sPlot for background substraction in B-meson decays

Karol Sowa
AGH UST

Kraków Applied Physics and Computer Science Summer School

Supervisor: Dr hab. inż. Agnieszka Obłąkowska-Mucha

15th September 2020

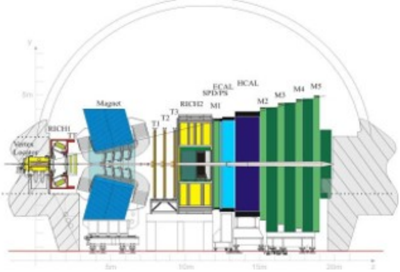
Outline

Introduction

Our project can be considered as a part of **B physics** which is a specialty within the field of particle physics concerned with studying the properties of B hadrons (hadrons containing at least one bottom quark).

Introduction

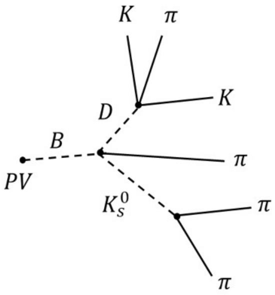
Our project can be considered as a part of **B physics** which is a specialty within the field of particle physics concerned with studying the properties of B hadrons (hadrons containing at least one bottom quark).
Research on this topic are conducted i.a. at LHCb experiment



Introduction

Motivation

Reconstruction of strange K_s^0 meson's mass distribution in the following decay:
$B_s^0 \to D_s^{\mp} K^{*\pm}$ then $K^{*\pm} \to K_s^0 \pi^{\pm}$



Introduction

What did we need to take into consideration?

Introduction

What did we need to take into consideration?

K_s^0 decays through weak interaction $K_s^0 \rightarrow \pi^+\pi^-$ and can travel only several mm in the detector ($c\tau = 2.68$ cm).

Introduction

What did we need to take into consideration?

K_s^0 decays through weak interaction $K_s^0 \to \pi^+\pi^-$ and can travel only several mm in the detector ($c\tau = 2.68$ cm).

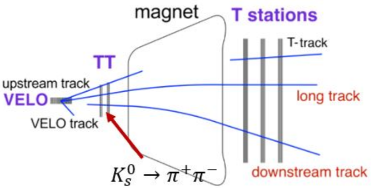
Pions from K_s^0 are usually worse reconstructed since they avoid part of the tracking stations.

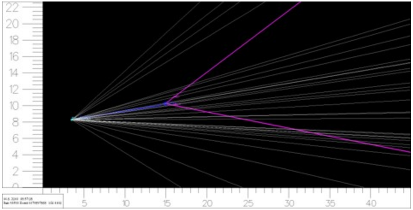
Introduction

What did we need to take into consideration?

K_s^0 decays through weak interaction $K_s^0 \rightarrow \pi^+\pi^-$ and can travel only several mm in the detector ($c\tau = 2.68$ cm).

Pions from K_s^0 are usually worse reconstructed since they avoid part of the tracking stations.





Introduction

Decay $B_s^0 \to D_s^0 K^{*+}$ includes 6 charged particles and pions from our signal $K_s^0 \to \pi^+\pi^-$ can easily be mixed up with pions from similar decay that takes place in PV or with random pions.

Introduction

Decay $B_s^0 \to D_s^0 K^{*+}$ includes 6 charged particles and pions from our signal $K_s^0 \to \pi^+\pi^-$ can easily be mixed up with pions from similar decay that takes place in PV or with random pions.

Aim:

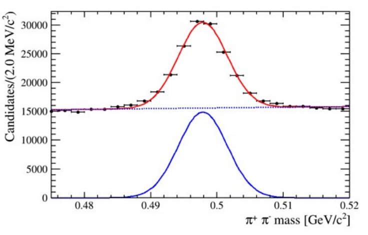
Find differences between „true" $K_s^0 \to \pi^+\pi^-$ - mass plot from signal and other $K_s^0 \to \pi^+\pi^-$ - and random pions.

Introduction

Decay $B_s^0 \to D_s^0 K^{*+}$ includes 6 charged particles and pions from our signal $K_s^0 \to \pi^+\pi^-$ can easily be mixed up with pions from similar decay that takes place in PV or with random pions.

Aim:

Find differences between „true" $K_s^0 \to \pi^+\pi^-$ - mass plot from signal and other $K_s^0 \to \pi^+\pi^-$ - and random pions.
Substract the background from the plot.



RapidSim

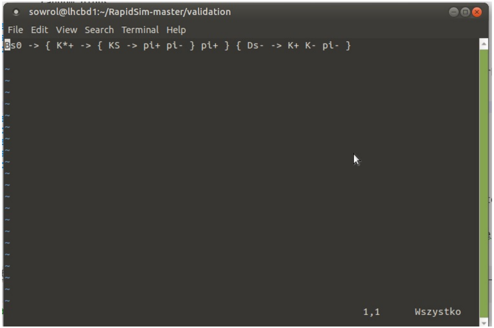
What is it?

