{(1/2)}$. Both of these strategies depend on two parameters *mu* and *lambda* and the user is invited to specify these values; the default ones are resp. 0.25 and 0.5. For any other strategies, the user can modify the value of the variable *gamma_gfm* in the definition of *gamma_grid* directly in the code.
- The initial value \hat{S}^0 of the FIEM path can be specified: either it is chosen at random by sampling the entries as standardized independent Gaussian

variables; or it is read in a file *InitStat.mat* - this file has to contain the variable *Sinit* of size $dim_theta \times 1$.

- Two sequences of length $Kmax$ containing indices in the range $\{1, \dots, n\}$ have to be selected: they indicate the examples used in the updating mechanism of the auxiliary variable \tilde{S}^{k+1} and the ones used in the updating mechanism of the statistics \hat{S}^{k+1} . Here again, the user is invited to choose between (i) a random selection; (ii) a choice stored in the file *RandomIndex.mat* (which contains the $NbrMC \times Kmax$ matrix *RandomIndexImatrix*) and *RandomIndexFIEM.mat* (which contains the $NbrMC \times Kmax$ matrix *RandomIndexJmatrix*).
- Finally, by default, the *optimized FIEM* is implemented (see section 2.3.4); the computation of the leverage coefficient λ_{k+1} is done through the call to the function *findlambda.m*. The user can choose other strategies by modifying directly in the code the value of the variable *Coeff*. For example, *Coeff = 1* corresponds to the original FIEM algorithm (see section 2.3.3), and *Coeff = 0* corresponds to Online EM (see section 2.3.1)

○ The outputs: a file *StoreCoeffopt.mat* containing

- *StoreCoeff*: the $NbrMC \times Kmax$ values of the leverage coefficient λ_k
- *FielH*: a $NbrMC \times Kmax$ matrix containing the squared norm of $H_{k+1} = (\hat{S}^{k+1} - \hat{S}^k)/\gamma_{k+1}$.
- *ExpFieldH*: a $NbrMC \times Kmax$ matrix containing the squared norm of the mean field $h(\hat{S}^k)$.
- *ErrorTheta*: a $NbrMC \times GG$ matrix containing the squared norm of the difference $\theta^{k+1} - \theta_{opt}$ where θ_{opt} is the unique optimum of the objective function, and is explicit in this toy example. This error is evaluated at GG time steps in the range $\{1, \dots, Kmax\}$.

○ The displayed graphs:

- Figure1 and Figure2: the squared norm of $\theta^{k+1} - \theta_{opt}$ along the path, with a zoom on the first iterations.
- Figure3: the value of the leverage coefficient λ_k along a FIEM path.
- Figure4: the squared norm of H_{k+1} along a FIEM path.
- Figure5: the evolution of each components of the FIEM statistic \hat{S}^k along the FIEM path.

○ The function *findlambda.m* is called by *FIEM_Gamma* and computes the optimal leverage coefficient λ_k - see section 2.3.4

○ The functions *findcstar_1.m* and *findcstar_2.m* are called by *FIEM_Gamma* and compute the constant learning rate γ_{k+1} resp. provided in Proposition 5 and Proposition 6.

How to use the matlab files available in the subdirectory *ToyExample* ?