RECREACIÓN DE GRAFICAS DE PAPER SOBRE 3 MESONES B

Josue Daniel Jaramillo Arroyave, Tomas Sosa Giraldo, Juan Jose Montoya Sanchez

UNIVERSIDAD DE ANTIOQUIA

INTRODUCCIÓN

"**S**olenoide **C**ompacto de **M**uones"

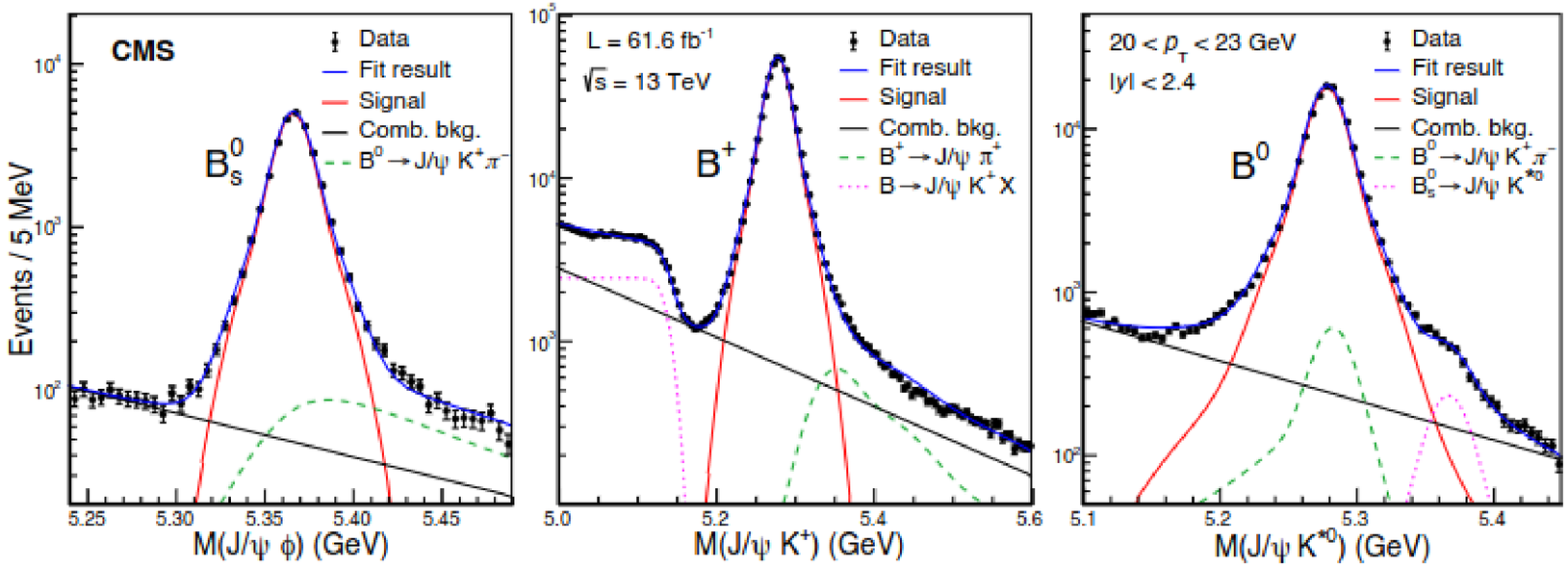
Uno de los dos detectores de partículas de propósito general del Gran Colisionador de Hadrones, que colisiona haces de protones en el CERN,.