Rapid Sim is a fast Monte Carlo generator for simulation of heavy-quark hadron decays.

RapidSim

What is it?

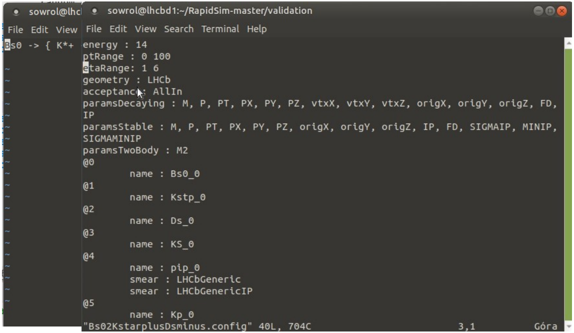
Rapid Sim is a fast Monte Carlo generator for simulation of heavy-quark hadron decays.



RapidSim

What is it?

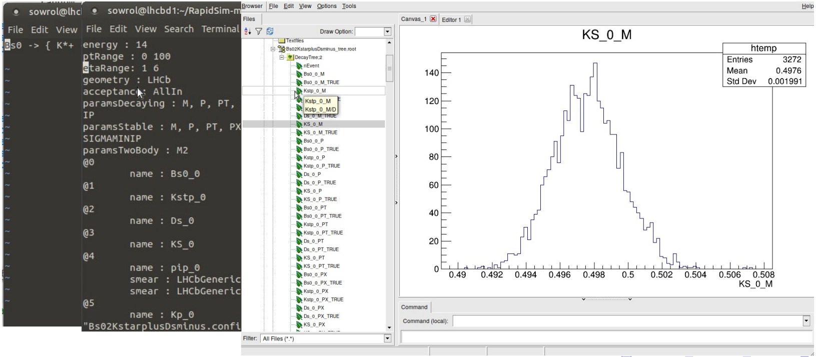
Rapid Sim is a fast Monte Carlo generator for simulation of heavy-quark hadron decays.



RapidSim

What is it?

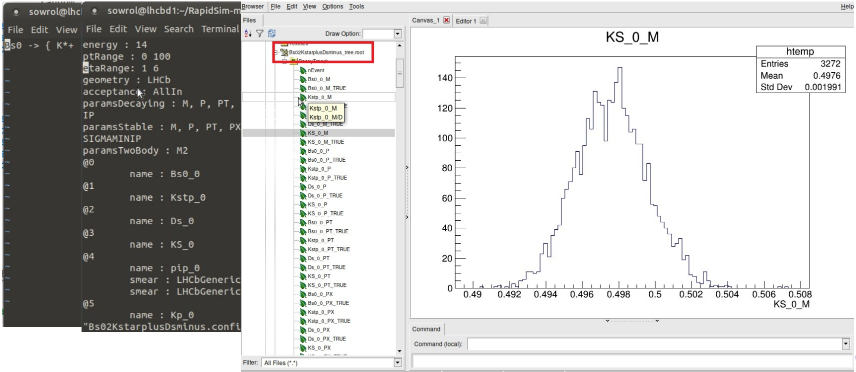
Rapid Sim is a fast Monte Carlo generator for simulation of heavy-quark hadron decays.



RapidSim

What is it?

Rapid Sim is a fast Monte Carlo generator for simulation of heavy-quark hadron decays.



ROOT and RooFit

ROOT

I believe that by now you are all familiar with this CERN framework :)

ROOT and RooFit

ROOT

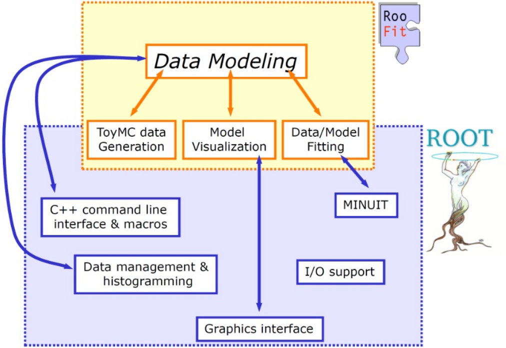
I believe that by now you are all familiar with this CERN framework :)

RooFit

The RooFit library provides a toolkit for modeling the expected distribution of events in a physics analysis. Models can be used to perform unbinned maximum likelihood fits, produce plots, and generate "toy Monte Carlo" samples for various studies.

ROOT and RooFit

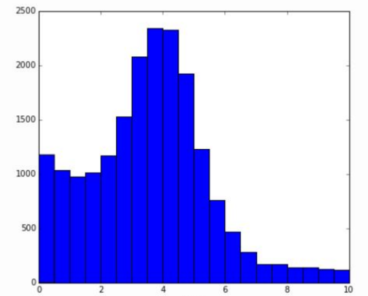
RooFit is an extenstion to ROOT - no overlap with existing functionality

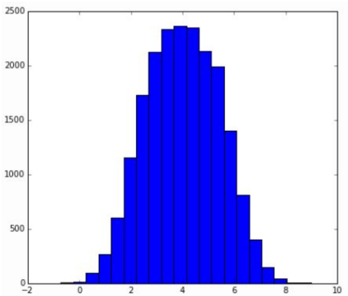


What is sPlot?

sPlot - "Statistical tool to unfold data distributions". It allows us to find the background and signal distribution and therefore separate them

In sPlot we use *discriminating* variable and *control* variable.





