OUTLINE of Exercise-4

PRACTICAL CLASS 4

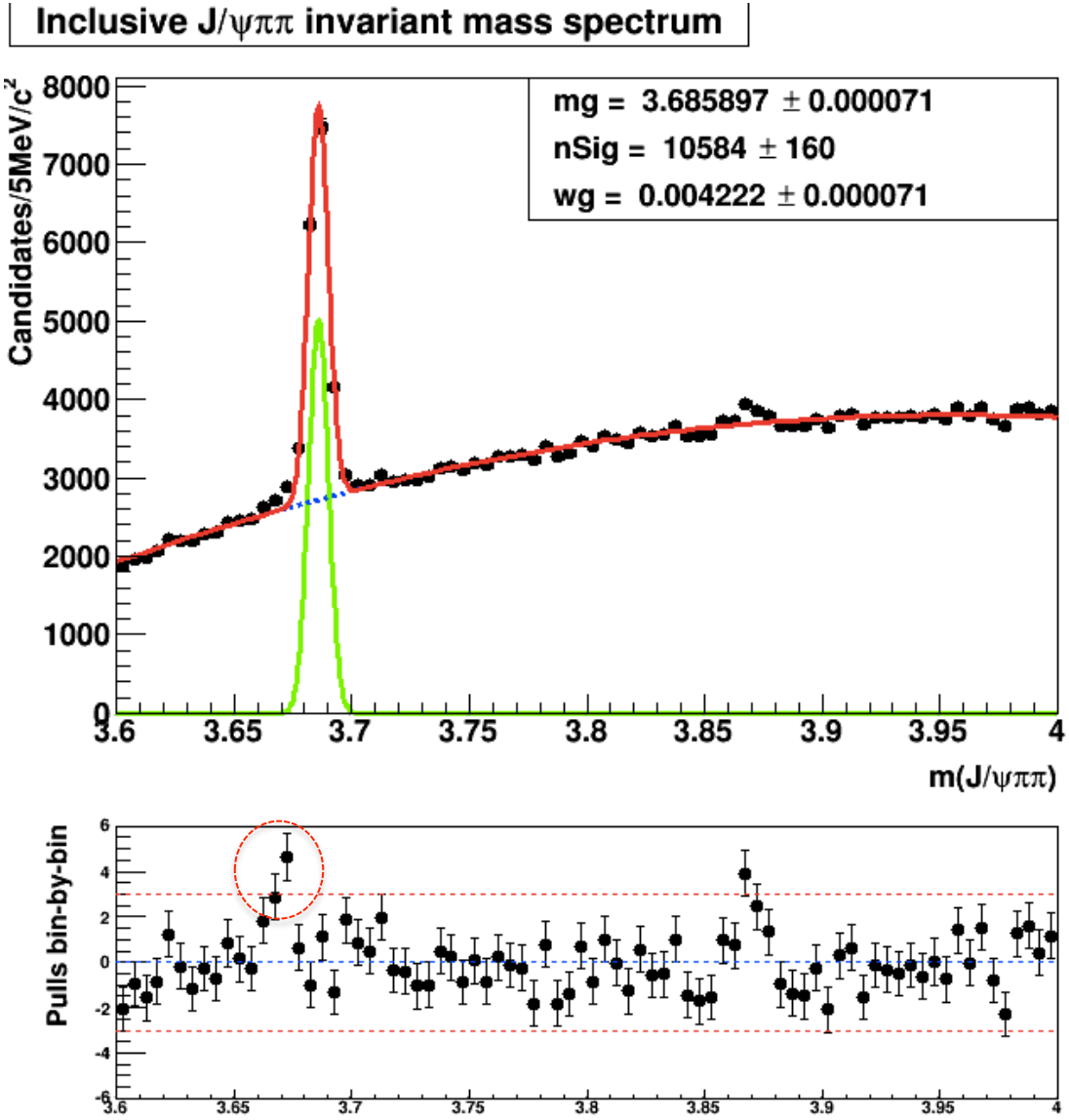
for the Course

Laboratorio Analisi Dati

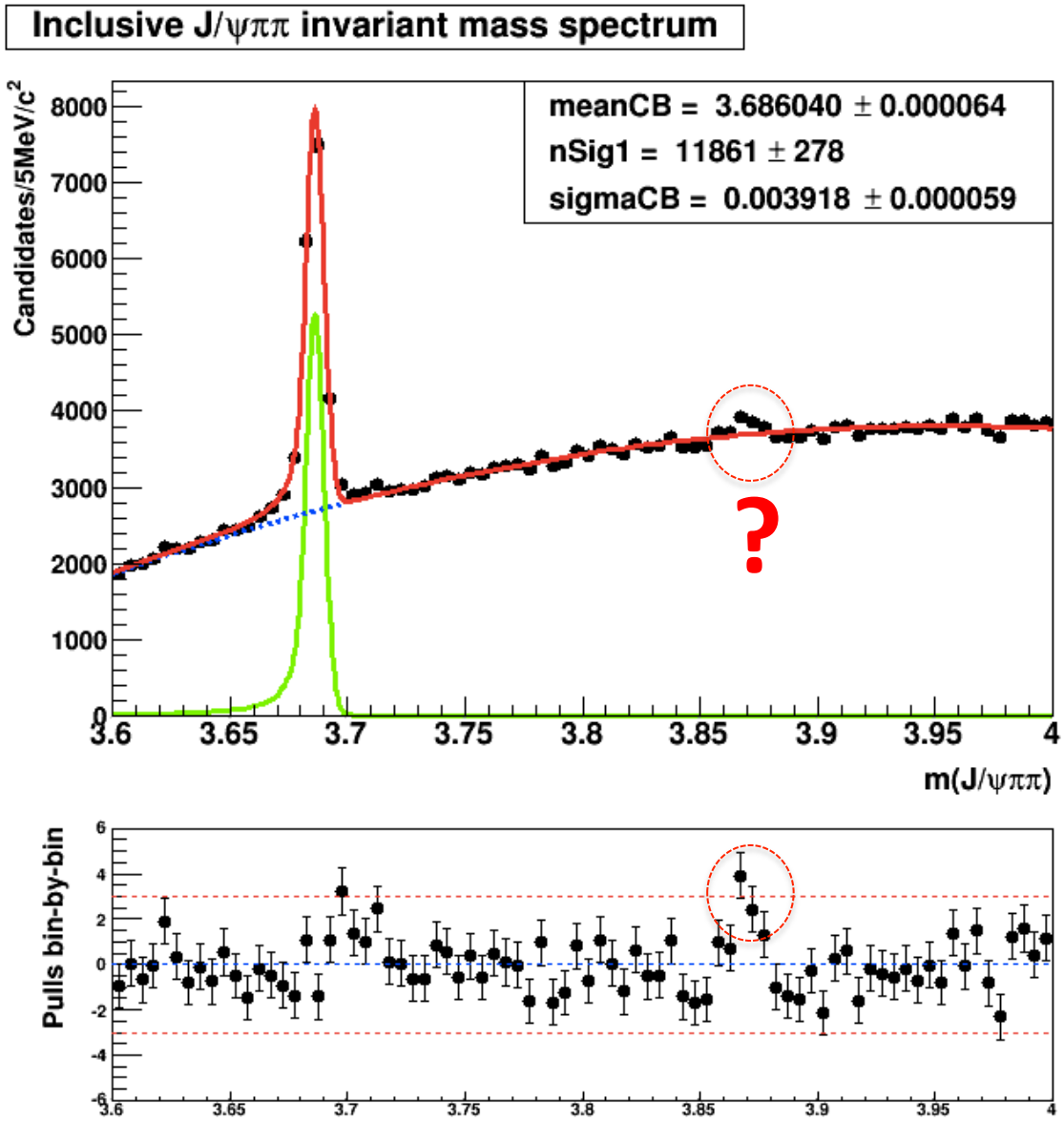
2017/2018

Prof. A.Pompili

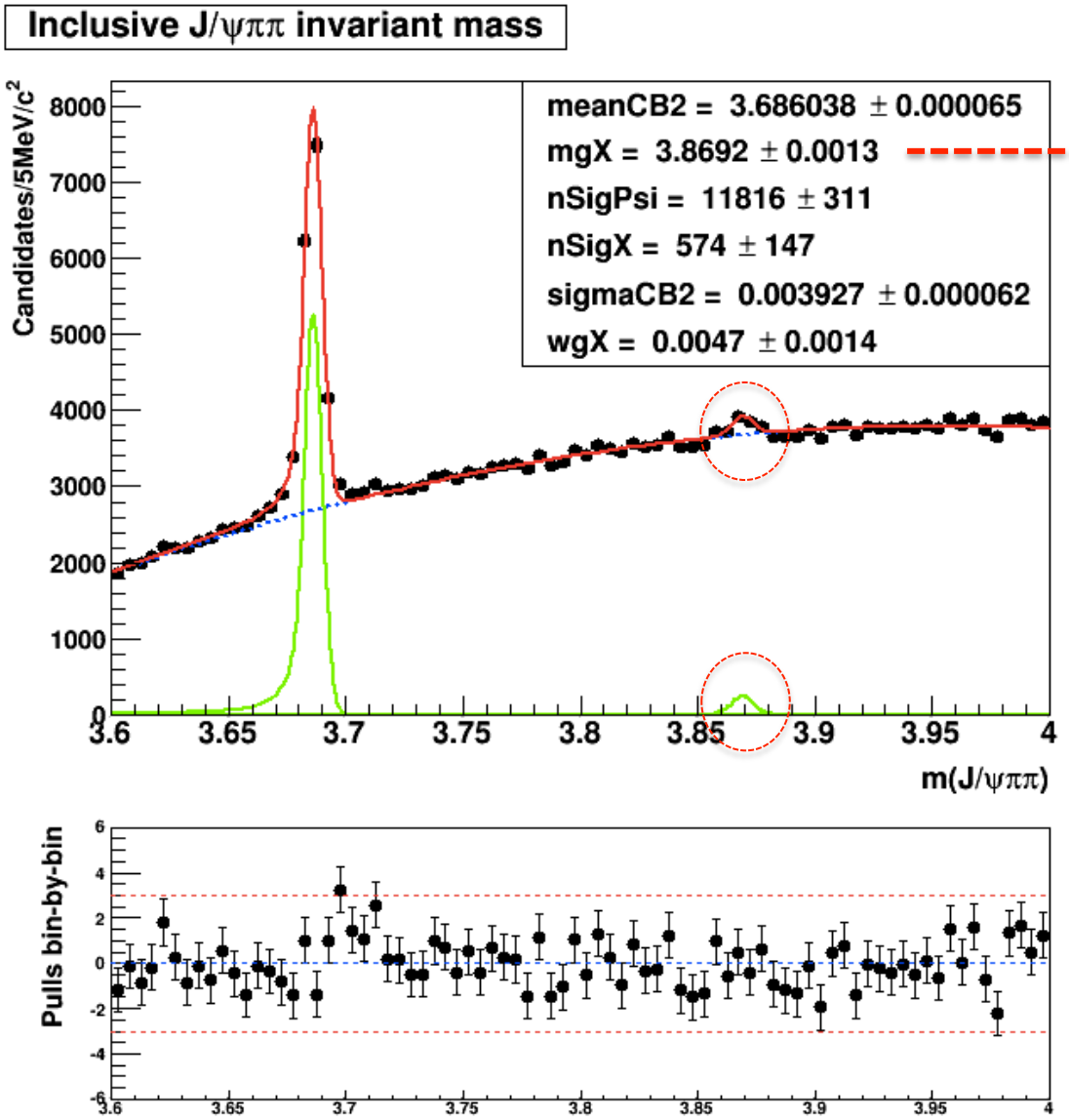
First attempt to fit:



To describe the radiative tail we can use as usual a single-sided Crystal Ball:



Finally we can introduce an additional signal (simple gaussian) :



This mass estimate hints, considering the final state, that it must a X(3872) :

Citation: C. Patrignani *et al.* (Particle Data Group), Chin. Phys. C, **40**, 100001 (2016)

X(3872) $I^G(J^{PC}) = 0^+(1^{++})$

First observed by CHOI 03 in $B \to K\pi^+\pi^- J/\psi(1S)$ decays as a narrow peak in the invariant mass distribution of the $\pi^+\pi^- J/\psi(1S)$ final state. Isovector hypothesis excluded by AUBERT 05B and CHOI 11.