CERN-EP-2022-248
2022/12/06

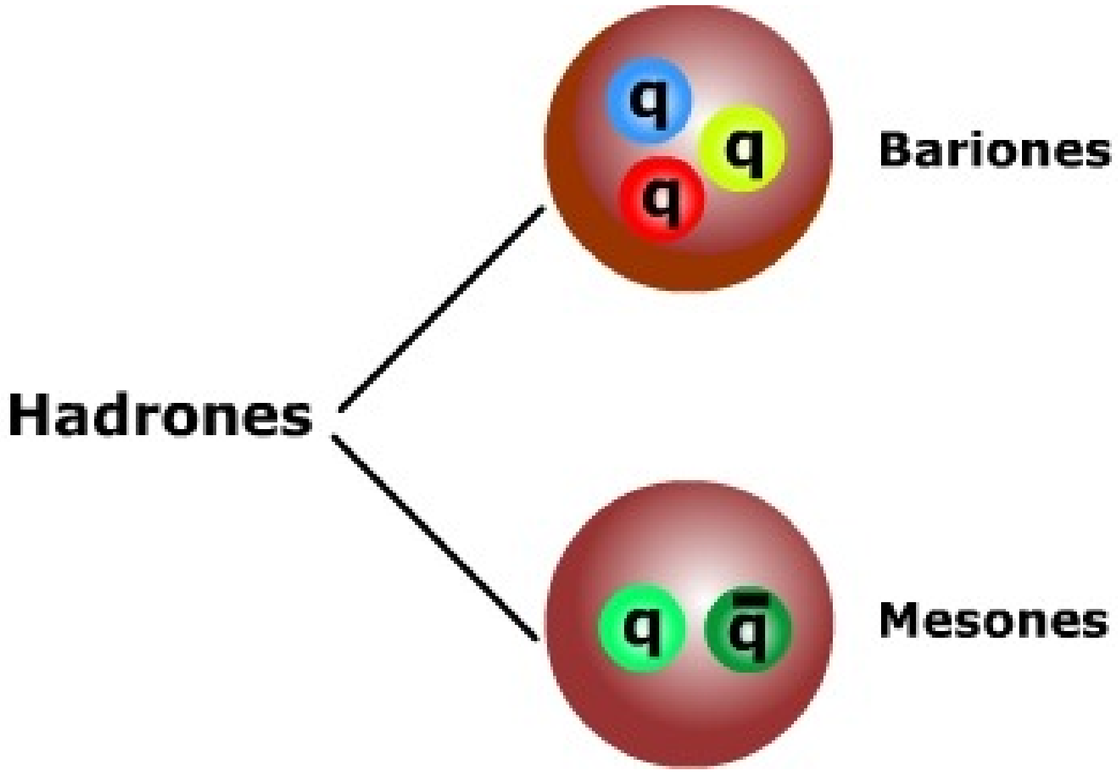
CMS-BPH-21-001

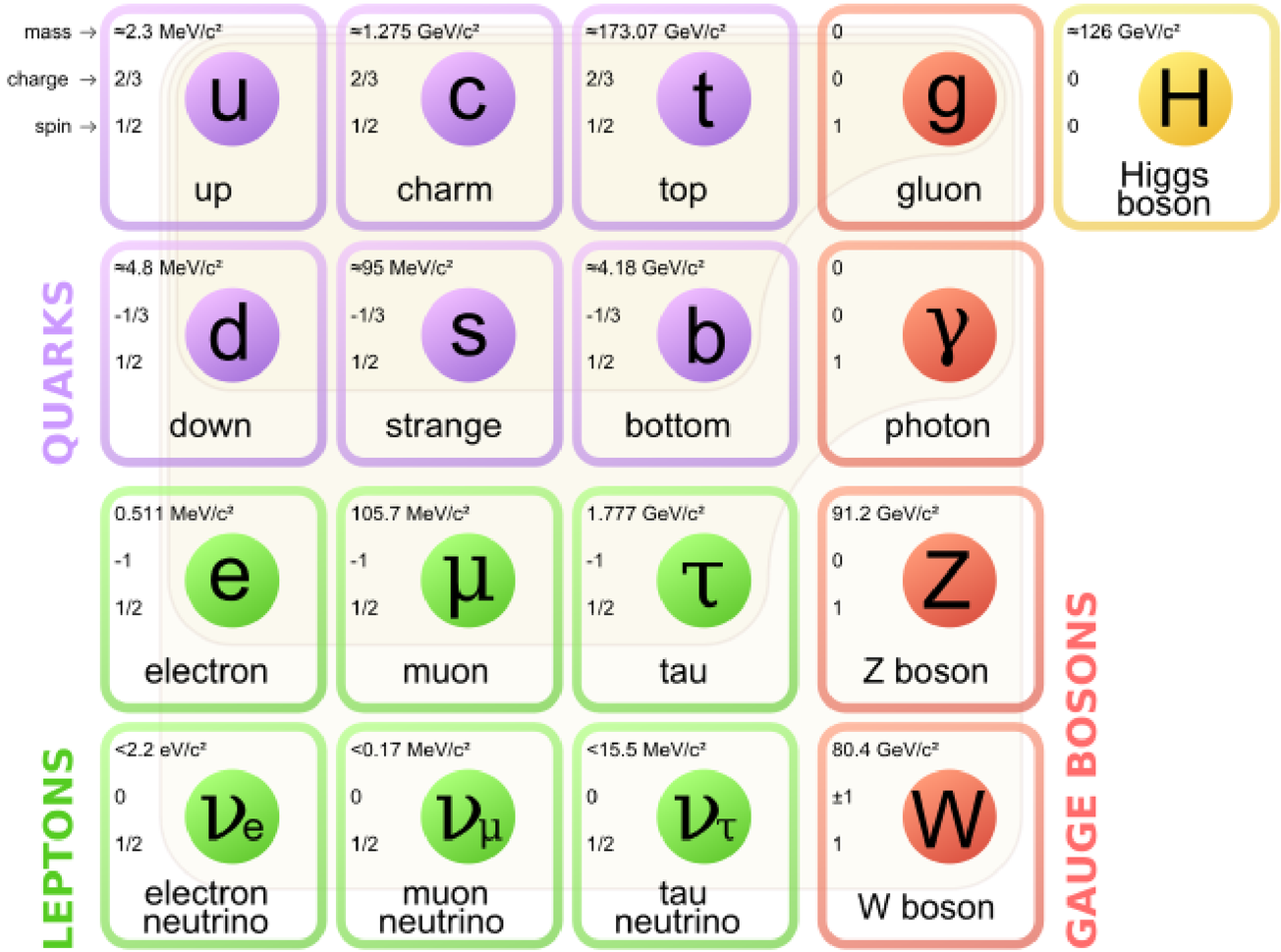
Measurement of the dependence of the hadron production fraction ratio f_s/f_u on B meson kinematic variables in proton-proton collisions at $\sqrt{s} = 13\,\text{TeV}$

The CMS Collaboration



MODELO ESTÁNDAR





B+ Meson (B+): Estos mesones estan hechos de un quark up y un antiquark bottom.
Tienen carga de +1.

B0 Meson (B0): Compuesto de un quark down y un antiquark bottom, estos mesones tienen una carga de 0, siendo neutrales.

Bs Meson (Bs): Este tipo de meson B esta hecho de un quark strange y un antiquark bottom.
Son partículas neutras.

Bc Meson (Bc): El meson Bc es un tanto especial comparado al resto. En lugar de emparejar el antiquark bottom con un quark up, down o strange, se empareja con un quark charm.
Tienen una carga de +1