What is sPlot?

The aim is thus to use the knowledge available for the discriminating variable to be able to infer the behavior of the control variable.

What is sPlot?

The aim is thus to use the knowledge available for the discriminating variable to be able to infer the behavior of the control variable.

Algorithm

1. Preparing the distributions

What is sPlot?

The aim is thus to use the knowledge available for the discriminating variable to be able to infer the behavior of the control variable.

Algorithm

1. Preparing the distributions
2. Applying standard statistical methods to fit the data. Assessment of signal and background event yields (RooFit).

What is sPlot?

The aim is thus to use the knowledge available for the discriminating variable to be able to infer the behavior of the control variable.

Algorithm

1. Preparing the distributions
2. Applying standard statistical methods to fit the data. Assessment of signal and background event yields (RooFit).
3. Calculating sWeights based on the discriminating variable that refer to the contribution of background and signal in the data

What is sPlot?

The aim is thus to use the knowledge available for the discriminating variable to be able to infer the behavior of the control variable.

Algorithm

1. Preparing the distributions
2. Applying standard statistical methods to fit the data. Assessment of signal and background event yields (RooFit).
3. Calculating sWeights based on the discriminating variable that refer to the contribution of background and signal in the data
4. Using sWeights in MLE to split control variable into signal and background and find their distributions

Plots

We started with data generating, using Rapid-Sim. We generated two sub-decays and tried to find differences:

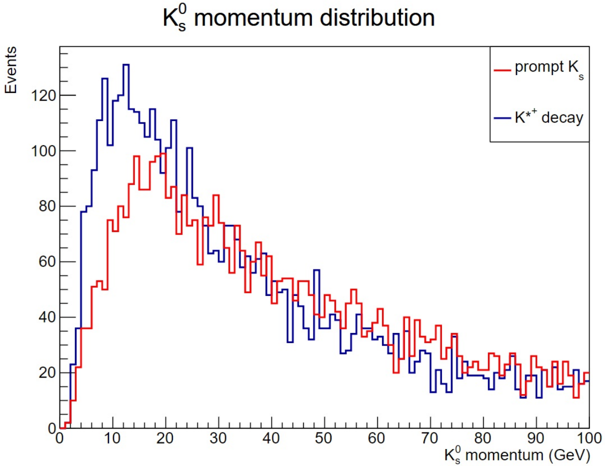
Signal $K^{*+} \to K_s^0 \pi^+$ then $K_s^0 \to \pi^+ \pi^-$

prompt K_s $K_s^0 \to \pi^+ \pi^-$ (proton-proton collisions)

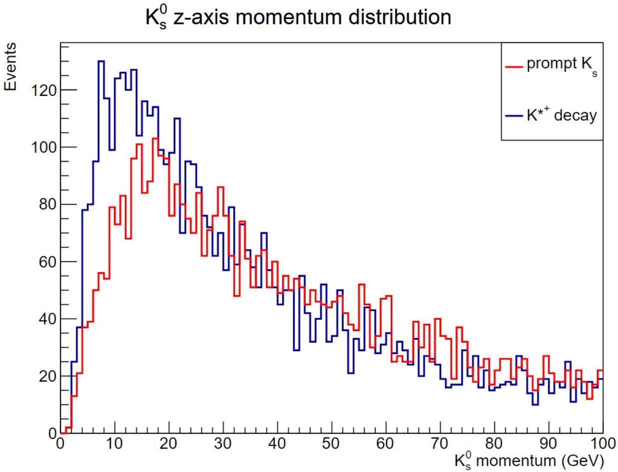
We made plots of various parameters and tried to find our *discriminating variable*

- P - momentum
- PZ - z-axis momentum
- PT - transverse momentum
- FD - flight distance
- IP - impact parameter

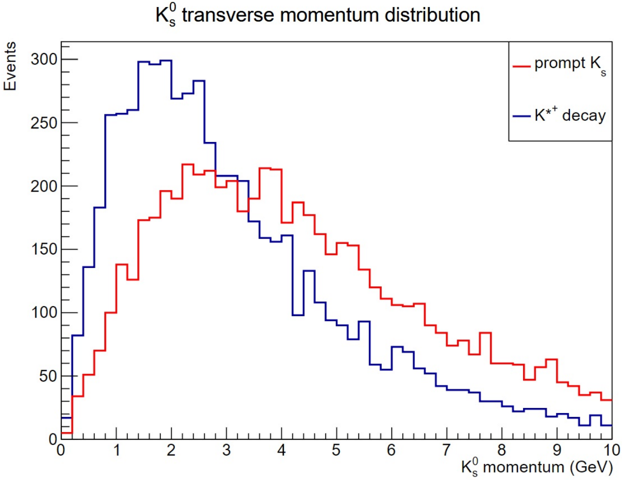
K_s^0 P


K_s^0 momentum distribution
Events
120
100
80
60
40
20
0
10
20
30
40
50
60
70
80
90
100
K_s^0 momentum (GeV)
prompt K_s
K^{*+} decay

