

```

1 // Analysis of data from collision events - Luca Morelli 2021
2
3 #include <iostream>
4 #include "TCanvas.h"
5 #include "TFile.h"
6 #include "TH1F.h"
7 #include "TStyle.h"
8
9 void analyze() {
10     // Setting graphs style
11     gStyle->SetOptStat("e");
12     gStyle->SetOptFit(1);
13     gStyle->SetFitFormat("7.6g");
14     gStyle->SetHistFillColor(kCyan);
15     gStyle->SetHistLineColor(kAzure + 10);
16
17     // Opening file with generated data
18     TFile* results = new TFile("Output.root", "READ");
19
20     // Getting histos from file
21     TH1F* hPType = (TH1F*)results->Get("hPType");
22     TH1F* hPhi = (TH1F*)results->Get("hPhi");
23     TH1F* hTheta = (TH1F*)results->Get("hTheta");
24     TH1F* hP = (TH1F*)results->Get("hP");
25     TH1F* hInvMass = (TH1F*)results->Get("hInvMass");
26     TH1F* hInvMOpp = (TH1F*)results->Get("hInvMOpp");
27     TH1F* hInvMSame = (TH1F*)results->Get("hInvMSame");
28     TH1F* hInvMPKSame = (TH1F*)results->Get("hInvMPKSame");
29     TH1F* hInvMPKOpp = (TH1F*)results->Get("hInvMPKOpp");
30     TH1F* hInvMDec = (TH1F*)results->Get("hInvMDec");
31
32     // Printing Particles types data and errors
33     std::cout << "Pions+:" << hPType->GetBinContent(1) << "+/-"
34         << hPType->GetBinError(1) << "\n"
35         << "Pions-:" << hPType->GetBinContent(2) << "+/-"
36         << hPType->GetBinError(2) << "\n"
37         << "Kaone+:" << hPType->GetBinContent(3) << "+/-"
38         << hPType->GetBinError(3) << "\n"
39         << "Kaons-:" << hPType->GetBinContent(4) << "+/-"
40         << hPType->GetBinError(4) << "\n"
41         << "Protons+:" << hPType->GetBinContent(5) << "+/-"
42         << hPType->GetBinError(5) << "\n"
43         << "Protons-:" << hPType->GetBinContent(6) << "+/-"
44         << hPType->GetBinError(6) << "\n"
45         << "K*" << hPType->GetBinContent(7) << "+/-"
46         << hPType->GetBinError(7) << "\n";
47
48     // Canvas for distribution histos
49     TCanvas* cDis = new TCanvas("cDis", "Distributions measured", 1500, 1000);
50     cDis->Divide(2, 2);
51
52     // Drawing histos and fits
53     cDis->cd(1);
54     hPType->DrawCopy();
55
56     cDis->cd(3);

```

```

57 hPhi->Fit("pol0");
58 hPhi->DrawCopy();
59
60 cDis->cd(4);
61 hTheta->Fit("pol0");
62 hTheta->DrawCopy();
63
64 cDis->cd(2);
65 hP->Fit("expo");
66 hP->DrawCopy();
67
68 // Canvas for K* masses histos
69 TCanvas* cMass = new TCanvas("cMass", "K* Masses", 3000, 500);
70 cMass->Divide(3, 1);
71
72 // Analysis of invariant mass histos
73 // Two new histos are created subtracting same and opposite charge histos
74
75 // Pions and Kaons histo
76 TH1F* hSubPK{new TH1F("hSubPK",
77     "Subtraction of invariant mass of Kaons and Pions of "
78     "Same and Opposite charge",
79     1000, 0, 5)};
80
81 // Fills, fits and draws histo
82 cMass->cd(3);
83 hSubPK->Add(hInvMPKOpp, hInvMPKSame, 1, -1);
84 hSubPK->Fit("gaus", "", "", 0.5, 1.5);
85 hSubPK->SetXTitle("Mass [GeV/C^2]");
86 hSubPK->SetYTitle("Occurrences");
87 hSubPK->SetAxisRange(0.65, 1.4);
88 hSubPK->DrawCopy();
89
90 // All particles histo
91 TH1F* hSub{new TH1F(
92     "hSub",
93     "Subtraction of invariant mass of particles of Same and Opposite charge",
94     1000, 0, 5)};
95
96 // Fills, fits and draws histo
97 cMass->cd(2);
98 hSub->Add(hInvMOpp, hInvMSame, 1, -1);
99 hSub->Fit("gaus", "", "", 0.5, 1.5);
100 hSub->SetXTitle("Mass [GeV/C^2]");
101 hSub->SetYTitle("Occurrences");
102 hSub->SetAxisRange(0.65, 1.4);
103 hSub->DrawCopy();
104
105 // Fits and draws histo of invariant masses of particles created by decayment
106 cMass->cd(1);
107 hInvMDec->Fit("gaus");
108 hInvMDec->SetAxisRange(0.6, 1.2);
109 hInvMDec->SetFillColor(kCyan);
110 hInvMDec->SetLineColor(kAzure + 10);
111 hInvMDec->DrawCopy();
112
113 // Print canvases onto png files

```

```
114 | cMass->Print("Comparison.png");
115 | cDis->Print("Distributions.png");
116 |
117 | // Close file
118 | results->Close();
119 | }
```