DECAIMIENTOS

$B^0_S \rightarrow J/\Psi\ \Phi$

$B^+ \rightarrow J/\Psi\ K^+$

$B^0 \rightarrow J/\Psi\ K^{*0}$

$J/\psi \rightarrow \mu^+\mu^-$

$\Phi \rightarrow K^+K^-$

CORTES

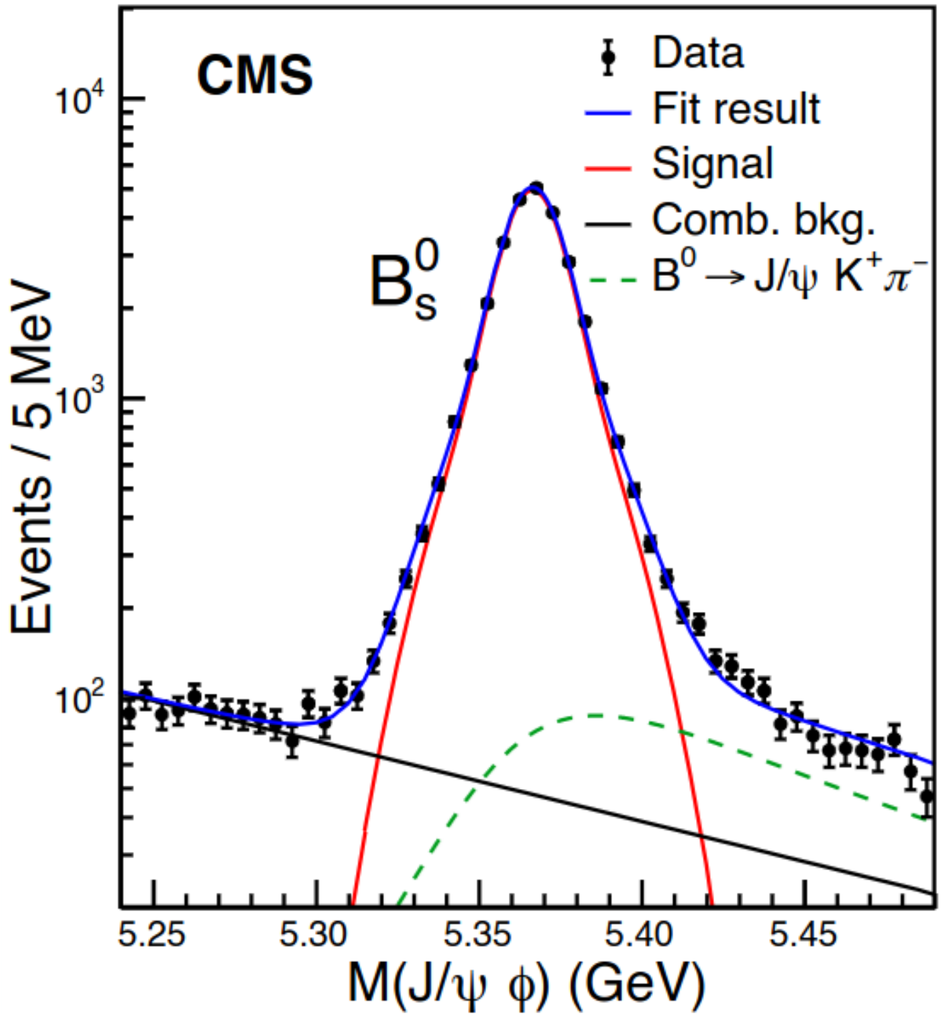
$M(J/\psi) = 3{,}0969\ GeV$

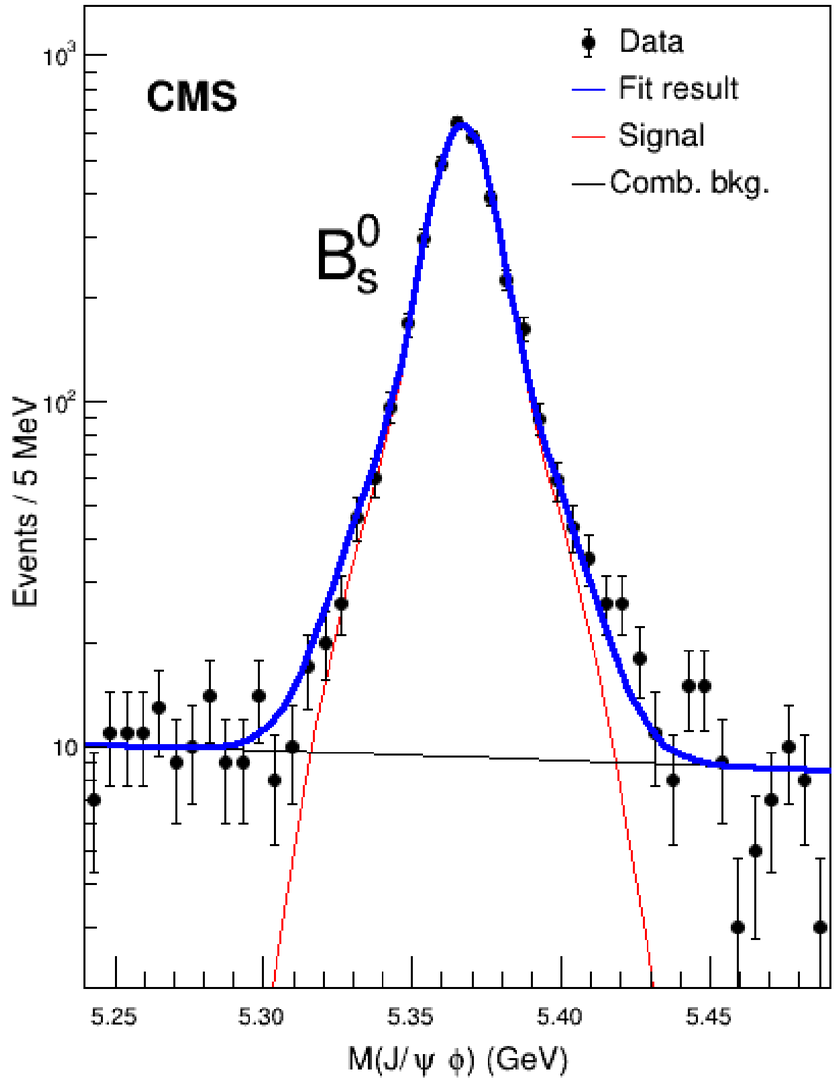
$|\eta(\mu)| < 2.4$

$20\ GeV < Pt(B) < 23\ GeV$

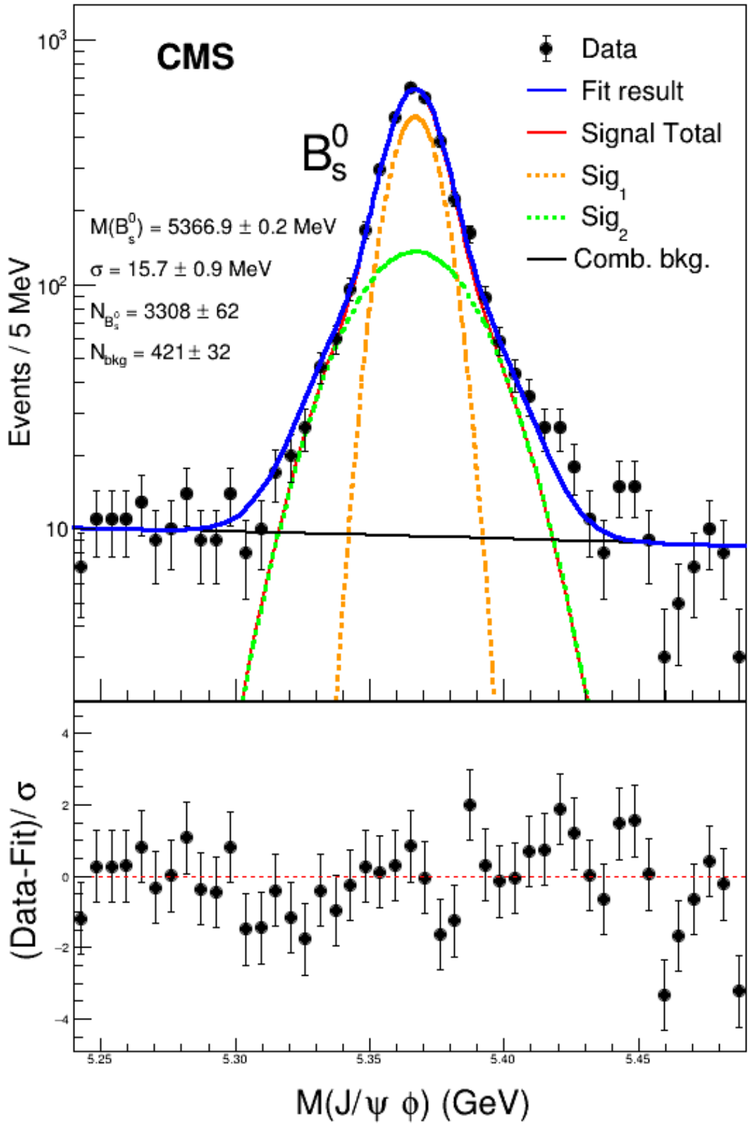
$|\eta(B)| < 2.4$

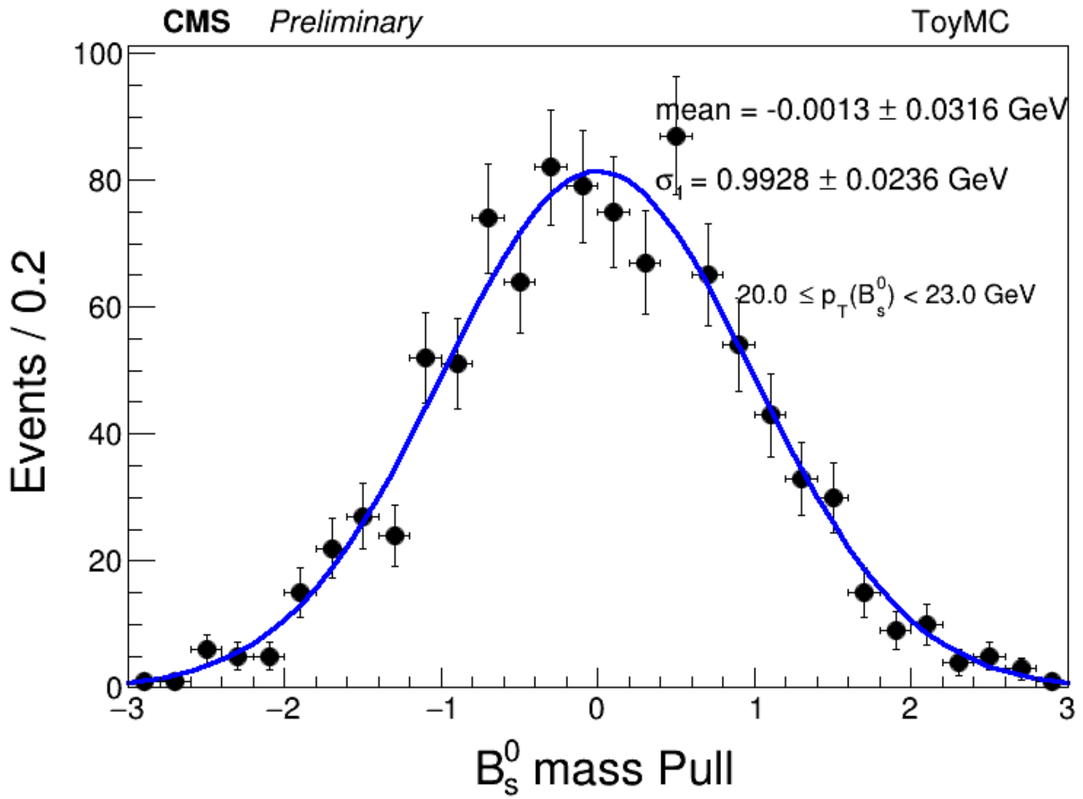
MASA B STRANGE





PULL Y MONTECARLO



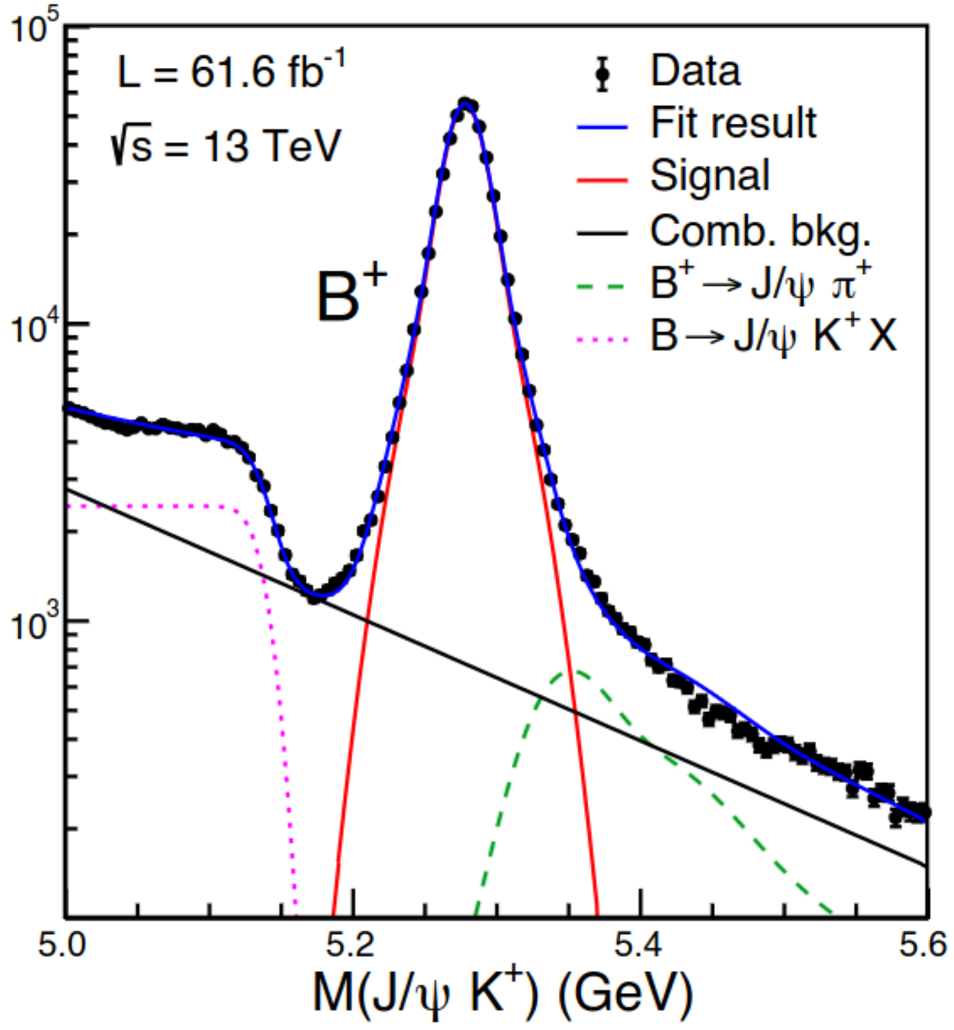


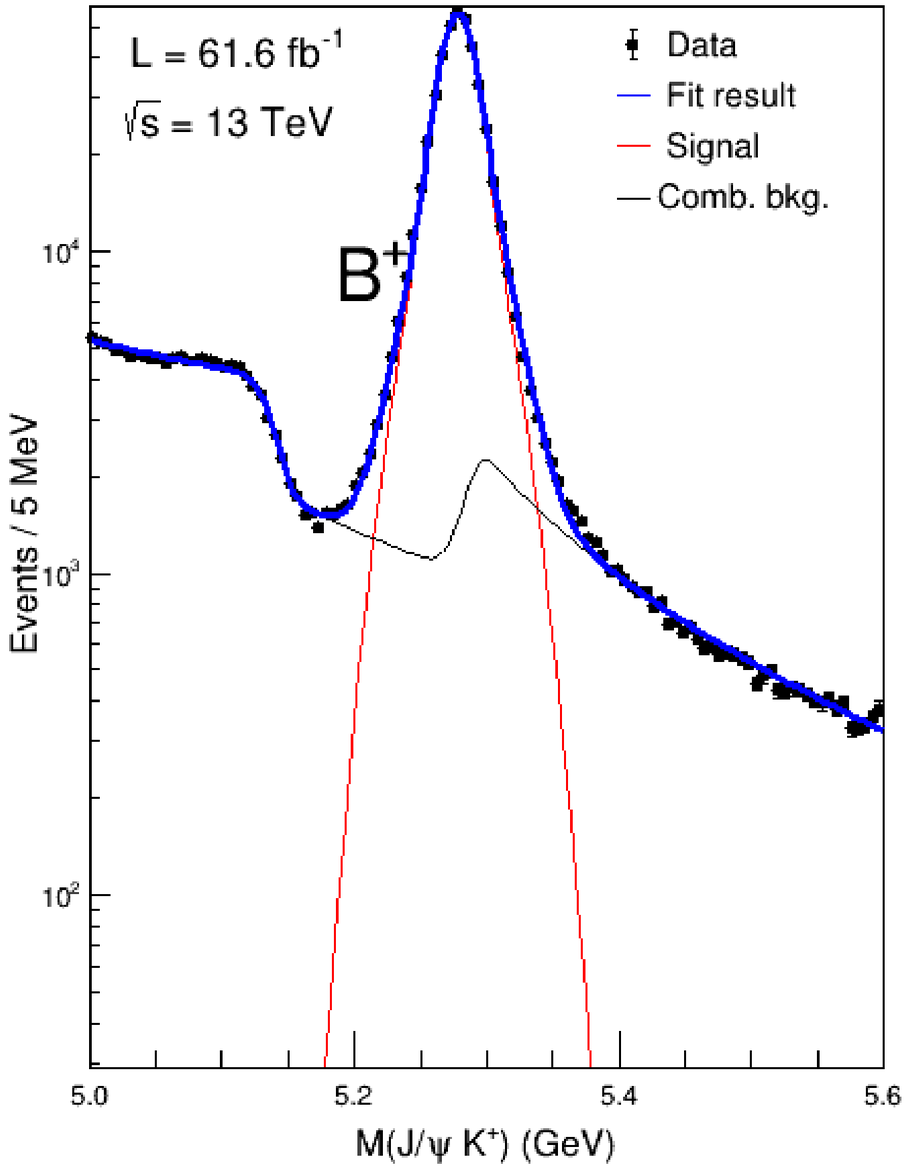
CORTES Y MODELO

```
for(Long64_t jentry=0; jentry<nentries;jentry++)
    {
      Long64_t ientry = t.LoadTree(jentry);
      if (ientry < 0) break;
      nb = t.fChain->GetEntry(jentry);   nbytes += nb;
      if(jentry%nTen==0) cout<<10*(jentry/nTen)<<"%-"<<flush;
      if(jentry==nentries-1) cout<<endl;

      //Mass cuts
      if(t.B_mass<=Mmin || t.B_mass>=Mmax) continue;
      if(t.J_mass<=3.0969-0.150 || t.J_mass>=3.0969+0.150) continue;
      if(t.phi_mass<=1.01946-0.010 || t.phi_mass>=1.01946+0.010) continue;

      if(t.bcpt<ptl || t.bcpt>=pth)continue;

      M=t.B_mass;
      data.add(RooArgSet(M));

    }
```

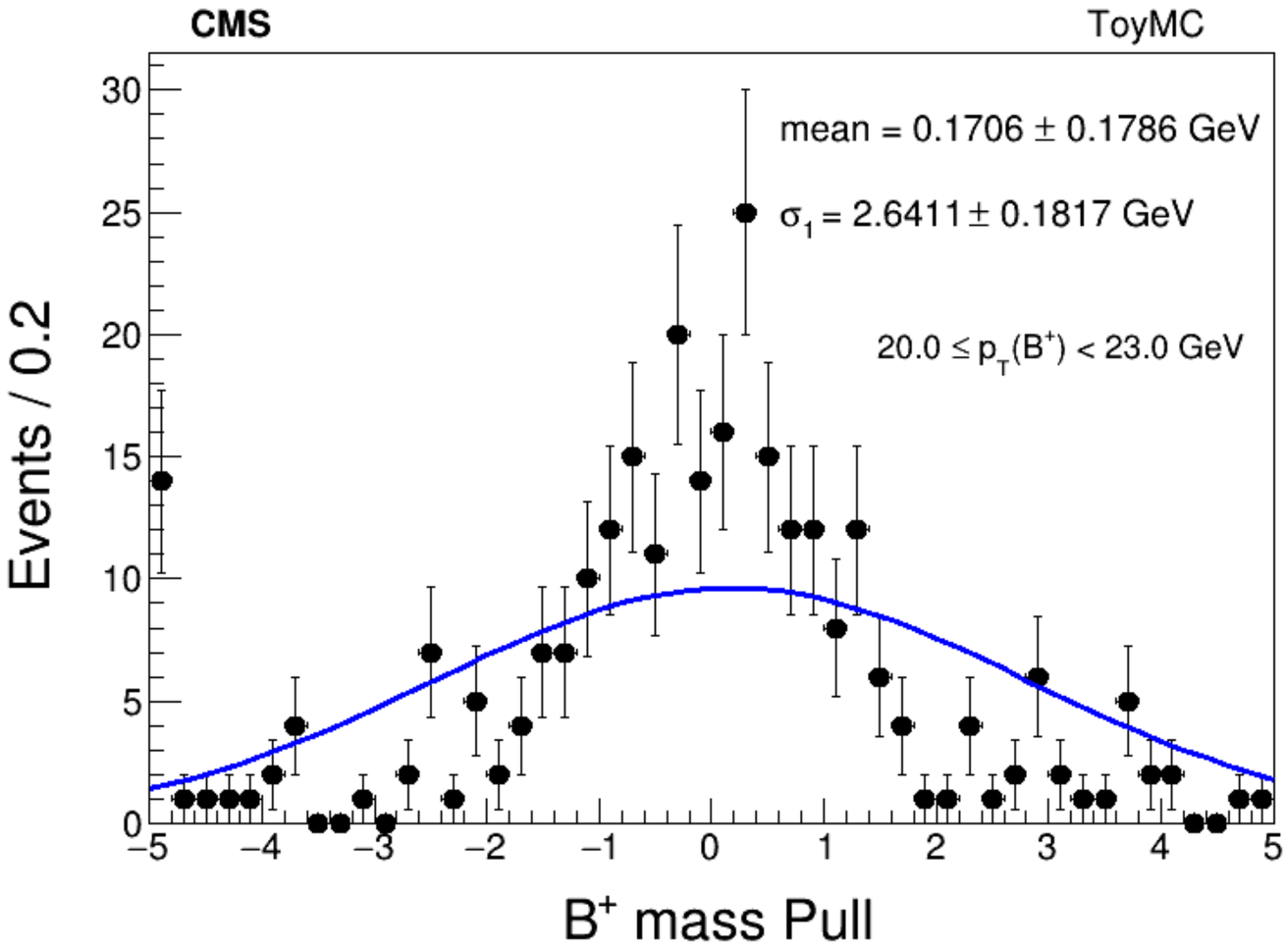
```
// ****define background ****
RooRealVar a0("a0","a0",0.23,-10.0,10.0);
RooChebychev bkg("bkg","Background",M,RooArgList(a0));

//gaussians
RooRealVar mean("mean"," Mass mean",5.366,5.320,5.400,"GeV");
RooRealVar width("width"," Mass width",0.010,0.001,0.015,"GeV");
RooGaussian Sig("Sig"," Signal PDF",M,mean,width);

RooRealVar width2("width2"," Mass width2 ",0.020,0.015,0.05,"GeV");
RooGaussian Sig2("Sig2"," Signal PDF B",M,mean,width2);

//********final PDF ********
RooRealVar Ns("Ns","Ns",0.,5400);
RooRealVar Nb("Nb","Nb",0.,5400);
RooRealVar fs("fs","fs",0.8,0.,1.);

RooAddPdf sumgau("sumgau","sumgau",RooArgList(Sig,Sig2),RooArgList(fs));
//model
RooAddPdf MassModel("MassModel","MassModel",RooArgList(sumgau,bkg),RooArgList(Ns,Nb));
```

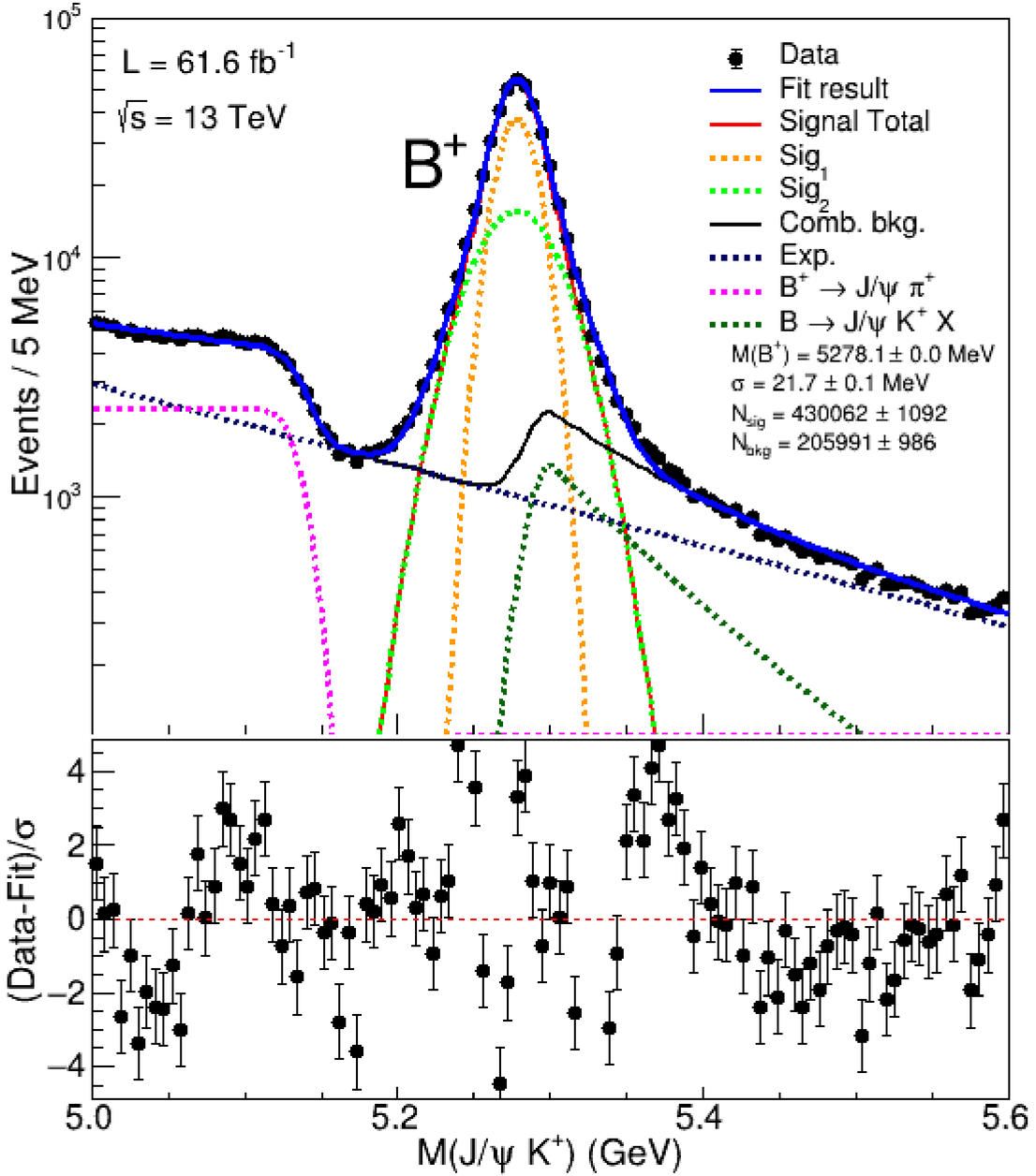
MASA B+


L = 61.6 fb-1
√s = 13 TeV
B+
Data
Fit result
Signal
Comb. bkg.
B+ → J/ψ π+
B → J/ψ K+ X
M(J/ψ K+) (GeV)


L = 61.6 fb-1
√s = 13 TeV
B+
Data
Fit result
Signal
Comb. bkg.
Events / 5 MeV
M(J/ψ K+) (GeV)

