Summer CERN stage Logbook

Nicolò Foppiani, Tommaso Pajero

Università degli studi di Pisa

Scuola Normale Superiore

13 agosto 2015

Indice

1 Week 01

1.1 20/07/2015

Access to Cern

The first day we did a lot of bureaucracy. It's not necessary to register the car, our badge permits us to pass. During the afternoon we have attended to a conference during which our tutors have presented the projects we will work on.

1.2 21/07/2015

First use of Castor and Marlin

Kinematics

The first process we analyze is the $e^+e^- \to t\bar{t}$ with energy $\sqrt{s} = 365$ Gev, where one top decays as $t \to bW \quad W \to l\nu$, and the other top in three jets. This process with two leptons are more rare, while, on the other and, the process with 6 jets are really difficult to study.

By analyzing the kinematics of the process we have found that the lepton has to cut in energy: the configuration with the minimum energy is when the W has the maximum energy (emitted in the top direction) and the lepton is emitted back respect to the W fly direction; the configuration with the maximum energy is when the W has the maximum energy (emitted in the top direction) and the lepton is emitted front respect to the W fly direction.

With the top energy of 182.5 Gev we obtain a minimum energy of 13.45 Gev and a maximum energy of 120.16 Gev.

Anyway there will be a lot of background to this process (also with high energy) due to the lepton produced by gamma pair production or pions decays.

1.3 22/07/2015

PyROOT and Git

Installing PyRoot and Git, and created a Git repository. You can find some useful files in the utilities folder.

1.4 23/07/2015

Tree Root

Thanks to Maurizio we understood a lot about Root trees. They are like tables: the rows represent the events, the columns are the branches and contain the information of the events.

Trees are contained in a ROOT file, to see the content we could use ROOT as follow: root -l myfile.root _file0-¿ls()

To see a tree we can use the command: mytree-¿Scan() and it permits to see it as a table.

To loop on a tree using C++ we have to create the class associated to the tree by using: mytree-¿MakeClass(myclass) This command creates the file myclass.h that contains the definition of the class with the basic methods. The class contains some variables and the pointers to the branches. The file myclass.C is a ROOT macro, with the function loop, which loops on the events of the tree: you can add your code in this function and make the loop by executing this macro.

In python it's more simple, because one can open the ROOT file and load the tree by using

myfile = TFile(myfile.root,READ) mytree = file.Get(mytree)

and then, to loop on the events of the tree

for event in mytree: my code

Grow space in Lxplus

Thanks to Maurizio we have also increased our free space on Lxplus and adopted the CMSSW_7_4_7 environment to use Root macro on Lxplus. See the file configurazioneLXPlus.txt in utilities to find some useful commands.

1.5   24/07/2015

Using Pyroot scripts

We have started writing python scripts. At the end of the script, all the canvas opened are automatically closed, so, to save the histograms and plots the best way is to save them into a Root file. Here is an example.

file_to_save1 = TFile(./file_to_save1.root,CREATE) Part_ID.Write()

where PartID is a TCanvas variable.

We can also launch Python script using

python -i myscript.py

which don't close the prompt at the end of the macro.

Glob

A useful command in Python is Glob, which gives an array of the files in one folder; we could then loop on this array.

It's also useful to keep the bash library on Python by importin os library.

SLCIO Marlin ntuples

We understood something more about the ntuple produced by Marlin.

mcpdg contains the ID of the simulated particles.

mcgst contains the status of the particles. There are 4 status possible: 0, 1, 2, 102 we have understood that the particles in final state (which are revealed by the detectors) have status 1, while particles which decay have status 2. bottom quarks have status 0 and we will check later which particles have status 102.

mcmox,y,z are the three components of the momentum of the particle mcvtx,y,z are the vertex coordinates where the particle is produced

We understood that there are lot of muons and electrons (also reconstructed) which are not produced in the W decay. They are produced, for instance, by the electrons which radiate. This effect creates a background to the leptons we want to study. In addition the leptons we want to study can radiate and it's difficult to reconstruct these leptons. Maybe, during the study of the montecarlo events, we could analyze the vertex in which the leptons are produced.

We need to ask Patrick and Patrizia if they prefer to import all ROOT library (like import ROOT as RT) or only the functions we need in that macro.

We need also to ask them if they prefer to write a logbook with us or by themselves.

We produced a few plots of the electrons and muons energy in both reconstructed and simulated events. We need to talk about them with Patrick and Patrizia. We need also to understand how the events were smiulated (for instance if each event is only one collision or a collision between two bunches).