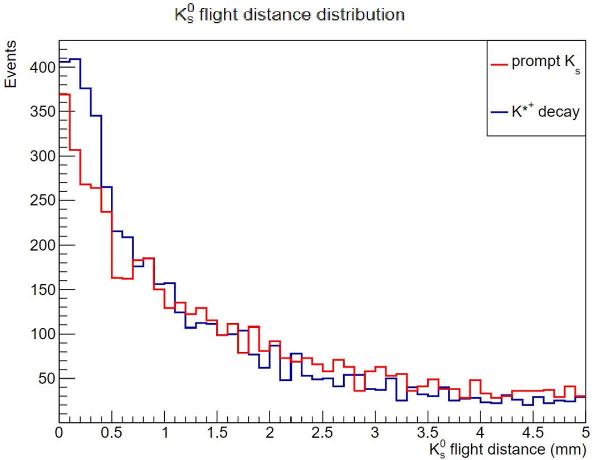
K_s^0 PZ


K⁰s z-axis momentum distribution
Events
prompt Ks
K*+ decay
0
20
40
60
80
100
120
0
10
20
30
40
50
60
70
80
90
100
K⁰s momentum (GeV)

K_s^0 PT


K⁰s transverse momentum distribution
Events
300
250
200
150
100
50
0
0
1
2
3
4
5
6
7
8
9
10
K⁰s momentum (GeV)
prompt Ks
K*+ decay

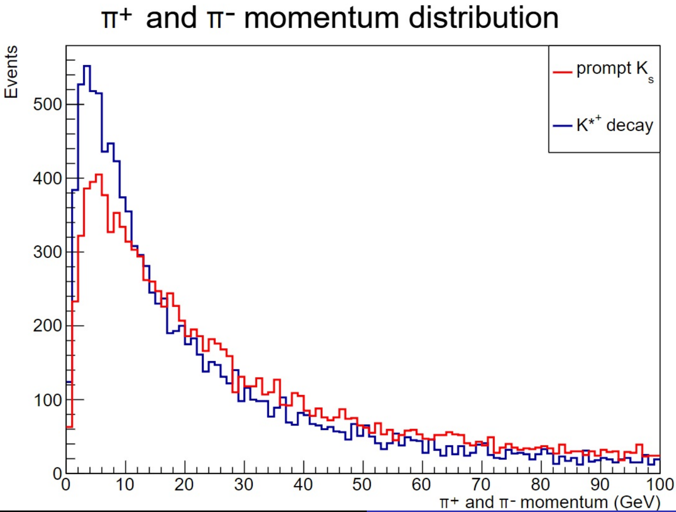
K_s^0 FD


K$_s^0$ flight distance distribution
Events
400
350
300
250
200
150
100
50
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
K$_s^0$ flight distance (mm)
prompt K$_s$
K^{*+} decay

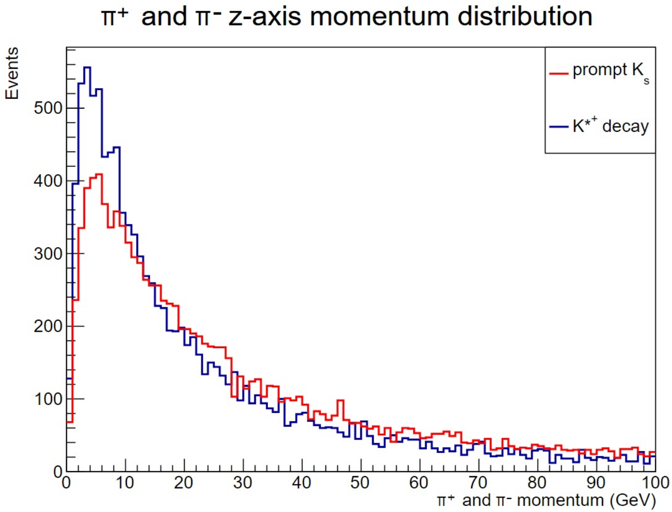
Results

PT plot looked the most promising, but in general - there were only slight differences. So we tried to do the same thing for daughter particles - pions.