PULL Y MONTECARLO



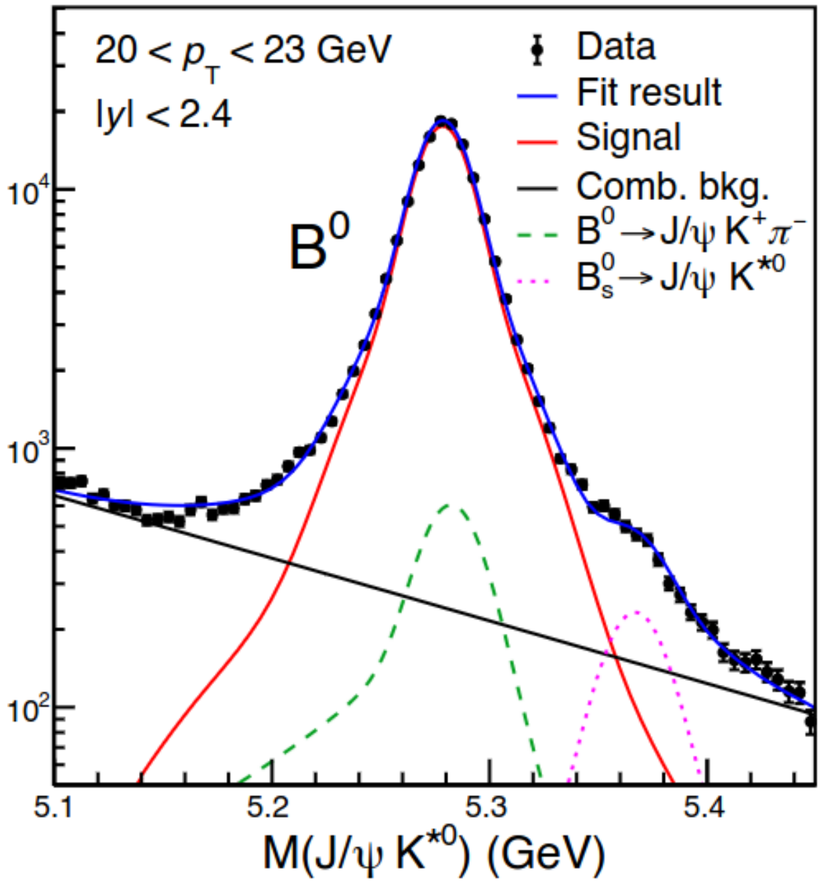


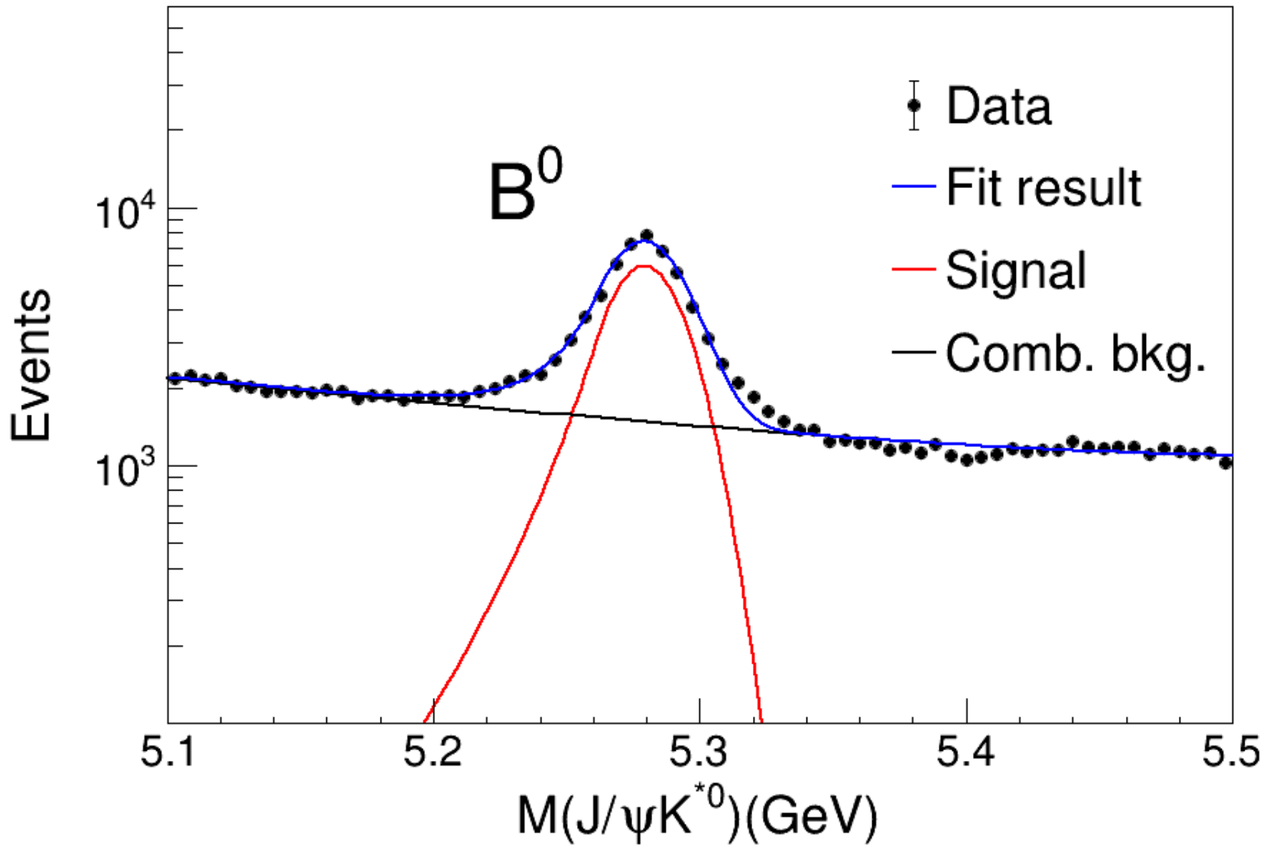
CORTES Y MODELO

```
for(Long64_t jentry=0; jentry<nentries;jentry++)
    {
    Long64_t ientry = tree->LoadTree(jentry);
    if (ientry < 0) break;
    nb = tree->GetEntry(jentry);   nbytes += nb;
    if(jentry%nTen==0) cout<<10*(jentry/nTen)<<"%-"<<flush;
    if(jentry==nentries-1) cout<<endl;

    //Mass windows cuts
    if(B_mass<=Mmin || B_mass>=Mmax) continue;
    if(J_mass<=3.0969-0.150 || J_mass>=3.0969+0.150) continue;
    if(mu1pt<4.0) continue;
    if(mu2pt<4.0) continue;
    if(bcpt<ptl || bcpt>=pth)continue;

    if( abs(mu1eta)>2.4)continue;
    if( abs(mu2eta)>2.4)continue;
    if( abs(Brapidity)>2.4) continue;
    if(trijk>10.0 || trijk<1.0)continue;
    //if(trijk>10.0)continue;
    if(abs(dxysig1)<2.0)continue;
    m=B_mass;
    data.add(RooArgSet(m));
    }
```

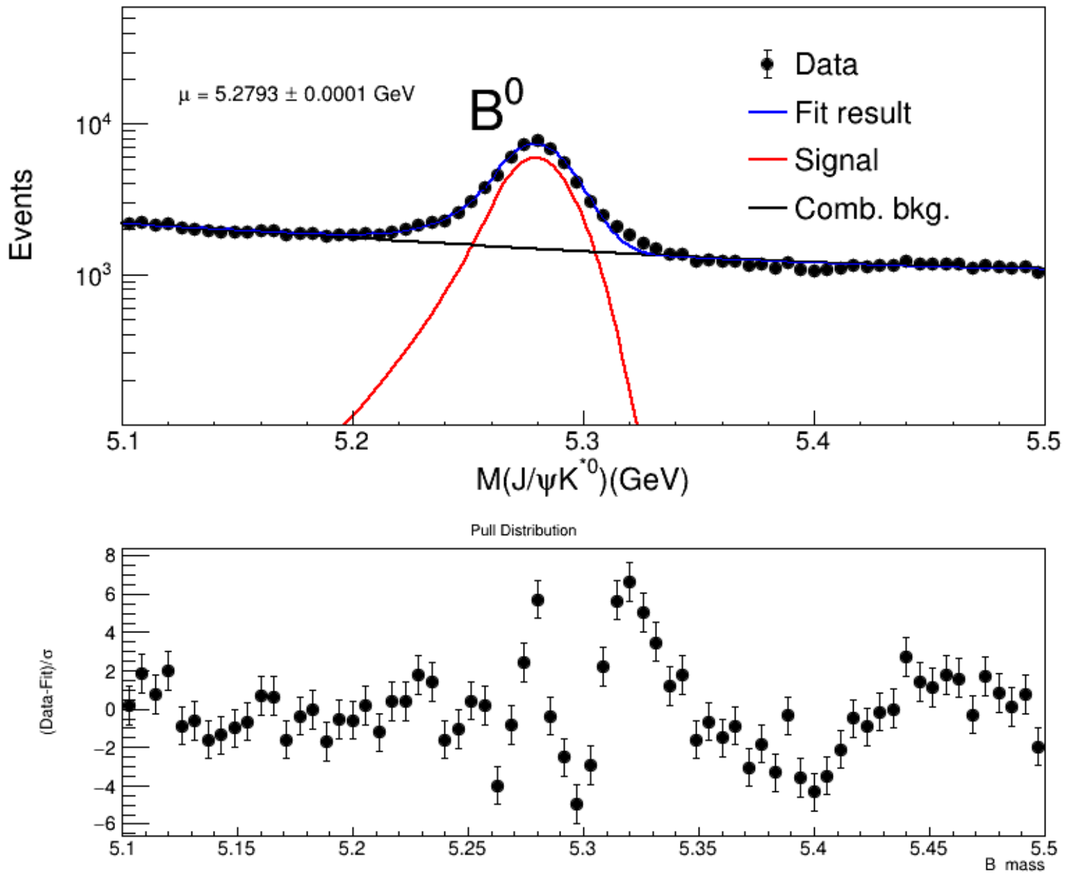
```
// Parametros
RooRealVar mean("mean","mean of gaussians",5.279,5.2,5.32);
RooRealVar sigma1("sigma1","width of gaussians",0.010,0.001,0.015);
RooRealVar sigma2("sigma2","width of gaussians",0.020,0.015,0.05);
// Gaussianas
RooGaussian sig1("sig1","Signal component 1",m,mean,sigma1);
RooGaussian sig2("sig2","Signal component 2",m,mean,sigma2);
// Suma de las gaussianas
RooRealVar sig1frac("sig1frac","fraction of component 1 in signal",0.8,0.,1.);
RooAddPdf sig("sig","Signal",RooArgList(sig1,sig2),RooArgList(sig1frac));
// Background model
// ERF
RooRealVar d0("d0","d0",0.035,0.01,0.085);
RooRealVar d1("d1","d1",5.14,5.05,5.2);
RooGenericPdf genpdf("genpdf","genpdf","(TMath::Erf((-m + d1)/d0)+1)",RooArgSet(d0,d1,m));
// Exponencial
RooRealVar c("c","c",-10.0,10.0);
RooExponential exp("exp","Exp. Background",m,c);
// Suma de la ERF y la exponencial
RooRealVar bkg0frac("bkg0frac","fraction of component 0 in background",0.8,0.,1.);
RooAddPdf bkg_("bkg_","Background_",RooArgList(genpdf,exp),RooArgList(bkg0frac));
// Crystal Ball function
RooRealVar mean2("mean2","mean of gaussian",5.355,5.3,5.4) ;
RooRealVar sigma3("sigma3","width of gaussian",0.010,0.001,0.015) ;
RooRealVar alpha("alpha", "alpha", -1, -10.0, 0.0);
RooRealVar n("n", "n", 3, 0, 10);
RooCBShape sig3("sig3", "Signal component 3", m, mean2, sigma3, alpha, n);
// Sumas de la Crystal ball con el resto del background
RooRealVar bkg1frac("bkg1frac","fraction of component 1 in background",0.8,0.,1.);
RooAddPdf bkg("bkg","Background",RooArgList(bkg_,sig3),RooArgList(bkg1frac)) ;
// Numero de eventos
RooRealVar nsig("nsig","number of signal events",510000,0.,800000) ;
RooRealVar nbkg("nbkg","number of background events",255000,0.,800000) ;
// Creamos el modelo y lo fiteamos con la data
RooAddPdf  model("model","sig+bkg", RooArgList(sig,bkg), RooArgList(nsig,nbkg));
```

