ROOT: Functions

Seema Bahinipati
IIT Bhubaneswar

Outline

- Plotting inbuilt functions

- Plotting user-defined functions

Fitting with pre-defined functions

```
hist.Fit("gaus");
```

- " `gaus` " Gaussian function with 3 parameters: `f(x) = p0*exp(-0.5*((x-p1)/p2)^2)`
- " `expo` "An Exponential with 2 parameters: `f(x) = exp(p0+p1*x)`
- " `pol` *`N`* " A polynomial of degree *N*, where N is a number between 0 and 9: `f(x) = p0 + p1*x + p2*x2 +...`
- " `chebyshev` *`N`* " A Chebyshev polynomial of degree *N*, where N is a number between 0 and 9: `f(x) = p0 + p1*x + p2*(2*x2-1) +...`
- " `landau` " Landau function with mean and sigma. This function has been adapted from the `CERNLIB` routine `G110 denlan` (see `TMath::Landau`).
- " `gausn` Normalized form of the gaussian function with 3 parameters `f(x) = p0*exp(-0.5*((x-p1)/p2)^2)/(p2 *sqrt(2PI))`

Creating user-defined functions

Creating a TF1 with a Formula

```
root[] TF1  *f1 = new TF1("f1","sin(x)/x",0,10)
```

```
root[] TF1  *f2 = new TF1("f2","f1*2",0,10)
```

Creating a TF1 with Parameters

```
TF1 *f1 = new TF1("f1","[0]*x*sin([1]*x)",-3,3);
```

```
f1->SetParameter(0,10);
```

```
f1->SetParameters(10,5);
```

```
f1->Draw()
```

Creating user-defined functions

Detailed Description

View Notebook Open in SWAN Read data from an ascii file and create a root file with an histogram and an ntuple.

See a variant of this macro in **basic2.C**.

```
#include "Riostream.h"
void basic() {
// read file $ROOTSYS/tutorials/tree/basic.dat
// this file has 3 columns of float data
   TString dir = gROOT->GetTutorialDir();
   dir.Append("/tree/");
   dir.ReplaceAll("/./","/");
   ifstream in;
   in.open(Form("%sbasic.dat",dir.Data()));

   Float_t x,y,z;
   Int_t nlines = 0;
   auto f = TFile::Open("basic.root","RECREATE");
   TH1F h1("h1","x distribution",100,-4,4);
   TNtuple ntuple("ntuple","data from ascii file","x:y:z");

   while (1) {
      in >> x >> y >> z;
      if (!in.good()) break;
      if (nlines < 5) printf("x=%8f, y=%8f, z=%8f\n",x,y,z);
      h1.Fill(x);
      ntuple.Fill(x,y,z);
      nlines++;
   }
   printf(" found %d points\n",nlines);

   in.close();

   f->Write();
}
```

Author

Rene Brun

Definition in file **basic.C**.

Creating user-defined functions

Detailed Description

View Notebook Open in SWAN Read data from an ascii file and create a root file with an histogram and an ntuple.

See a variant of this macro in basic2.C.

```
#include "Riostream.h"
void basic() {
// read file $ROOTSYS/tutorials/tree/basic.dat
// this file has 3 columns of float data
   TString dir = gROOT->GetTutorialDir();
   dir.Append("/tree/");
   dir.ReplaceAll("/./","/");
   ifstream in;
   in.open(Form("%sbasic.dat",dir.Data()));

   Float_t x,y,z;
   Int_t nlines = 0;
   auto f = TFile::Open("basic.root","RECREATE");
   TH1F h1("h1","x distribution",100,-4,4);
   TNtuple ntuple("ntuple","data from ascii file","x:y:z");

   while (1) {
      in >> x >> y >> z;
      if (!in.good()) break;
      if (nlines < 5) printf("x=%8f, y=%8f, z=%8f\n",x,y,z);
      h1.Fill(x);
      ntuple.Fill(x,y,z);
      nlines++;
   }
   printf(" found %d points\n",nlines);

   in.close();

   f->Write();
}
```

Author

Rene Brun

Reference

- https://root.cern.ch/doc/v610/group__tutorial__tree.html