AAIJ 13Q perform a full five-dimensional amplitude analysis of the angular correlations between the decay products in $B^+ \to X(3872)K^+$ decays, where $X(3872) \to J/\psi\pi^+\pi^-$ and $J/\psi \to \mu^+\mu^-$, which unambiguously gives the $J^{PC} = 1^{++}$ assignment under the assumption that the $\pi^+\pi^-$ and J/ψ are in an S-wave. AAIJ 15AO extend this analysis with more data to limit D-wave contributions to $< 4\%$ at 95% CL.

See our note on "Developments in Heavy Quarkonium Spectroscopy".

X(3872) MASS FROM $J/\psi X$ MODE

VALUE (MeV)	EVTS	DOCUMENT ID	TECN	COMMENT
3871.69± 0.17 OUR AVERAGE				

This is the RooFit code (*test.C*) to execute the 3 fits in a sequence:

```
#include <TROOT.h>
#include <TFile.h>
#include <TH1.h>
#include <TF1.h>
#include <TF2.h>
#include <TFormula.h>
#include <TStyle.h>
#include <TCanvas.h>
#include <TProfile.h>
#include <TString.h>
#include <TLine.h>
#include <TPad.h>
#include <TMath.h>
#include <TLatex.h>
#include <TLegend.h>
#include <iostream>
#include <TColor.h>
#include "TAxis.h"

using namespace RooFit;

TStyle *myStyle= new TStyle("myStyle","myStyle");

//////////////////////////---inizio main  ///// to execute: .L test.C + main()

//void main(TString date, TString extens) {

void main() {
  //
  gROOT->SetStyle("Plain");
  gStyle->SetCanvasColor(0);
  gStyle->SetOptStat(10);
  //
  //gROOT->SetStyle("myStyle");
  //myStyle->SetFrameBorderMode(0); myStyle->SetCanvasBorderMode(0);
  //myStyle->SetPadBorderMode(0); myStyle->SetPadColor(0);
  //myStyle->SetStatColor(0); myStyle->SetFillColor(0);
  //myStyle->SetStatBorderStyle(1);
  //
  TCanvas* myC = new TCanvas("myC","Plots",700,700);
  myC->SetFrameFillColor(0);
  //myC->cd(1)->SetBottomMargin(0.41); myC->cd(1)->SetTopMargin(0.05) ;
  //
  ////////////////////////////////////////////////////////////////////////
  //
  TFile f1("./esame-dec2014.root","READ");
  TH1D *hist = (TH1D*)f1.Get("PsiPrime_Mass_cut6");
  //
  RooRealVar x("x","x",3.6,4.0);
  RooDataHist jpsipipi_mass(hist->GetName(),hist->GetTitle(),RooArgSet(x),RooFit::Import(*hist, kFALSE));
  //
  ////////////////////////////////////////////////////////////////////////
```