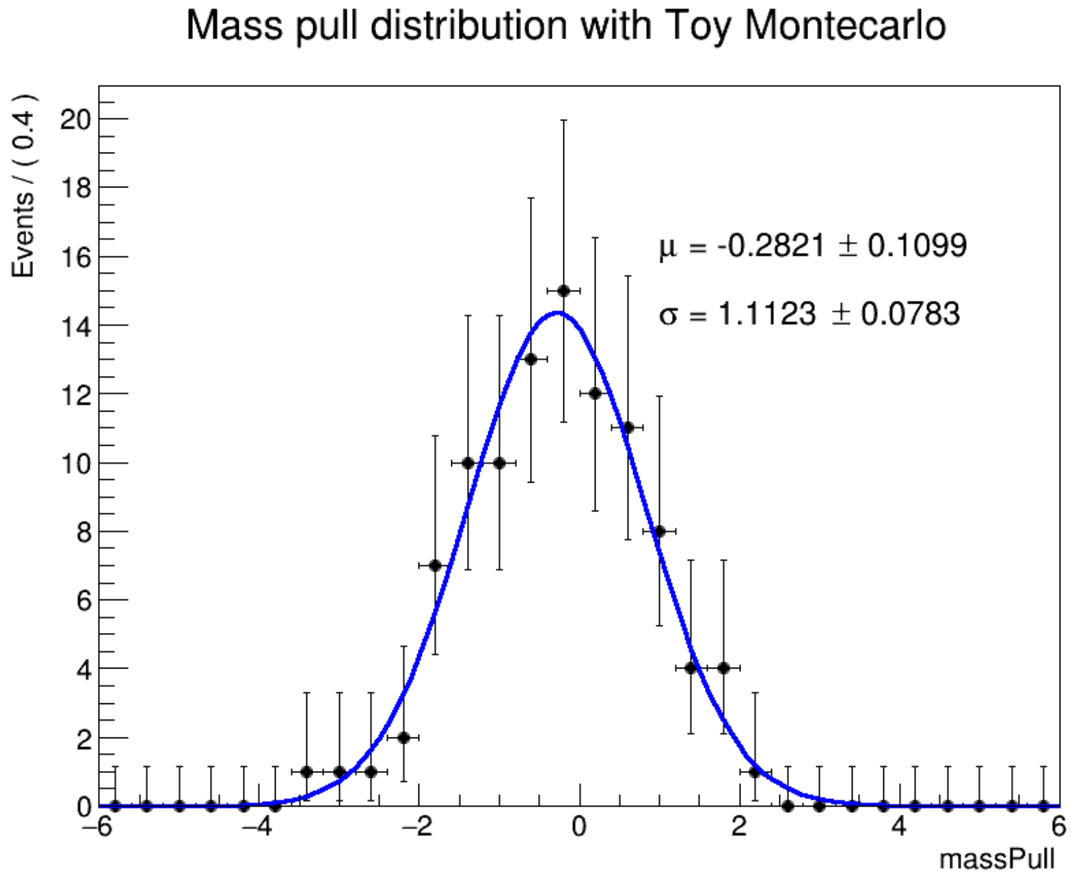
MASA B^0


20 < p_T < 23 GeV
|y| < 2.4
Data
Fit result
Signal
Comb. bkg.
$B^0 \to J/\psi\, K^+\pi^-$
$B_s^0 \to J/\psi\, K^{*0}$
B^0
10^4
10^3
10^2
5.1
5.2
5.3
5.4
M(J/ψ K^{*0}) (GeV)


Events
B^0
Data
Fit result
Signal
Comb. bkg.
10^4
10^3
5.1
5.2
5.3
5.4
5.5
M(J/ψK^{*0})(GeV)

PULL Y MONTECARLO





CORTES Y MODELO

```
for (int i = 0; i < t->GetEntries(); i++) {
  t->GetEvent(i);
  //Mass windows cuts
  if(massB<5.1 || massB>5.5) continue;
  if(massJ<=3.0969-0.150 || massJ>=3.0969+0.150) continue;
  if(Bpt<20.0 || Bpt>23.0) continue;
  if(mu1pt<4.0) continue;
  if(mu2pt<4.0) continue;
  if( abs(mu1eta)>2.4)continue;
  if( abs(mu2eta)>2.4)continue;
  if( abs(Brapidity)>2.4) continue;

  Bm = massB;
  massDS->add(RooArgSet(Bm));
}
```

```
//Signal
RooRealVar mean("mean", "mean", 5.28, 5.1, 5.5);
RooRealVar sigma("sigma", "sigma", .02, 0.0, 0.04);
RooRealVar alpha("alpha", "alpha", 1, 0.0, 5);
RooRealVar n("n", "n", 3, 0, 5);

RooCBShape sig("sig", "sig", Bm, mean, sigma, alpha, n);

//Bkg
RooRealVar a0("a0", "a0", -1,1);
RooRealVar a1("a1", "a1", -1,1);
RooChebychev bkg("bkg", "bkg", Bm, RooArgList(a0, a1));

//Num eventos
RooRealVar nsig("nsig","nsig",1000,0.,148085);
RooRealVar nbkg("nbkg","nbkg",2000,0.,148085);

RooAddPdf model("model","sig+bkg", RooArgList(sig,bkg), RooArgList(nsig,nbkg));
RooFitResult* result = model.fitTo(*massDS, Extended(kTRUE), Save(kTRUE), NumCPU(4));
```

GRACIAS