

Análise de Dados em FAE

(01/11/2024)

Estudos de Estatística para Análise de Dados em HEP

Professores: Eliza Melo, Dilson Damião e Mauricio Thiel

Nome: Matheus Figueiredo de Paiva Nascimento

Exercício aula 5

O exercício da Aula 5 consiste em analisar a distribuição de momento transversal (p_T), pseudorapidez (η) e ângulo azimutal (ϕ) de múons registrados em uma amostra de dados de eventos. Além disso, é solicitado calcular a massa invariante dos dois múons com maior p_T e representar graficamente essas distribuições.

1 Método

O código em C++ a seguir foi utilizado para realizar a análise, utilizando a biblioteca ROOT para criação dos histogramas e cálculo da massa invariante dos dois múons de maior p_T .

```
#define MassaZZ_cxx
#include "MassaZZ.h"
#include <TH1F.h>
#include <TCanvas.h>
#include <iostream>
#include <cmath>

void MassaZZ::Loop() {
    if (fChain == 0) return;

    TH1F *h_Muon_pt = new TH1F("h_Muon_pt", "Distribuicao de pT dos Muons; pT (GeV); Eventos", 100, 0, 100);
    TH1F *h_Muon_eta = new TH1F("h_Muon_eta", "Distribuicao de eta dos Muons; eta; Eventos", 100, -4, 4);
    TH1F *h_Muon_phi = new TH1F("h_Muon_phi", "Distribuicao de phi dos Muons; phi; Eventos", 100, 0, 360);

    TH1F *h_MassInvariant = new TH1F("h_MassInvariant", "Massa Invariante dos Dois Muons com maior pT", 100, 0, 100);

    Long64_t nentries = fChain->GetEntriesFast();
```

```

for (Long64_t jentry = 0; jentry < nentries; jentry++) {
    Long64_t ientry = LoadTree(jentry);
    if (ientry < 0) break;
    fChain->GetEntry(jentry);

    if (nMuon > 1) {
        int idx1 = 0;
        int idx2 = 1;
        if (Muon_pt[idx2] > Muon_pt[idx1]) std::swap(idx1, idx2);
        for (int i = 2; i < nMuon; i++) {
            if (Muon_pt[i] > Muon_pt[idx1]) {
                idx2 = idx1;
                idx1 = i;
            } else if (Muon_pt[i] > Muon_pt[idx2]) {
                idx2 = i;
            }
        }

        h_Muon_pt->Fill(Muon_pt[idx1]);
        h_Muon_eta->Fill(Muon_eta[idx1]);
        h_Muon_phi->Fill(Muon_phi[idx1]);

        double deltaEta = Muon_eta[idx1] - Muon_eta[idx2];
        double deltaPhi = Muon_phi[idx1] - Muon_phi[idx2];
        double massInvariant = sqrt(2 * Muon_pt[idx1] * Muon_pt[idx2] * (cosh(deltaEta)
        h_MassInvariant->Fill(massInvariant);
    } else if (nMuon == 1) {
        h_Muon_pt->Fill(Muon_pt[0]);
        h_Muon_eta->Fill(Muon_eta[0]);
        h_Muon_phi->Fill(Muon_phi[0]);
    }
}

TCanvas *c1 = new TCanvas("c1", "Distribuicoes", 1200, 800);
c1->Divide(2, 2);

c1->cd(1); h_Muon_pt->Draw();
c1->cd(2); h_Muon_eta->Draw();
c1->cd(3); h_Muon_phi->Draw();
c1->cd(4); h_MassInvariant->Draw();

c1->SaveAs("distribuicoes_ZZ.png");
}

```

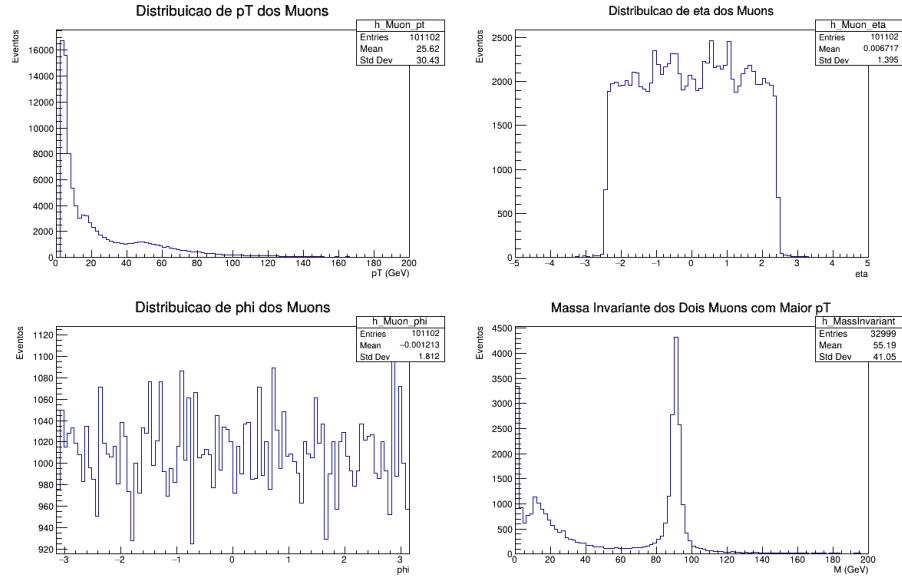


Figure 1: Distribuições de p_T , η , ϕ dos múons e massa invariante dos dois múons com maior p_T .

2 Resultado

O código acima produz quatro histogramas, que mostram as distribuições de p_T , η e ϕ dos múons, além da massa invariante dos dois múons com maior p_T . Abaixo, é apresentado o gráfico gerado pelo programa: