1 8th of November 2018 — A. Frangioni

1.1 Good practice of designing a project

In this lecture we focused on how to implement the gradient method inn Matlab and some good hints were underlined:

- when we have to implement a function the first question we need to ask ourselves is: "how many parameters should the function take?";
- the interface needs to be designed, i.e. in the case of the gradient method we would like the user to be able to run it altough he might not know what a good starting point could be;
- there might be some parameters with the same sematic of hyperparameters (using machine learning terminology), so they need to be adjusted in order to specify the algorithm;
- another thing that should be taken into consideration is that sometimes, when measuring the number of function calls we need to count also the calls to functions inside a single step;
- it's important to check the conditions that should be satisfied by the input data for the theortical coherence. In case of not valid input an error message should be returned;
- another important thing to be done is building a set of test functions, possibly limit case for the problem, to see how the algorithm works;
- `diff(f, x)` calculates the gradient of the function f in the variable x. If we want to get a simplified version, we may run `simplify(diff(f,x))`;
- a very nice tool to compute derivatives for arbitrary functions is **Adigator** (click here).