LHC Data Analysis Tutorial

Preparation of the working area

- ssh -XY <your-username>@summer.ncg.ingrid.pt
- mkdir datatutorial; cd datatutorial
- mkdir plots
- cp /lstore/calo/aluisa/TutorialDataAnalysis/dimuon* .
- module load root

Presentation of the dataset and the macro

- The dataset is located in /lstore/calo/aluisa/TutorialDataAnalysis/Skim4.root (there is no need to copy it in your home, the macro automatically look for it in that location)
- You can check what is inside the dataset by opening it:

  `root -l <filename.root>`

- A `TTree` is present, containing the 4-momenta of the two muons and of the dimuon system. The 4-momentum of a particle contains the information about its relativistic energy and classical momentum vector: (E,px,py,pz)
- Each 4-momentum is saved using the `TLorentzVector` class that, in addition to allow access to the 4-momentum components, has many functions to easily compute several quantities out of it, for example the invariant mass: dimuon_p4.M() in our exercise we will need the invariant mass function, and the transverse component of the momentum [dimuon_p4.Pt()], but if you are interested you can find the list of available functions here: https://root.cern.ch/doc/master/classTLorentzVector.html
- The macro is composed of two files: the main one [`dimuons.C`] and the header [`dimuon.h`]. At the beginning of the main file there is the main function [void `dimuons()`] from which you can decide which other functions are called by (un)commenting them.
- The interesting functions are:
  - [`GetSpectrum()`] which reads the dataset and create the graph with the dimuon invariant mass spectrum, on which the particles' peaks are visible
  - [`Cut()`] (this is placed in the header file) where you can apply some selections on the events you are considering (like on the transverse momentum of the particles)
  - [`SelectPeak()`] which reads the dataset and creates a histogram of the dimuon mass spectrum in a specific mass range that you can customise (the default is set to select the region of the J/psi peak)
  - [`FitPeak()`] which is the old way to fit the peak and you don't need to use it
  - [`FitPeakRoofit()`] which reads the histogram produced by [SelectPeak()], defines the functions to describe the signal and the background (the default are a Gaussian and an exponential function, respectively), and performs the fit
- The documentation for the classes used in the macro can be found in https://root.cern.ch/doc/master/ (find your class in the 'class list' on the left menu)

- You can compile and run the macro by running in the terminal:

```
root -l -b -q dimuons.C++
```

- The results are saved in the plots folder as png files. You can open them by running: `display <filename.png>`
- During the tutorial I will show how to use the default configuration to fit the J/psi peak

Exercise

- Change the [`a._sdig`] value in the main function to match the day of the month of your birthday (easy way to randomise this value). This will select a different sample in the dataset for each of you.
- Change the peak to fit: choose whether to fit the psi' or the Z peak and modify the mass range in [`SelectPeak()`] in order to show that peak. To have a successful fit try to mimic the histogram we produced for the Jpsi while choosing the range: the peak in the middle and a nice portion of background should be present on each side of it.
- If you see that the default functions poorly describe the peak, try changing them. You can use a sum of two Gaussians, for example, or the CrystalBall function (it is present in the macro, just uncomment it), or maybe the sum of Gaussian + CrystalBall. When you sum more functions to describe one peak, make sure they use the same parameter for the mean (you don't want a bimodal function). To know more about the Crystal Ball function look on Wikipedia.
- Try applying some selection cuts in the [Cut()] function, for example on the pT of each muon and check if/how this changes the number of signal events and the shape of the peak. The default minimum pT value in the macro is 10 GeV/c (by default is commented, uncomment to activate), after testing it try to decrease or increase it. What is the selection efficiency for this cut, on your signal?
- Identify in the fit results the measured mass value and the number of events, with their statistical uncertainty.
- Once you have the resulting plot of the fit for your chosen fit, upload it on the slack channel.
- Test different functions to describe your signal and your background and verify how the results on the mass and number of signal events depend on the chosen function. This variation is a systematic effect of your measurement and is an additional source of error, not included in the statistical error of the fit. Estimate this uncertainty and compute the total error summing in quadrature the statistical and the systematic one.
- Once you have the measurement of the mass and number of signal events, with their errors, report them in the slack channel.
- Bonus exercise 1. Fit the upsilon peaks. Pay attention that they are too close to fit them individually.
- Bonus exercise 2. Perform a differential measurement of the number of signal events for your favorite peak: produce yield plots as a function of pT and eta of the dimuon system. To do this you need to use the whole statistics: change [`a._sdig`] to any negative value to read the whole dataset.