

# Computational Math notes

Marco Natali

# INDICE

1	INTRODUCTION	4
---	--------------	---

## ELENCO DELLE FIGURE

# 1 | INTRODUCTION

In Computational Math course we will get a general understanding of several different classes of numerical algorithms and their underlying mathematical principles to be also able to actually implement, debug, and tune a few of them.

This course is for ordinary AI expert who needs a good hands-on knowledge of mathematics without being a mathematician and also all concept introduced are related with optimization and numerical analysis.

Huge amounts of data is generated and collected, but one has to learn it in order to use it, so we take something big and unwieldy and produce something small, called *model* that can be used to predict successive data output.

A mathematical model should have 3 properties:

1. accurate (describes well the process at hand)
2. computationally inexpensive (gives answers rapidly)
3. general (can be applied to many different processes)

It is typically impossible to achieve all these three properties so we have to find, within the family of usually infinitely many model, the one that better represent our phenomenon.

In other words the model is parametric, so we have to learn the right values of parameters that will provide the best representation for our phenomenon and also this is *fitting*, and it is clearly some sort of optimization problem, where also solving the fitting problem is typically the computational bottleneck.

However,  $ML \gg fitting$ , infact fitting minimizes training error instead ML aims at minimizing test error that will minimize our risk and generalization error.