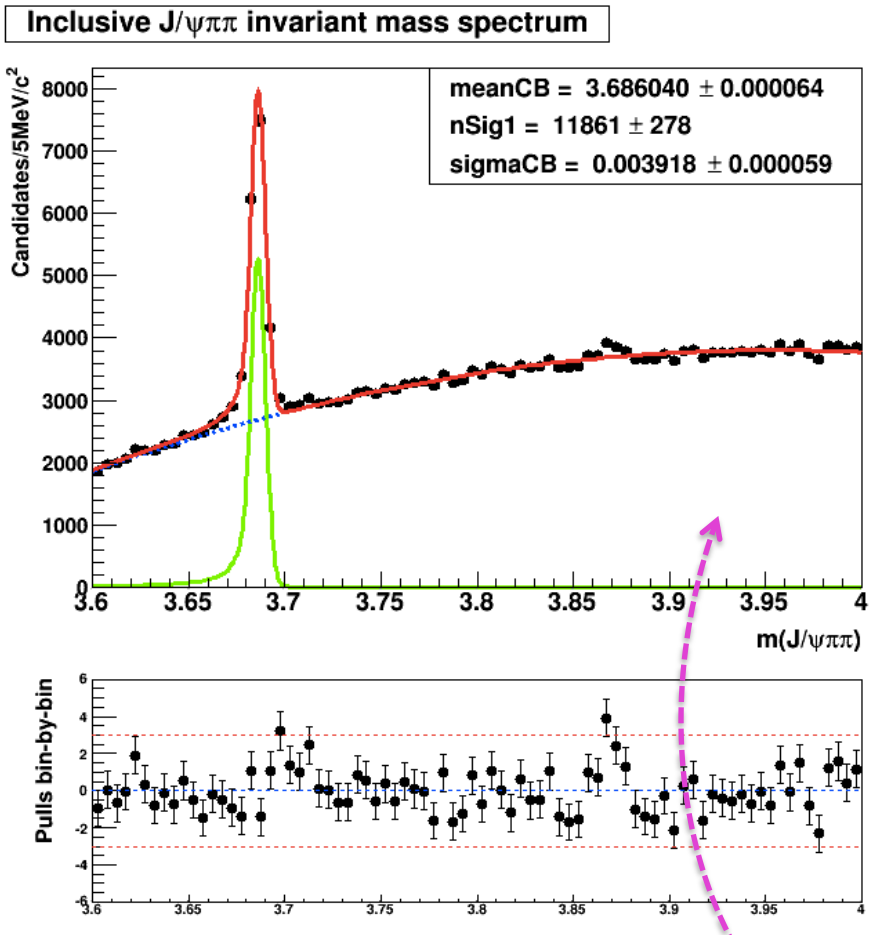
First fit:

```
//
RooPlot* xframe = x.frame(Title(""));
xframe->SetTitle("Inclusive J/#psi#pi#pi invariant mass spectrum");
xframe->SetTitleOffset(1.32,"y");
xframe->SetYTitle("Candidates/5MeV/c^{2}");
xframe->SetTitleOffset(1.26,"x");
xframe->SetXTitle("m(J/#psi#pi#pi)");
//
jpsipipi_mass.plotOn(xframe);
//xframe->Draw(); // to have immediately a first look to the histogram content
//
char *title[128]; jpsipipi_mass->SetTitle(*title); title ="";
//
////////////////////////////////////////////// FIT
//
// signal
RooRealVar mg("mg","Gaussian's mean",3.685, 3.675, 3.695);
RooRealVar wg("wg","Gaussian's width",0.01, 0.001, 0.05);
RooGaussian myGauss("myGauss","Gauss(x,mg,wg,)",x,mg,wg);
//
// background
RooRealVar c0("c0","1st coeff",0.3,-100000,100000);
RooRealVar c1("c1","2nd coeff",-0.1,-100000,100000);
RooChebychev cheby("cheby","Chebyshev",x,RooArgList(c0,c1));
//
////RooRealVar c2("c2","3rd coeff",1.,-100000,100000);
////RooRealVar c3("c3","4th coeff",0.5,-1000,1000);
////RooChebychev cheby("cheby","Chebyshev",x,RooArgList(c0,c1,c2,c3));
//
// total pdf : f*gauss + (1-f)*cheby
//RooRealVar fsig("fsig","signal fraction",0.02,0.0,0.7);
//
RooRealVar nSig("nSig","Number of signal cands", 4e+5, 1.,1e+7);
RooRealVar nBkg("nBkg","Number of bkg componet", 120e+3, 1., 1e+8);
RooAddPdf* totalPdf = new RooAddPdf("totalPdf","totalPdf",RooArgList(myGauss,cheby),RooArgList(nSig,nBkg));
//
//
totalPdf->fitTo(jpsipipi_mass,Extended(kTRUE));
totalPdf->plotOn(xframe,RooFit::LineColor(kRed));
totalPdf->plotOn(xframe,RooFit::Components(RooArgSet(myGauss)), LineColor(kGreen));
totalPdf->plotOn(xframe,RooFit::Components(cheby),RooFit::LineStyle(kDashed));
// plot full fit again to make correct pulls
totalPdf->plotOn(xframe,RooFit::LineColor(kRed));
//totalPdf->paramOn(xframe);
totalPdf->paramOn(xframe, Parameters(RooArgSet(mg,wg,nSig)), Layout(0.45,0.9,0.9));
//
```

