

## EE613 - Linear regression - Exercises - Nov. 4, 2015

The main folder contains seven examples `demo_LS01.m`, `demo_LS_polFit01.m`, `demo_LS_polFit02.m`, `demo_LS_nullspace01.m`, `demo_LS_weighted01.m`, `demo_LS_recursive01.m` and `demo_LS_IRLS01.m`. These codes can be run either from Matlab or from GNU Octave. First run the examples, visualize the results and try to change the parameters.

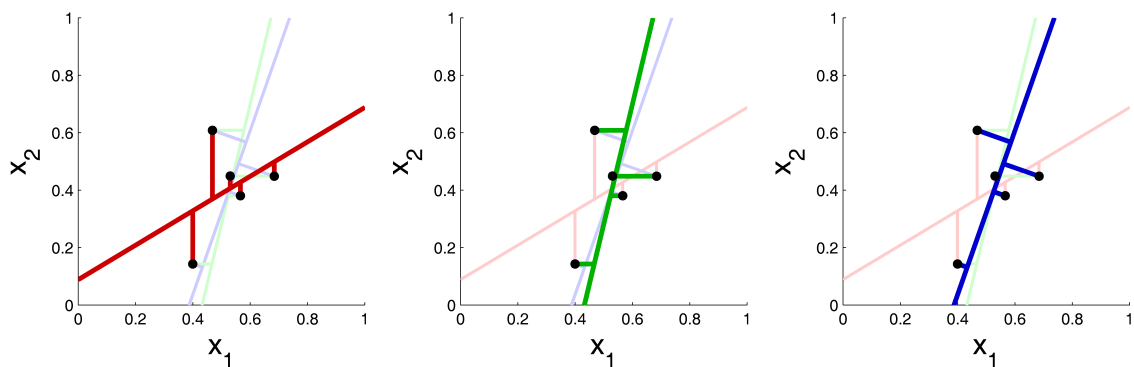
### Exercise 1: Efficient computation of least squares

- By taking `demo_LS01.m` as an example, implement least squares with these different techniques:
  1. By explicitly inverting  $\mathbf{X}^T \mathbf{X}$  in the computation of  $\hat{\mathbf{A}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}$
  2. By using Cholesky decomposition (function `chol`)
  3. By using singular value decomposition (function `svd`)
- For each computation technique, evaluate the computation time (functions `tic/toc`). In order to facilitate the comparison, use a set of dimensions  $D^x$ ,  $D^o$  and  $N$  that helps at distinguishing the different techniques. Try also with  $N < D^x$  (by verifying the condition number before inverting matrices). Which conclusions can you draw from the use of these different techniques?

### Exercise 2: Recursive least squares

- The example `demo_LS_recursive01.m` shows that the Sherman-Morrison-Woodbury formula can be exploited to update a least squares estimate when a new group of datapoints become available. Based on this example, write a program that iteratively updates the least squares estimate one datapoint at a time, by starting from the initial estimates  $\mathbf{A} = \mathbf{0}$  and  $\mathbf{B}^{-1} = 10^{10} \mathbf{I}$ .
- Show that the final estimate is the same as the batch estimate using the whole dataset.
- Show that the recursive least squares does not require the use of a matrix inversion operator (i.e., only scalar division).

### Exercise 3: Various forms of line fitting



- Fit a line to a small number of datapoints by considering the three minimization problems shown in the above figure, respectively displayed in red, green and blue colors (the segments connected to the points depict the errors to minimize).