2 Week 02

2.1 27/07/2015

Stage and copy file from CERN servers

Patrick have copied all the .slcio files in his directory and processed them with Marlin using two bash scripts. We have copied the ntuples from eos (see the glossary) to our folder, and now we can loop among them.

He used two files: jobCopy.sh and jobMarlin.sh to process the files.

jobCopy first stage the files from the tape (Castor) to the disk, and then copies the files into EOS folder. jobMarlin loops above all the files writing the correct steering.xml file and create all the ntuple.

More about the slcio ntuples

We have found out something else about our ntuples:

We discovered that all the top quark have status 3, and the bottoms have sometimes status 2 and sometimes status 3.

By using the parent information (mcpa0 e mcpa1, for daughters mcda0,1,2,3,4) we managed to select only the leptons which come from the W decay. Then, with a pyroot macro we filled the 2-D histograms of the number of leptons versus the energy and the angle.

2.2 28/07/2015

Lepton vs energy-angle histogram

We managed to loop above all the .root files. We use 256 bins (16 for the angle and 16 for the energy), that are quite similar to the square root of the entries (they are about 40k, the 40% of 100k events).

We have also used the reduced energy defined as

$$x = \frac{2E}{m_t} \sqrt{\frac{1-\beta}{1+\beta}}$$

$$\beta = \sqrt{1 - 4m_t^2/s}$$

We plotted the histo of the number of lepton in function of the $cos(\theta)$ and of the reduced energy. We expected a curve distribution in both the variables, and, to do this, we have to plot only the positive leptons (or the negative); otherwise there can't be any asymmetry in the angular distribution.

The result can be seen in figure 2.1, 2.2, 2.3

Unfortunately the angular distribution is flat: this implies that Pythia simulation don't consider the top polarization. So we cannot use these simulation to perform our analysis, and we need to proceed with different data.

Writing Root Tree using PyROOT

We have also learn how to write a Tree Root using PyROOT. We need first to define a TTree variable, then define the branches, and then fill the branches with a 0-D array (this is important), by looping on the events. Further information in the glossary folder.

2.3 29/07/2015

Copy and Marlin new simulation files

We decided to use other simulation files made with Wizard (instead that Pythia), we could find them at this site clicca qui These simulations are made at 365 GeV and would consider the polarization of the top quarks (we hope so).

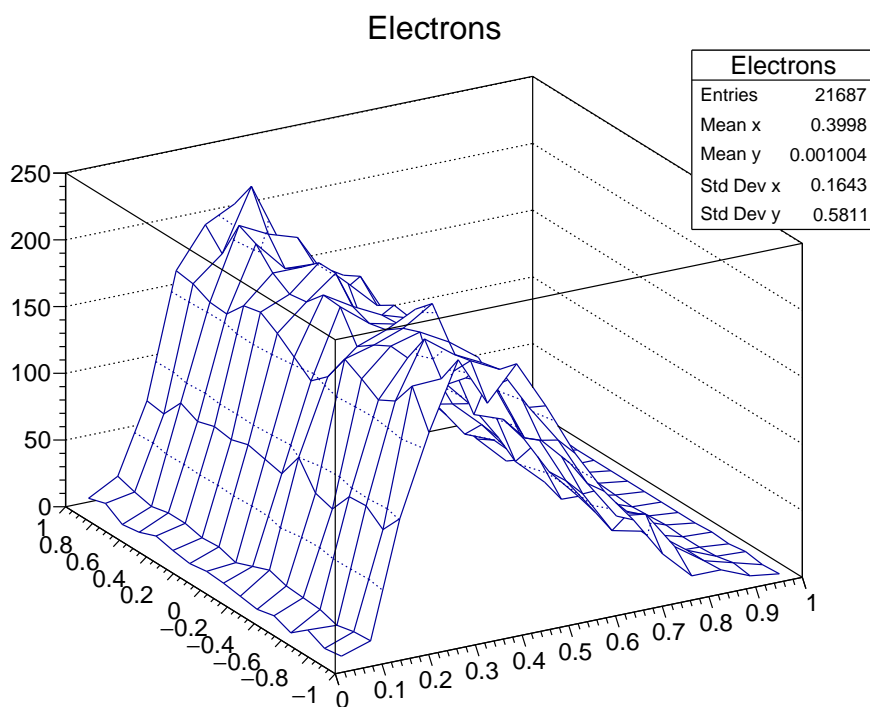By using some scripts bash we have staged all the files, copied and Marlin them.