Rapresentation of 1st fit:

```
//
RooPlot *framePull = x.frame("");
framePull->addObject((TObject*)xframe->pullHist(),"p");
framePull->SetTitle("");
 framePull->SetLabelSize(0.055,"y");
framePull->SetTitleSize(0.085,"y");
framePull->SetTitleOffset(0.35,"y");
framePull->SetYTitle("Pulls bin-by-bin");
framePull->SetLabelSize(0.055,"x");
framePull->SetXTitle(" ");
framePull->SetMinimum(-6.);
framePull->SetMaximum(6.);
//
myC->Divide(0,2);
myC->cd(2);
gPad->SetPad(0.,0.,1.,0.3);
//framePull->SetTitleOffset(1.25,"y");
//framePull->SetTitleSize(0.1,"y");
gStyle->SetLabelSize(0.06,"Y");
gStyle->SetTitleYSize(0.03);
framePull->Draw();
TLine* lineplus = new TLine(3.6,3.,4.,3.);
TLine* lineminus = new TLine(3.6,-3.,4.,-3.);
TLine* linezero = new TLine(3.6,0.,4.,0.);
lineplus->SetLineStyle(2);
lineplus->SetLineColor(2);
lineplus->Draw("same");
lineminus->SetLineStyle(2);
lineminus->SetLineColor(2);
lineminus->Draw("same");
linezero->SetLineStyle(2);
linezero->SetLineColor(4);
linezero->Draw("same");
myC->cd(1);
gPad->SetPad(0.,0.3,1.,1.);
xframe->Draw();
//
myC->SaveAs("./psiprime_gauss_cheby2.png");
//myC->Update();
delete myC;
```

Second fit:




```
////////////////////////////////////////////////////////// NEW FIT
//
TCanvas* myC1 = new TCanvas("myC1","Plots",700,700);
myC1->SetFrameFillColor(0);
//
RooPlot* xframe1 = x.frame("");
xframe1->SetTitle("Inclusive J/#psi#pi#pi invariant mass spectrum");
xframe1->SetTitleOffset(1.32,"y");
xframe1->SetLabelSize(0.035,"y");
xframe1->SetTitleSize(0.037,"y");
xframe1->SetYTitle("Candidates/5MeV/c^{2}");
xframe1->SetTitleOffset(1.26,"x");
xframe1->SetXTitle("m(J/#psi#pi#pi)");
jpsipipi_mass.plotOn(xframe1);
//
// alternative (CB)
RooRealVar meanCB("meanCB", "meanCB", 3.685, 3.675, 3.695);
RooRealVar sigmaCB("sigmaCB", "sigmaCB", 0.0042222, 0.0004, 0.005);
RooRealVar alpha("alpha","alpha", 1.0, 0.00001, 10000.);
RooRealVar nCB("nCB","nCB", 1.0, 0.0001, 10000.);
//
RooCBShape myCB("myCB", "myCB", x, meanCB, sigmaCB, alpha, nCB);
//
RooRealVar nSig1("nSig1","Number of signal cands", 1e+4, 100.,1e+7);
RooRealVar nBkg1("nBkg1","Number of bkg componet", 2e+5, 1000., 1e+8);
RooAddPdf* totalPdf1 = new RooAddPdf("totalPdf1","totalPdf1",RooArgList(myCB,cheby),RooArgList(nSig1,nBkg1));
//
totalPdf1->fitTo(jpsipipi_mass,Extended(kTRUE));
totalPdf1->plotOn(xframe1,RooFit::LineColor(kRed));
totalPdf1->plotOn(xframe1,RooFit::Components(RooArgSet(myCB)), LineColor(kGreen));
totalPdf1->plotOn(xframe1,RooFit::Components(cheby),RooFit::LineStyle(kDashed));
// plot full fit again to make correct pulls
totalPdf1->plotOn(xframe1,RooFit::LineColor(kRed));
//totalPdf1->paramOn(xframe1); // non mettere proprio le stime dei parametri restituite dal fit
totalPdf1->paramOn(xframe1, Parameters(RooArgSet(meanCB,sigmaCB,nSig1)), Layout(0.45,0.9,0.9));
//
////////////////////////////////
//
RooPlot *framePull1 = x.frame("");
framePull1->addObject((TObject*)xframe1->pullHist(),"p");
framePull1->SetTitle("");
framePull1->SetLabelSize(0.055,"y");
framePull1->SetTitleSize(0.085,"y");
framePull1->SetTitleOffset(0.35,"y");
framePull1->SetYTitle("Pulls bin-by-bin");
framePull1->SetLabelSize(0.055,"x");
framePull1->SetXTitle(" ");
framePull1->SetMinimum(-6.);
framePull1->SetMaximum(6.);
//
myC1->Divide(0,2);
myC1->cd(2);
//
gPad->SetPad(0.,0.,1.,0.3);
framePull1->Draw();
TLine* lineplus1 = new TLine(3.6,3.,4.,3.);
TLine* lineminus1 = new TLine(3.6,-3.,4.,-3.);
TLine* linezero1 = new TLine(3.6,0.,4.,0.);
lineplus1->SetLineStyle(2);
lineplus1->SetLineColor(2);
lineplus1->Draw("same");
lineminus1->SetLineStyle(2);
lineminus1->SetLineColor(2);
lineminus1->Draw("same");
linezero1->SetLineStyle(2);
linezero1->SetLineColor(4);
linezero1->Draw("same");
//
myC1->cd(1);
gPad->SetPad(0.,0.3,1.,1.);
xframe1->Draw();
//
myC1->SaveAs("./psiprime_cb_cheby2.png");
```

Third fit:

```
////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////  NEW FIT
////////////////////////////////////////////////////////////////////
//
gROOT->SetStyle("Plain");
gStyle->SetCanvasColor(0);
gStyle->SetOptStat(10);
//
TCanvas* myC2 = new TCanvas("myC2","Plots",700,700);
myC2->SetFrameFillColor(0);
//
RooPlot* xframe2 = x.frame(Title(""));
xframe2->SetTitle("Inclusive J/#psi#pi#pi invariant mass");
xframe2->SetTitleOffset(1.32,"y");
xframe2->SetLabelSize(0.035,"y");
xframe2->SetTitleSize(0.037,"y");
xframe2->SetYTitle("Candidates/5MeV/c^{2}");
xframe2->SetTitleOffset(1.26,"x");
xframe2->SetXTitle("m(J/#psi#pi#pi)");
jpsipipi_mass.plotOn(xframe2);
//
// signal
RooRealVar mgX("mgX","Gaussian's mean",3.868, 3.85, 3.88);
RooRealVar wgX("wgX","Gaussian's width",0.005, 0.002, 0.015);
//wgX.setConstant(kTRUE);
RooGaussian myGaussX("myGaussX","GaussX(x,mgX,wgX)",x,mgX,wgX);
//mgX.setConstant(kTRUE);
//
//RooRealVar meanCB2("meanCB2", "meanCB2", 3.685, 3.675, 3.695);
//RooRealVar sigmaCB2("sigmaCB2", "sigmaCB2", 0.004, 0.0001, 0.05);
//RooRealVar alpha2("alpha2","alpha2", 1.0, 0.00001, 10000.);
//RooRealVar nCB2("nCB2","nCB2", 1.0, 0.0001, 10000.);
//
RooRealVar meanCB2("meanCB2", "meanCB2", 3.686038, 3.68, 3.692);
RooRealVar sigmaCB2("sigmaCB2", "sigmaCB2", 0.003919, 0.001, 0.05);
RooRealVar alpha2("alpha2","alpha2", 1.41, 0.01, 10.);
RooRealVar nCB2("nCB2","nCB2", 1.64, 0.1, 10.);
// start fixing the CB parameters from previous fit and release them later (leave just # candidates free)
//meanCB2.setConstant(kTRUE);
//sigmaCB2.setConstant(kTRUE);
//alpha2.setConstant(kTRUE);
//nCB2.setConstant(kTRUE);
//
RooCBShape myCB2("myCB2", "myCB2", x, meanCB2, sigmaCB2, alpha2, nCB2);
//
RooRealVar nSigPsi("nSigPsi","Number of signal psi cands", 11858, 10000.,15000.); //start from previous plot to help
////////////////////////////////////////////nSigPsi.setConstant(kTRUE);
//
RooRealVar nSigX("nSigX","Number of signal X cands", 550, 350., 1500.);
//nSigX.setConstant(kTRUE);
//
////////RooAddPdf* totalSig2 = new RooAddPdf("totalSig2","totalSig2",RooArgList(myCB2,myGaussX),RooArgList(nSigPsi,nSigX));
//RooAddPdf totalSig2("totalSig2","totalSig2",RooArgList(myCB2,myGaussX),RooArgList(nSigPsi,nSigX));
//
//RooRealVar nSig2("nSig2","Number of total sig component", 2e+5, 1000., 1e+7);
RooRealVar nBkg2("nBkg2","Number of bkg component", 2e+5, 1000., 1e+8);

//RooAddPdf* totalPdf2 = new RooAddPdf("totalPdf2","totalPdf2",RooArgList(totalSig2,cheby),RooArgList(nSig2,nBkg2));
RooAddPdf* totalPdf2 = new RooAddPdf("totalPdf2","totalPdf2",RooArgList(myCB2,myGaussX,cheby),RooArgList(nSigPsi,nSigX,nBkg2));
// one shot fit !
//
gStyle->SetLineWidth(1); // cambia nulla
gStyle->SetFuncWidth(1);
//
totalPdf2->fitTo(jpsipipi_mass,Extended(kTRUE));
totalPdf2->plotOn(xframe2,RooFit::LineColor(kRed),RooFit::LineWidth(1));
// by default LineWidth is 3 pixels (somehow thick, while 1 is too subtle)
totalPdf2->plotOn(xframe2,RooFit::Components(RooArgSet(myGaussX)), LineColor(kGreen),RooFit::LineWidth(2));
totalPdf2->plotOn(xframe2,RooFit::Components(RooArgSet(myCB2)), LineColor(kGreen),RooFit::LineWidth(2));
totalPdf2->plotOn(xframe2,RooFit::Components(cheby),RooFit::LineStyle(kDashed),RooFit::LineWidth(2));
// plot full fit again to make correct pulls
totalPdf2->plotOn(xframe2,RooFit::LineColor(kRed),RooFit::LineWidth(2));
totalPdf2->paramOn(xframe2, Parameters(RooArgSet(meanCB2,sigmaCB2,nSigPsi,mgX,wgX,nSigX)), Layout(0.45,0.9,0.9));
```

