

---

## LAB 3: Sparsity-based learning

---

- This lab is about feature selection within the framework of sparsity based regularization, using elastic net regularization.
- The aim of the lab is to play with the libraries and to get a practical grasp of what we have discussed in class.
- Follow the instructions below.

### Goal:

This lab is divided in two parts depending of their level of complexity (**Beginner**, **Intermediate**). Your goal is to complete entirely, at least, one of the two parts.

### Setup instructions:

#### *PC Login (LABS)*

If you don't have a laptop with MATLAB, use a PC in SW1 and SW2, DIBRIS

- Username: regml
- Password: RegML2016

#### *Running MATLAB*

1. Double-click on Windows 7 icon on the desktop
2. Run MATLAB from Start → All Programs → MATLAB → R2013a → Matlab executable
3. Download the file `regml2016_lab3.zip` from the syllabus on the course website (<http://lcs1.mit.edu/courses/regml/regml2016/#syllabus>), extract it and add all the sub-folders to the MATLAB path. This file includes all the code you need!

**Important notice:** All files on desktop PCs will be erased at the end of each lab session. If you want to keep your files, you will have to save them to other locations (e.g. personal cloud storage services).

## Toy problem

We focus on a regression problem where the target function is linear. We will consider synthetic data generated (randomly sampled) according to a given probability distribution and affected by noise. You will have the possibility of controlling the sizes of training and test sets, data dimensionality and number of relevant features.

**NOTE:**

In the code we use a different notation from what you have seen in the classes. The minimized functional is:

$$\min_{\beta \in \mathbb{R}^p} \|X\beta - Y\|^2 + \text{L1\_par} \|\beta\|_1 + \text{L2\_par} \|\beta\|_2^2.$$

In addition, in some MATLAB scripts the sparsity parameter `L1_par` can be also referenced as `tau`, and the smoothing parameter `L2_par` as `smooth_par`.

Download the file `regml2016_lab3.zip`, extract it and add all the sub-folders to the path. This file includes all the code you need!

## PART I: Beginner

### Overture: Warm up

Run the file `gui_1112.m` and the GUI will start. Have a look at the various components.

- Generate a training set with the default parameters.
- Press the `run` button to start a training phase with the selected `L1_par` and `L2_par` parameters and perform testing.
- Change values for `L1_par` and `L2_par` and have a look at test error and number of selected variables:
  - First set `L2_par = 0` and vary `L1_par`, trying to obtain a sparser or denser solution. What do you notice?
  - Repeat the experiment with a `L2_par > 0`. How do test error and number of selected features vary?
- Now select KCV for `L1_par` tuning and observe the KCV error curve.

### Interlude: The Geek Part

Back on the MATLAB shell, have a look to the content of directory `./PROXIMAL_TOOLBOXES/L1L2_TOOLBOX`. There you will find, among the others, the code for commands `l1l2_algorithm` (used for variable selection), `l1l2_kcv` (used for model selection with `kcv` or `loo`), `l1l2_pred` (for prediction on a test set).

For more information about the parameters and the usage of those scripts, type:

```
1 >> help l1l2_algorithm
2 >> help l1l2_kcv
3 >> help l1l2_pred
```

Finally, you may want to have a look at file `l1l2_demo_simple.m` for a complete example of analysis.

### Allegro con brio: Analysis

Carry out the following experiments either with the GUI, when it is possible, by personalizing the file `demo_1112.m` or by writing appropriate scripts.

i) *Prediction*: Considering elastic net regularization, observe how the training and test errors change

- Depending on the variations of the (cross-validated) regularization parameter  $\tau$  associated with the  $\ell_1$ -norm.

- When we change (increase or decrease) the regularization parameter  $\lambda$  associated with the  $\ell_2$ -norm. *Note:  $\lambda$  is defined at line 70.*
- The training set size  $N$  (line 62) grows (for instance, `[10,100,1000]`, as long as MATLAB supports you!).
- The amount of noise (`err_y`, line 67) on the generated data grows (the test set is generated with the same parameter of the training).

Change one parameter at a time!

ii) *Selection*: Considering elastic net regularization, observe how the number and values of non-zero coefficients in the solution change

- Depending on the variations of the (cross-validated) regularization parameter  $\tau$  associated with the  $\ell_1$ -norm.
- When we change (increase or decrease) the regularization parameter associated with the  $\ell_2$ -norm.
- The training set size grows.
- The amount of noise on the generated data grows.

iii) *Large  $p$  and small  $n$* : Perform experiments similar to those above changing  $p$  (dimensionality of the points),  $n$  (number of training points) and  $s$  (number of relevant variables)

- Set  $p \gg n$  and  $s > n$
- Set  $p \gg n$  and  $s < n$

## PART II: Intermediate

### Crescendo: Data standardization

Consider the classification dataset given in `part3-data.mat` (use the scripts and NOT the gui):

- Use `l1l2_algorithm` to analyze the features selected with different values of the regularization parameters.
- Tune the sparsity parameter `tau` to select only one variable. Is there another variable that can provide a better solution? (*hint*: only the first ten columns of `x` are correlated with `y`).
- Can you figure out why the selected variable is not the one that you would expect? (*hint*: Analyse the correlation between the columns of `x` and `y` and the ranges of the columns of `x`, e.g. with `imagesc(X(:,1:10)); colorbar`).