Exercicios da aula 4 de estatística - RooFit

Sérgio da Silva dos Santos Júnior
Professores: Dilson Damião, Eliza Melo e Mauricio Thiel

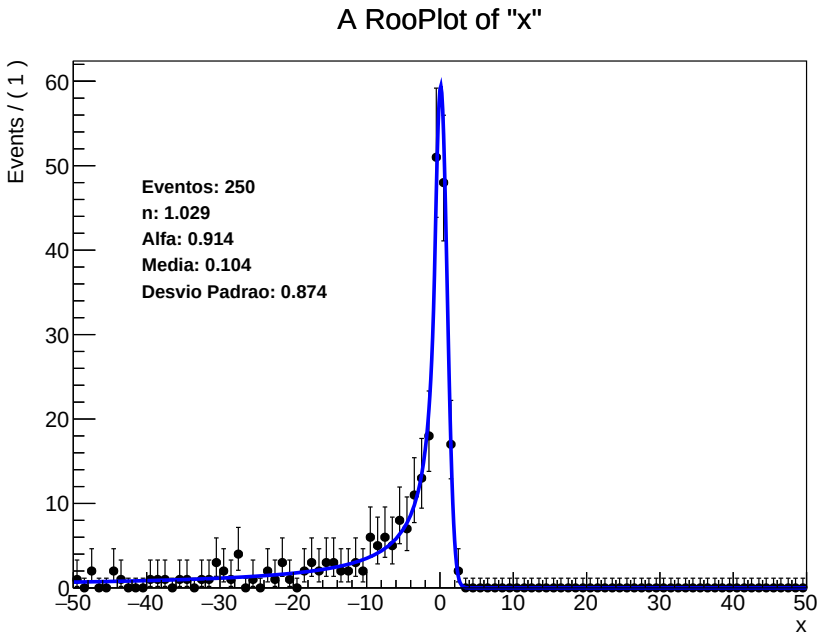
2024/2

1 Exercício 1

O código para o exercício foi:

```
#include <iostream>
#include <RooRealVar.h>
#include <RooPlot.h>
#include <RooFitResult.h>
#include <RooFit.h>
#include <RooDataSet.h>
#include <TCanvas.h>
#include <RooCrystalBall.h>
#include <TLatex.h>

void cristall() {

    RooRealVar x("x", "x", -50, 50);
    RooRealVar media("media", "media", 0, -50, 50);
    RooRealVar sigma("sigma", "sigma", 1, 0.3, 10);
    RooRealVar alfa("alfa", "alfa", 1, 0.2, 10);
    RooRealVar n("n", "n", 1, 0, 15);

    RooCrystalBall crystalball("crystalball", "CrystalBall", x,
    media, sigma, alfa, n);


    RooDataSet* dados = crystalball.generate(RooArgSet(x), 250);
    crystalball.fitTo(*dados, RooFit::Save());


    TCanvas canvas("cristal", "Crystal Ball", 800, 600);
    RooPlot* frame = x.frame();
    dados->plotOn(frame);
    crystalball.plotOn(frame);
    frame->Draw();


    TLatex latex;
    latex.SetNDC();
    latex.SetTextSize(0.03);

    latex.DrawLatex(0.175, 0.70, Form("Eventos: %d", (int)dados->
    numEntries()));
    latex.DrawLatex(0.175, 0.66, Form("n: %.3f", n.getVal()));
    latex.DrawLatex(0.175, 0.62, Form("Alfa: %.3f", alfa.getVal()))
    ;
    latex.DrawLatex(0.175, 0.58, Form("Media: %.3f", media.getVal()
    ));
    latex.DrawLatex(0.175, 0.54, Form("Desvio Padrao: %.3f", sigma.
    getVal()));


    canvas.SaveAs("Cristall.pdf");
}
```

Obtendo como resultado:

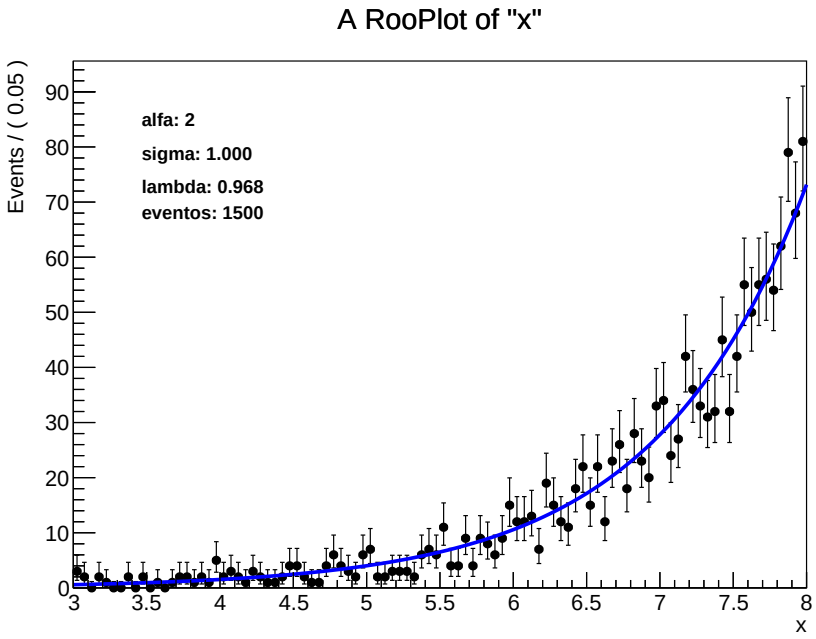

A RooPlot of "x"
Events / (1)
Eventos: 250
n: 1.029
Alfa: 0.914
Media: 0.104
Desvio Padrao: 0.874
x

2 Exercício 2

O código para o exercício foi:

```
#include <RooRealVar.h>
#include <RooDataSet.h>
#include <RooExponential.h>
#include <RooPlot.h>
#include <TCanvas.h>
#include <TLatex.h>

void exp() {

    RooRealVar x("x", "x", 3, 8);
    RooRealVar lambda("lambda", "lambda", 1, 0.1, 2);
    RooRealVar eventos("eventos", "Eventos", 0, 1500, 8);
    RooRealVar sigma("sigma", "sigma", 1, 0.1, 3);
    RooRealVar alfa("alfa", "alfa", 1.5, 0.1, 8);

    RooExponential expDecay("expDecay", "Decaimento Exponencial", x
    , lambda);


    RooDataSet* dados = expDecay.generate(RooArgSet(x), 1500);
    expDecay.fitTo(*dados, RooFit::Save(), RooFit::Extended(kTRUE))
    ;

    TCanvas canvas("canvas", "Ajuste Exponencial", 800, 600);
    RooPlot* frame = x.frame();
    dados->plotOn(frame);
    expDecay.plotOn(frame);
    frame->Draw();

    TLatex latex;
    latex.SetNDC();
    latex.SetTextSize(0.03);
    latex.DrawLatex(0.175, 0.70, Form("lambda: %.3f", lambda.getVal
    ()));
    latex.DrawLatex(0.175, 0.66, Form("eventos: %.0f", eventos.
    getVal()));
    latex.DrawLatex(0.175, 0.75, Form("sigma: %.3f", sigma.getVal()
    ));
    latex.DrawLatex(0.175, 0.8, Form("alfa: %.0f", alfa.getVal()));

    canvas.SaveAs("Exponencial.pdf");
}
```

Obtendo como resultado:


A RooPlot of "x"
Events / (0.05)
alfa: 2
sigma: 1.000
lambda: 0.968
eventos: 1500
x

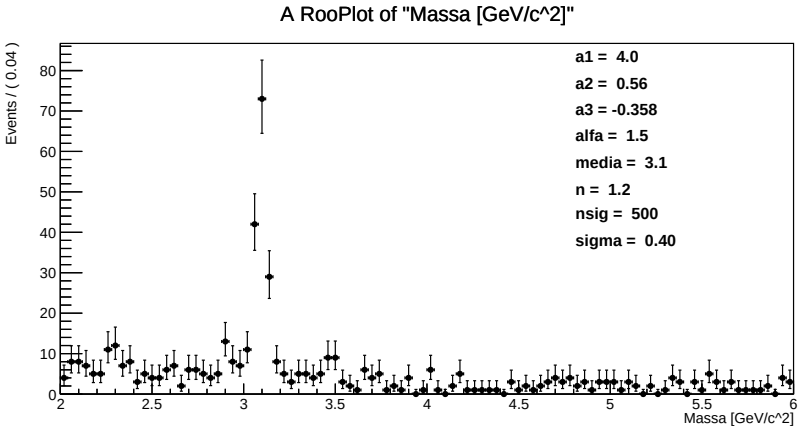
3 Exercício 3

O código para o exercício foi:

```
#include <iostream>
#include <RooRealVar.h>
#include <RooPlot.h>
#include <RooFit.h>
#include <RooDataSet.h>
#include <TCanvas.h>
#include <TFile.h>
#include <RooCrystalBall.h>
#include <RooPolynomial.h>
#include <RooAddPdf.h>
#include <TLatex.h>
#include <TH1F.h>

void model2() {

    TFile *file = TFile::Open("DataSet_lowstat.root");
    if (!file) {
        std::cerr << "Erro ao abrir o arquivo!" << std::endl;
        return;
    }

    RooDataSet* data = dynamic_cast<RooDataSet*>(file->Get("data"))
    ;
    if (!data) {
        std::cerr << "Erro ao carregar o conjunto de dados!" << std
    ::endl;
        return;
    }

    std::cout << "Numero de entradas: " << data->numEntries() <<
    std::endl;

    RooRealVar mass("mass", "Massa [GeV/c^2]", 2, 6);

    TH1F* hist = new TH1F("hist", "Distribuicao de Massa", 100, 2,
    6);
    data->fillHistogram(hist, RooArgList(mass));

    TCanvas canvasHist("canvasHist", "Histograma de Dados", 800,
    600);
    hist->Draw();
    canvasHist.SaveAs("data_histogram.png");

    RooRealVar nsig("nsig", "Numero de eventos de sinal", 500, 0,
    1000);
    RooRealVar media("media", "Media", 3.1, 2.9, 3.3);
    RooRealVar sigma("sigma", "Desvio Padrao", 0.4, 0.0001, 1.5);
    RooRealVar alfa("alfa", "Alfa", 1.5, -5., 6);
    RooRealVar n("n", "n", 1.2, 0.1, 4);
    RooCrystalBall signal("signal", "Sinal", mass, media, sigma,
    alfa, n);

    RooRealVar a1("a1", "a1", -0.5, -4., 4.);
    RooRealVar a2("a2", "a2", 0.5, -4., 4.);
```

```
    RooRealVar a3("a3", "a3", -0.5, -4., 4.);
    RooPolynomial background("background", "Fundo", mass,
    RooArgList(a1, a2, a3));

    RooAddPdf model("model", "Sinal + Fundo", RooArgList(signal,
    background), RooArgList(nsig));

    RooFitResult* result = model.fitTo(*data, RooFit::Save(),
    RooFit::Extended());

    result->floatParsFinal().Print();

    TCanvas canvas("canvas", "Ajuste J/(psi)", 1200, 600);
    RooPlot* frame = mass.frame();


    data->plotOn(frame);
    model.plotOn(frame, RooFit::LineColor(kRed), RooFit::DrawOption
    ("L"));
    model.paramOn(frame);


    double chi2 = frame->chiSquare();
    int ndf = data->numEntries() - result->floatParsFinal().getSize
    ();
    double chi2_ndf = chi2 / ndf;


    TLatex latex;
    latex.SetNDC();
    latex.SetTextSize(0.04);
    latex.DrawLatex(0.15, 0.75, Form("X^2/ndf: %.2f", chi2_ndf));


    frame->Draw();
    canvas.SaveAs("model2.pdf");


    file->Close();
}
```

Obtendo como resultado:



No gráfico, por algum motivo, não apareceu $\frac{\chi^2}{ndf}$ e nem a linha do gráfico.