General

Welcome to the HASCO 2017 ROOT tutorial! We have a set of four exercises available, which we recommend you tackle in order. For exercises 1 and 2 you will have to use some of the commands shown in yesterday's lecture and demos, and solutions will be unveiled as we go. The second two are in case you have some ROOT experience and get through 1 and 2 much faster than expected. If you get stuck, ask each other, the internet and us!

Setup

Open a terminal, change to home directory, make your own subdirectory there

cd
mkdir <yourname>
cd <yourname>

download things from https://indico.cern.ch/event/607726/contributions/2475151/ and untar the relevant files

mkdir Exercise1
cd Exercise1
cp ~/Downloads/Ex1.tar ./
tar -xvf Ex1.tar

Now you can get going!

Exercise 1: Search for H -> gamma gamma

- **Introduction and setup**
  - We want to perform a search for H -> gamma gamma using the invariant diphoton mass.
  - The H -> gamma gamma channel has a tiny branching ratio (10^-3) but is very pure and has a great mass resolution due to the distinct signature of two photons with good energy resolution.
  - Untar `Ex1.tar`, you will find three files `PseudoData_Histogram_100fb.root`, `Signal_1fb.root` and `Background_1fb.root`
  - Advice: it might be a good idea to copy your sequence of successful commands into a macro as you go so you can re-run quickly in case root crashes
- **Plot the data**
  - `PseudoData_Histogram_100fb.root` is the data we have measured corresponding to 100 inverse fb
  - Inside the data file is a TH1D histogram called `signal`, which shows the invariant diphoton mass, plot it.
- **Plot the background simulation**
  - `Signal_1fb.root` and `Background_1fb.root` are the signal and background simulations corresponding to 1 inverse fb.
  - Inside the simulation files a TTree is stored called `tree`, which contains two variables `invariantMass and eventWeight`, the invariant diphoton mass and the event weight, respectively.
  - Create histograms and fill them with `invariantMass` weighted by the `eventWeight`.
  - Scale the simulation to the correct integrated luminosity of the data and compare the data to the background-only hypothesis. (Hint: event-by-event loop, SetBranchAddress, etc)
- **Background-only hypothesis test**
  - Perform a fit to the background in order to get a stable background model.
  - Compare the data with the background model by plotting the **difference** in a sub-plot below the main plot.

- **Signal strength extraction**:
  - Perform a signal + background fit. For stable fit behaviour, set the initial fit values of the background part to those obtained from the previous fit, and the signal ones to values that seem sensible.
  - From the signal component of this fit, determine the number of signal events, and compare to the number we expect from the signal simulation. The ratio extracted : expected is called the signal strength.
- **Finalize the plot**
  - Add the signal simulation to the background histogram.
  - Plot the signal component of the signal + background fit in the sub-plot
  - Make the plot look *nice* (but don't let perfect be the enemy of good!)

Exercise 2: Introduction to kinematics

- **Introduction**
  - In the previous exercise, we had some invariant mass spectra, for signal and background. In this exercise, we see how to create such distributions.
  - We want to examine the jet kinematics of a new physics signal: a so-called Z' - a new force mediator particle.
  - This force can couple quarks to dark matter, and also therefore quarks to other quarks
  - We are interested here in the case q q -> Z' -> q q, with a Z' of mass 450 GeV - what does this look like?
- **Setup**
  - Untar `Ex2.tar`, you will find the files `signal_dijet.root`, `myMacro.C` and `dijet_mass_example.png`.
- **Examine the data**
  - Look at the .root file in a TBrowser. This is an easy way to see what a root file contains and to have simple plots of the data created for you.
  - What do you expect the jet multiplicity and pT spectra to look like? What do they look like? Can you think why?
- **Plot the data from the tree**
  - Plotting more specific information is useful. Use the TTree->Draw() functionality to plot the leading jet pt
  - Does this make sense given your thoughts above?
  - Try and plot the lead jet energy for jets with: |eta| < 0.5 and |eta| > 1.5. Can you explain what you see?
- **Derive and plot more complicated kinematic variables**
  - What we are really interested in is the invariant mass of the two leading jets.
  - This is a bit cumbersome to plot from the tree, it's better to do this event-by-event.
  - Run `myMacro.C` by typing `root -bl myMacro.C` - it is a small root macro to loop over all events and make a leading jet pT histogram, which it saves to lead_jet_pt.png. This should look the same as the one from the previous step!
  - Following the example of `jet_pt`, change macro to also read `jet_eta`, `jet_phi` and `jet_E`
  - **once you reach this point, let the helpers know**
  - Use a TLorentzVector - https://root.cern.ch/doc/v608/classTLorentzVector.html - to calculate the dijet invariant mass for each event. Hint: SetPtEtaPhiE is a useful function.
  - Make a new histogram and save it. Does this look like you'd expect? Can you think of some reasons why it looks the way it does?

Exercises 3 and 4 – introductions to pyroot and TMVA

- These have step-by-step instructions to start with, then become open for you to explore. See the Exercise3 and Exercise4 .pdf, .html and .ipynb files (the html and ipynb may be better for copying across to the terminal, pdfs have been known to introduce annoying line breaks)