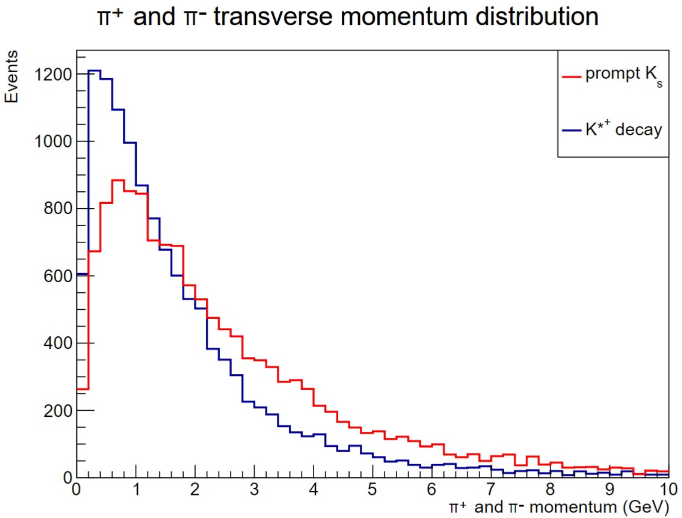
π^+ and π^- P


π+ and π- momentum distribution
Events
prompt K_s
K*+ decay
π+ and π- momentum (GeV)
0
100
200
300
400
500
0
10
20
30
40
50
60
70
80
90
100

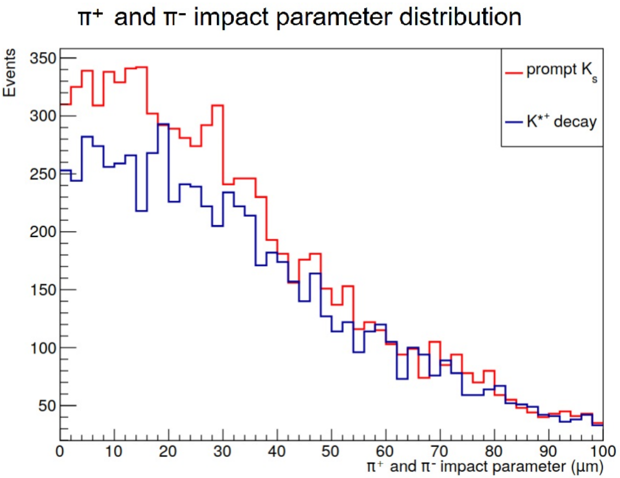
π^+ and π^- PZ


π+ and π- z-axis momentum distribution
Events
prompt Ks
K*+ decay
π+ and π- momentum (GeV)
0
100
200
300
400
500
0
10
20
30
40
50
60
70
80
90
100

π^+ and π^- PT


π⁺ and π⁻ transverse momentum distribution
Events
1200
1000
800
600
400
200
0
0
1
2
3
4
5
6
7
8
9
10
π⁺ and π⁻ momentum (GeV)
prompt K_s
K*⁺ decay

π^+ and π^- IP


π⁺ and π⁻ impact parameter distribution
Events
350
300
250
200
150
100
50
0
10
20
30
40
50
60
70
80
90
100
π⁺ and π⁻ impact parameter (μm)
prompt K_s
K*⁺ decay

Results

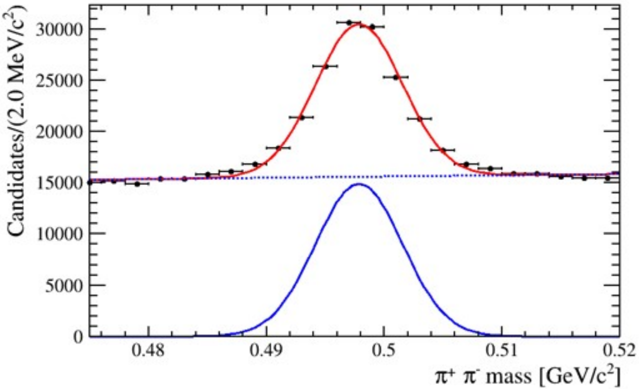
Unfortunately

The differences are still very hard to spot. It seems rather impossible to make a proper fit for background and signal...

So we decided to try something easier - just to test if the method works!

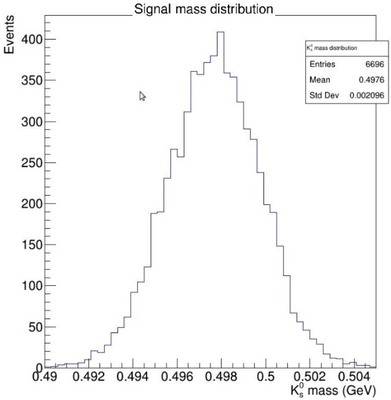
First thing to do

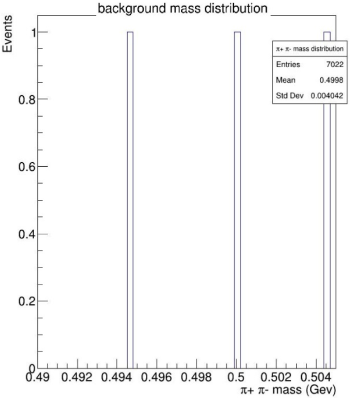
We didn't have any data from the experiment, so we had to prepare them ourselves. Our first objective was to prepare the mass plot similar to this one:



Random pions

At the beginning, we generated single pions in Rapid-Sim.
But the results were rather poor...



