

1-Motivation-and-Introduction

April 4, 2016

0.0.1 Abstract

ROOT is a software framework for data analysis and I/O: a powerful tool to cope with the demanding tasks typical of state of the art scientific data analysis. Among its prominent features are an advanced graphical user interface, ideal for interactive analysis, an interpreter for the C++ programming language, for rapid and efficient prototyping and a persistency mechanism for C++ objects, used also to write every year petabytes of data recorded by the Large Hadron Collider experiments. This introductory guide illustrates the main features of ROOT which are relevant for the typical problems of data analysis: input and plotting of data from measurements and fitting of analytical functions.

1 1 Motivation and Introduction

Welcome to data analysis!

Comparison of measurements to theoretical models is one of the standard tasks in experimental physics. In the most simple case, a “model” is just a function providing predictions of measured data. Very often, the model depends on parameters. Such a model may simply state “the current I is proportional to the voltage U ”, and the task of the experimentalist consists of determining the resistance, R , from a set of measurements.

As a first step, a visualisation of the data is needed. Next, some manipulations typically have to be applied, e.g. corrections or parameter transformations. Quite often, these manipulations are complex ones, and a powerful library of mathematical functions and procedures should be provided - think for example of an integral or peak-search or a Fourier transformation applied to an input spectrum to obtain the actual measurement described by the model.

One specialty of experimental physics are the inevitable uncertainties affecting each measurement, and visualisation tools have to include these. In subsequent analysis, the statistical nature of the errors must be handled properly.

As the last step, measurements are compared to models, and free model parameters need to be determined in this process. Below you will find an example of a function (model) fit to data points. Several standard methods are available, and a data analysis tool should provide easy access to more than one of them. Means to quantify the level of agreement between measurements and model must also be available. Quite often, the data volume to be analyzed is large - think of fine-granular measurements accumulated with the aid of computers. A usable tool therefore must contain easy-to-use and efficient methods for storing and handling data.

In Quantum mechanics, models typically only predict the probability density function (“pdf”) of measurements depending on a number of parameters, and the aim of the experimental analysis is to extract the parameters from the observed distribution of frequencies at which certain values of the measurement are observed. Measurements of this kind require means to generate and visualize frequency distributions, so-called histograms, and stringent statistical treatment to extract the model parameters from purely statistical distributions.

Simulation of expected data is another important aspect in data analysis. By repeated generation of “pseudo-data”, which are analysed in the same manner as intended for the real data, analysis procedures can be validated or compared. In many cases, the distribution of the measurement errors is not precisely known, and simulation offers the possibility to test the effects of different assumptions.

A powerful software framework addressing all of the above requirements is ROOT, an open source project coordinated by the European Organisation for Nuclear Research, CERN in Geneva.

ROOT is very flexible and provides both a programming interface to use in own applications and a graphical user interface for interactive data analysis. The purpose of this document is to serve as a beginners guide and provides extendable examples for your own use cases, based on typical problems addressed in student labs. This guide will hopefully lay the ground for more complex applications in your future scientific work building on a modern, state-of the art tool for data analysis.

This guide in form of a tutorial is intended to introduce you quickly to the ROOT package. This goal will be accomplished using concrete examples, according to the “learning by doing” principle. Also because of this reason, this guide cannot cover all the complexity of the ROOT package. Nevertheless, once you feel confident with the concepts presented in the following chapters, you will be able to appreciate the ROOT Users Guide (The ROOT Users Guide 2015) and navigate through the Class Reference (The ROOT Reference Guide 2013) to find all the details you might be interested in. You can even look at the code itself, since ROOT is a free, open-source product. Use these documents in parallel to this tutorial!

The ROOT Data Analysis Framework itself is written in and heavily relies on the C++ programming language: some knowledge about C++ is required. Jus take advantage from the immense available literature about C++ if you do not have any idea of what this language is about.

ROOT is available for many platforms (Linux, Mac OS X, Windows...), but in this guide we will implicitly assume that you are using Linux. The first thing you need to do with ROOT is install it, don't you ? Obtaining the latest ROOT version is straightforward. Just seek the “Pro” version on this webpage <http://root.cern.ch/downloading-root>. You will find precompiled versions for the different architectures, or the ROOT source code to compile yourself. Just pick up the flavour you need and follow the installation instructions.

Let's dive into ROOT!