Representation of 3rd fit:

```
//
RooPlot *framePull2 = x.frame("");
framePull2->addObject((TObject*)xframe2->pullHist(),"p");
framePull2->SetTitle(""); // elimina titolo
framePull2->SetLabelSize(0.055,"y");
framePull2->SetTitleSize(0.085,"y");  // ingrandisce ma sposta anche verso sinistra
framePull2->SetTitleOffset(0.35,"y"); // risposta a destra
framePull2->SetYTitle("Pulls bin-by-bin");
framePull2->SetLabelSize(0.055,"x");
framePull2->SetXTitle(" ");  //framePull2->SetXTitle("m(J/#psi#pi#pi)"); // pleonastico
framePull2->SetMinimum(-6.);
framePull2->SetMaximum(6.);
//
myC2->Divide(0,2);
myC2->cd(2);
//
gPad->SetPad(0.,0.,1.,0.3);
framePull2->Draw();
//
TLine* lineplus2 = new TLine(3.6,3.,4.,3.);
TLine* lineminus2 = new TLine(3.6,-3.,4.,-3.);
TLine* linezero2 = new TLine(3.6,0.,4.,0.);
lineplus2->SetLineStyle(2);
lineplus2->SetLineColor(2);
lineplus2->Draw("same");
lineminus2->SetLineStyle(2);
lineminus2->SetLineColor(2);
lineminus2->Draw("same");
linezero2->SetLineStyle(2);
linezero2->SetLineColor(4);
linezero2->Draw("same");
//
myC2->cd(1);
gPad->SetPad(0.,0.3,1.,1.);
xframe2->Draw();
//
myC2->SaveAs("./psiprime_cb_cheby2_x3872.png");
//myC2->Clear();
delete myC2;
//
////////////////////////////////////////////
//
f1.Close();
f1.Delete();
//
gROOT->Reset();
gROOT->Clear();
//
}
```