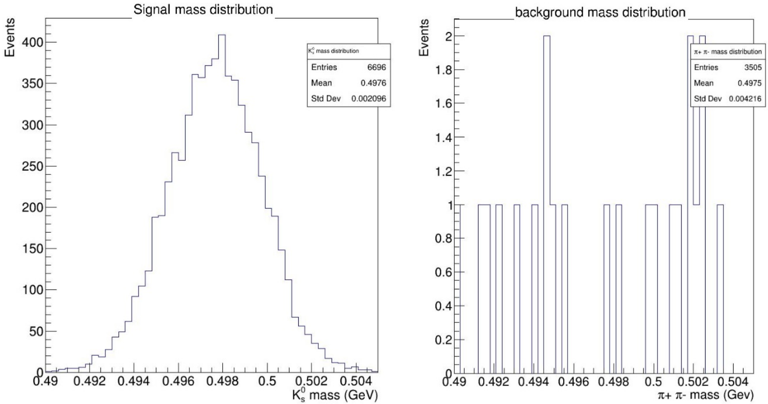


Another decay

So we tried something else - background decay: $B_s^0 \rightarrow \pi^+\pi^-\pi^+D_s^-$

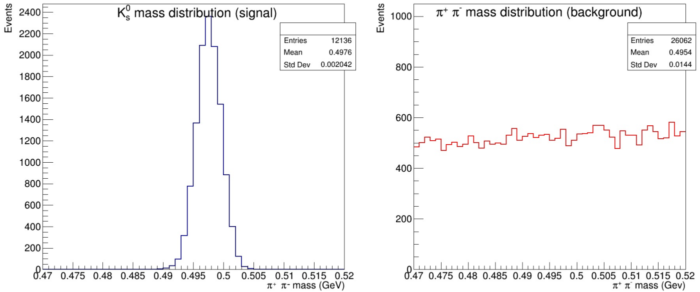
Another decay

So we tried something else - background decay: $B_s^0 \rightarrow \pi^+\pi^-\pi^+D_s^-$
But again - just a few events...



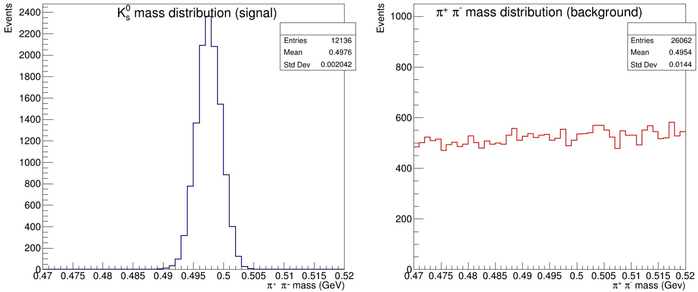
More events

We needed to significantly increase the number of generated events for the background.



More events

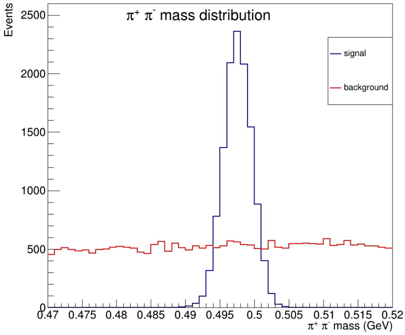
We needed to significantly increase the number of generated events for the background.



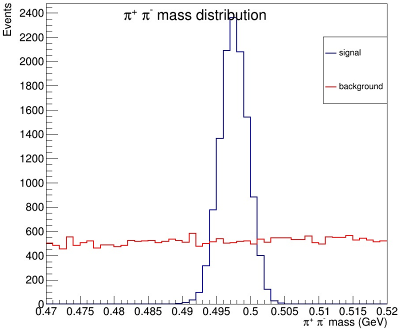
And it worked!

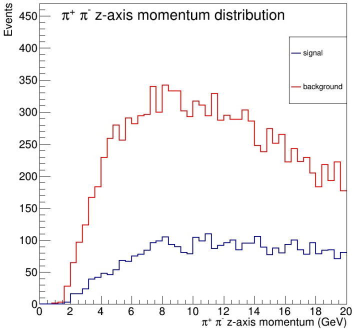
PT

Then, we started looking for a candidate to become our *discriminating variable* (again).

