

Exercício 7

Professores: Mauricio, Eliza e Sandro

Name: Dalmo da Silva Dalto

EXERCICIO 1

O pico de escolha é o Ψ' como mostrado na Figura 1. A Figura 2 e Figura 3 apresentam o resultados encontrados nos fits com o cut de $P_t < 10$ e cut $P_t < 8$ e $\eta < 2.4$, podendo-se concluir que o yield diminui devido ao cut com o P_t e o η . Isso ocorre devido a restrição de pseudo-rapidez, pois o cut seleciona um número menor de entradas que inteferem nessa interação. Entretanto, o teste do χ^2 permanece compatível junto a massa invariante.

```

1 {
2
3 //Modificacoes no arquivo.C para selecionar o pico psi
4 a._mmin = 3.5;
5 a._mmax = 3.9;
6 extraString = "Psi"; // NOTE: this is used in file name:
7 RooChi2Var chi2_var("chi2_var", "chi2_var", *model, dh, RooFit::DataError(RooAbsData
8   ::SumW2), RooFit::NumCPU(1));
9 double chi2 = chi2_var.getVal();
10 int ndf = frame->GetNbinsX() - model->getParameters(dh)->getSize(); // Number of bins
11   minus number of parameters
12 L.DrawLatex(0.13,0.85,"Dimuon Spectrum");
13 L.SetTextSize(0.03);
14 L.DrawLatex(0.13,0.80,"resonance: #psi'");
15 L.DrawLatex(0.13,0.75,Form("#chi^2: %5.3f",chi2/ndf));

```

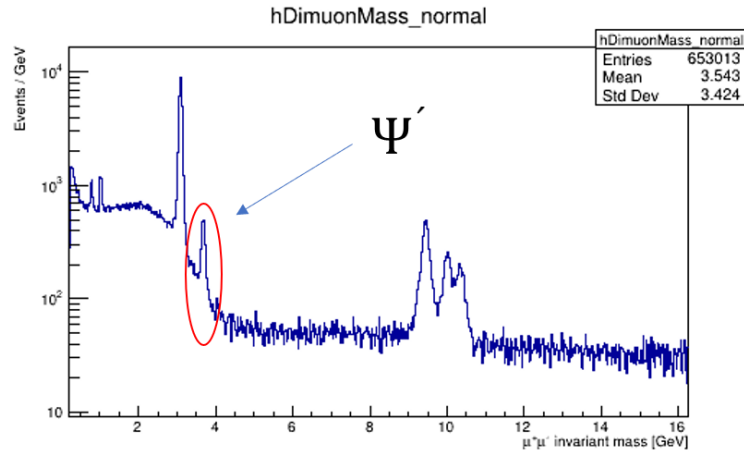


Figura 1: Pico de escolha para os fits

```

160
161 Int_t anadimuon::Cut(Long64_t entry)
162 {
163     // This function may be called from Loop.
164     // returns 1 if entry is accepted.
165     // returns -1 otherwise.
166
167     double mass = dimuon_p4->M();
168     double pt1 = muonN_p4->Pt();
169     double pt2 = muonP_p4->Pt();
170     if(pt1<10 || pt2<10 ) return -1;
171     if(mass>200) return -1;
172     //cout << "Cut mass:" << mass << endl;
173
174     return 1;
175 }

```

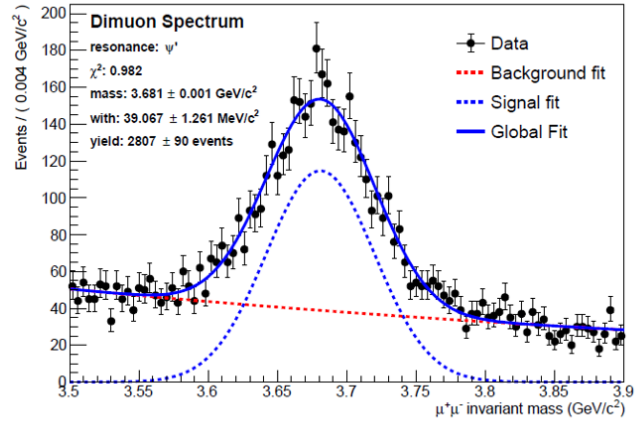


Figura 2: Resultado do ajuste sem o cut de Pt e pseudo-rapidez

```

Int_t anadimuon::Cut(Long64_t entry)
{
    // This function may be called from Loop.
    // returns 1 if entry is accepted.
    // returns -1 otherwise.

    double mass = dimuon_p4->M();
    double pt1 = muonN_p4->Pt();
    double pt2 = muonP_p4->Pt();
    double rapidity = dimuon_p4->Rapidity();
    if(pt1<8 || pt2<8 && rapidity<2.4 ) return -1;
    if(mass>200) return -1;
    //cout << "Cut mass:" << mass << endl;

    return 1;
}

#endif // #ifdef anadimuon_cxx

```

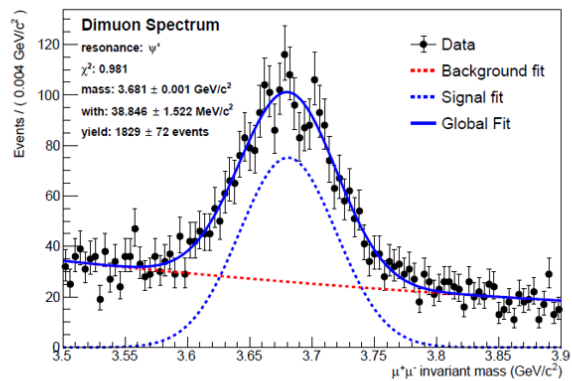


Figura 3: Resultado do ajuste com o cut de Pt e pseudo-rapidez