Zbias.cpp instructions

How to use it

Once the program has been compiled with RooFit, simply using: `g++ Zbias.cpp -o executable 'root-config --glibs --cflags' -lRooFit -lRooFitCore` the user will be asked to specify the number of toys he wants to generate and what the bkg models (for the generation and the fit respectively) should be.

The program can be executed by typing

`./executable commands`

where the commands are as follow:

- `--nsim=zzz` where `zzz` is a number (integer). It specifies the number of TOYMC the program will generate (200 will be enough).

- `--sim=XY` where `X` is the bkg model used to generate the TOYMC and `Y` is bkg model to use in the fitting pdf. The allowed vaues are `E` for the Exponentials and `B` for a 4th-grade Bernstein polynomial.

The program is not optimized, which means it is slow in the generation of toys and there are not controls about the input values (X and Y must be only E or B). Default values are `zzz`=0, `XY`=BB.

What it needs

The program will need few files, namely:

- `Ztemplate.root`. It's a ROOT file which contains the Z template (it actually is an histogram) that will be taken in order to generate the signal pdf.

- `generateSimulation.h`. This files contains the algorithms the program will use to generate the simulations.

What it does

The program takes the signal pdf directly from the root file (**Ztemplate.root**). It starts producing several TOYMC and fit them by using t first RooChi2Var and RooMinimizer (calling migrad, hesse and minos in sequence) and then using RooNLLVar and RooMinimizer (calling again migrad, hesse and minos in sequence).

In both cases the number of signal events calculated $N_{measured}$, and their errors, will be saved. Once the simulations are over, the quantity

$$\frac{N_{measured} - N_{simulated}}{\sigma_{measured}}$$

will be plot. These results will then be fitted with two gaussian-pdfs in order to obtain the corresponding means and estimate the biases. The results obtained from the Chi2-method interpolation will be in red, while the results from the NLL-method interpolation will be in blu.

What the user should see, in the ideal case, are two gaussian with their means equal to zero and their sigma equal to one.

What it produces

The program will produce a folder (**Ztry**) in which it will store, in an orderly way, the results. There will be two text files, one for the Chi2-method and one for the NLL-method. Using the same notations about the bkg-models, the names of the files will be `datas_XY_K.txt` where `K` is the interpolation method used (`Chi2` or `NLL`). The data contained in these files will be stored as follow:

$$N_{measured} \qquad \sigma_{measured} \qquad (N_{measured} - N_{simulated})/\sigma_{measured}$$

The program produces as well a pdf file (`plot_XY.pdf`) with the output of the program.