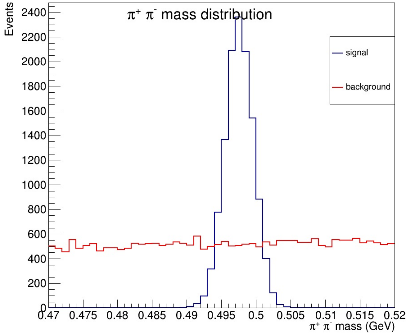


PZ

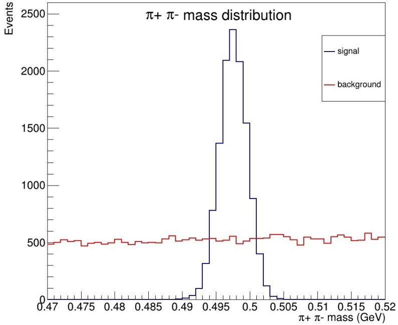

$\pi^+ \pi^-$ mass distribution
Events
signal
background
$\pi^+ \pi^-$ mass (GeV)

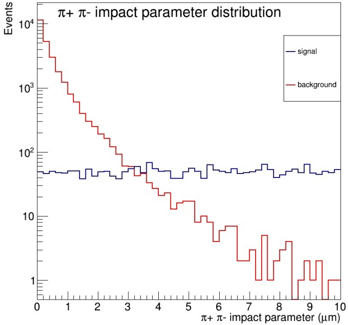

$\pi^+ \pi^-$ z-axis momentum distribution
Events
signal
background
$\pi^+ \pi^-$ z-axis momentum (GeV)

P


Events
$\pi^+ \pi^-$ mass distribution
signal
background
2400
2200
2000
1800
1600
1400
1200
1000
800
600
400
200
0
0.47 0.475 0.48 0.485 0.49 0.495 0.5 0.505 0.51 0.515 0.52
$\pi^+ \pi^-$ mass (GeV)
Events
$\pi^+ \pi^-$ momentum distribution
signal
background
450
400
350
300
250
200
150
100
50
0
0 2 4 6 8 10 12 14 16 18 20
$\pi^+ \pi^-$ momentum (GeV)

IP

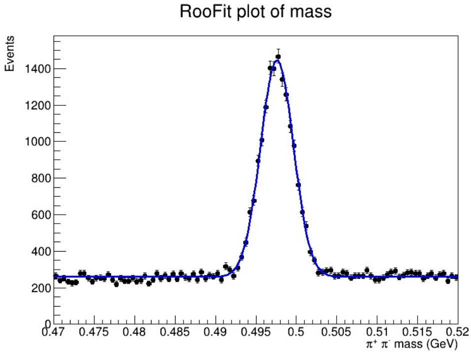




This seems to be the most interesting candidate - significant difference can be spotted when we use logarithmic scale. However, further investigation needs to be done in the future.

"Switch"

We are going to continue our research, but for now we decided to "switch" our variables. Mass became our *discriminating variable*. Signal and background distributions are clearly different what makes it easier to perform ML fit.

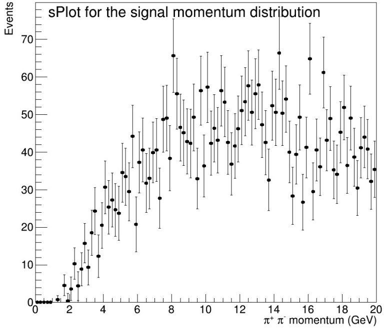


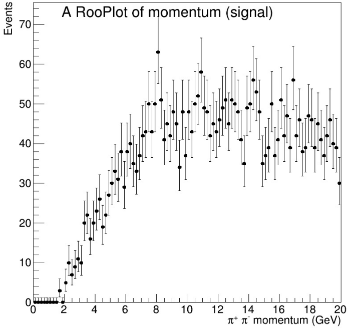
sPlot code

Thanks to RooStats library, performing an sPlot is much easier than it used to be in the past. It contains some extremely helpful tools, like sPlot class and customised functions.

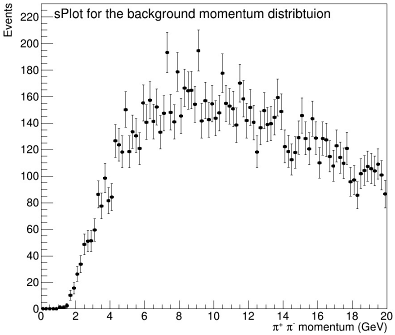
```
SPlot* sDataX = new SPlot("sData","An SPlot", *total_data, &xPdf, RooArgList(nSig, nBG));
cout << endl <<  "Yield of signal is " << nSig.getVal() << ".  From sWeights it is " << sDataX->GetYieldFromSWeight("nSig") << endl;
cout << "Yield of background is " << nBG.getVal() << ".  From sWeights it is " << sDataX->GetYieldFromSWeight("nBG") << endl << endl;

RooDataSet * dataw_sig = new RooDataSet(total_data->GetName(),total_data->GetTitle(),total_data, *total_data->get(),0,"nSig_sw");
RooDataSet * dataw_bg  = new RooDataSet(total_data->GetName(),total_data->GetTitle(),total_data, *total_data->get(),0,"nBG_sw");

cout << "Making splots of the signal and background" << endl;
RooPlot* frame_sig_PT = PT.frame();
frame_sig_PT->SetTitle("sPlot for the signal momentum distribution");
dataw_sig->plotOn(frame_sig_PT, RooFit::DataError(RooAbsData::SumW2));

RooPlot* frame_bg_PT = PT.frame(20);
frame_bg_PT->SetTitle("sPlot for the background momentum distribtuion");
dataw_bg->plotOn(frame_bg_PT, RooFit::DataError(RooAbsData::SumW2));
```

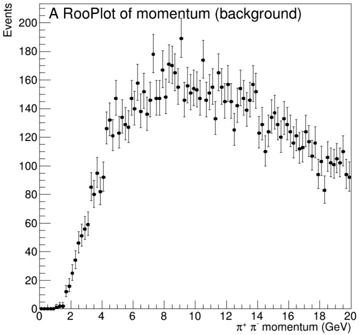
sPlot for P (signal)



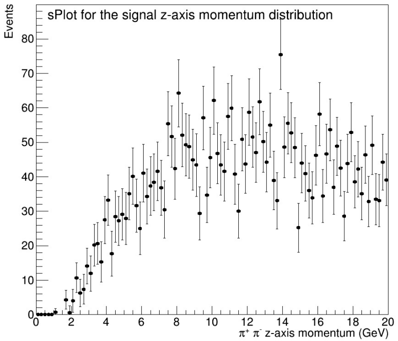


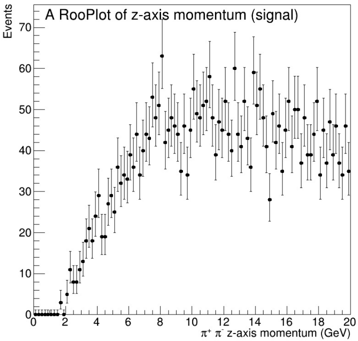
sPlot for P (background)



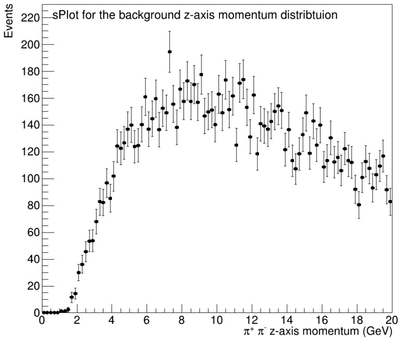


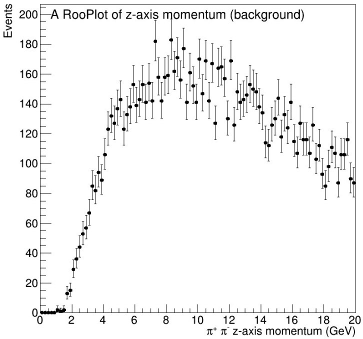
sPlot for PZ (signal)



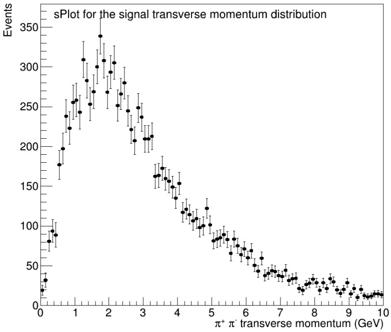


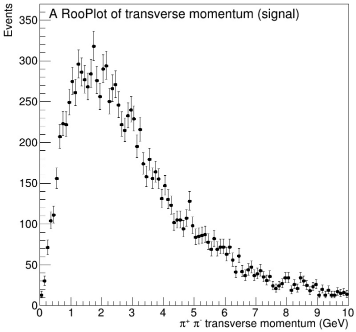
sPlot for PZ (background)



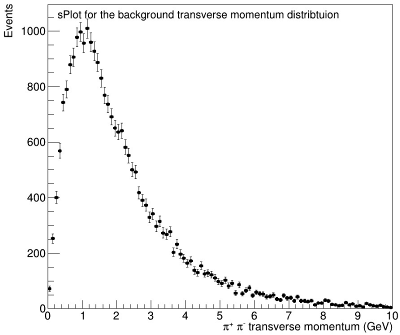


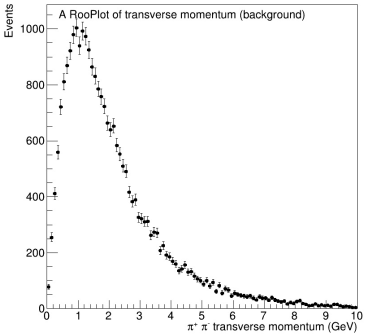
sPlot for PT (signal)





sPlot for PT (background)





Summary

- sPlot is a useful statistical tool that allows to reconstruct distributions of control variable (for signal and background respectively) using information obtained from discriminating variable distribution
- We managed to get quite interesting results - the reconstructed distributions match the real ones
- We will not stop here - we are going to conduct further research and try to make use of sPlot to split mass distribution into signal and background
- Personally, during this Project, I learned how to use Rapid Sim MC generator, use ROOT and RooFit to fit data and present them and finally - I gained some knowledge in field of statistics

Thank you for your attention!

Backup

References:

1. https://arxiv.org/abs/physics/0402083
2. http://arogozhnikov.github.io/2015/10/07/splot.html