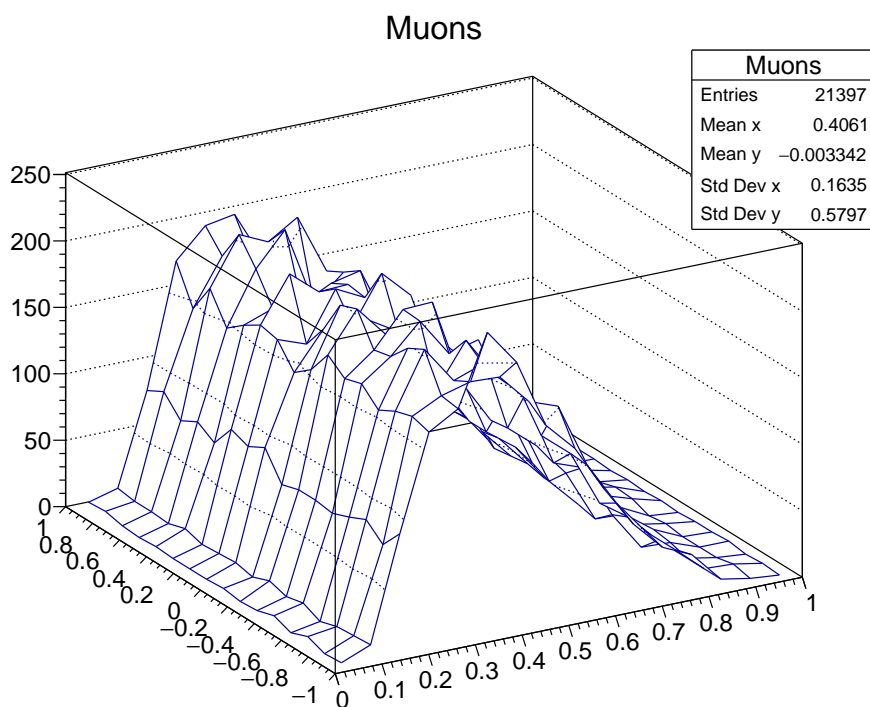Figura 2.1: Energy-angle distribution of the electrons

Figura 2.2: Energy-angle distribution of the negative muons

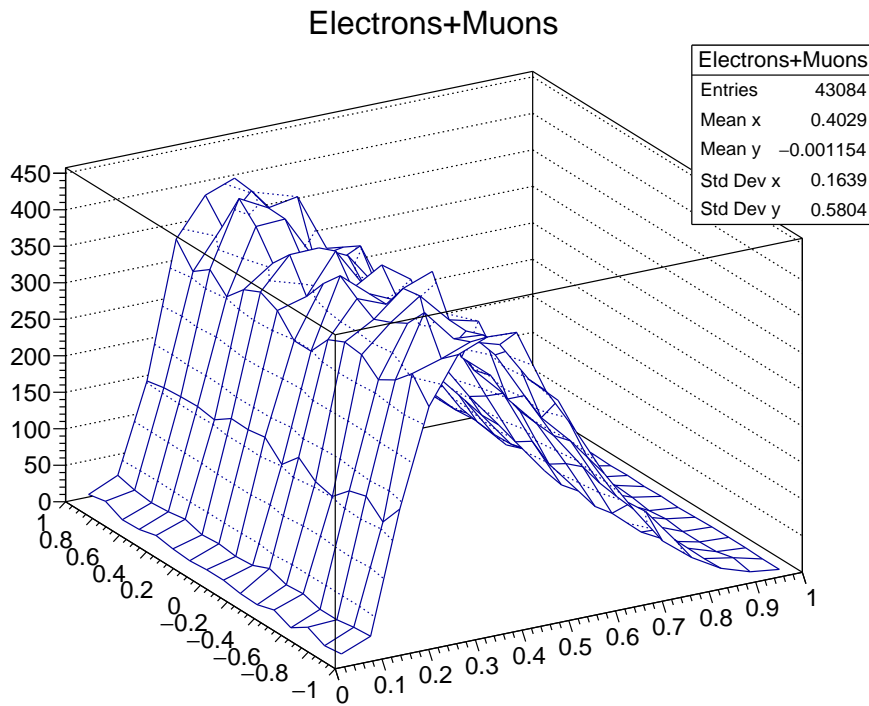## 2.4   30/07/2015

Whizard ntuples

Figura 2.3: Energy-angle distribution of the electrons and negative muons

We have started to analyze the new ntuples obtained with Whizard simulation. The tree branches have the same name. Though we found out that all the events contains the following particles: 0,1 : the two incoming electrons 2,3 : the two incoming electrons after radiating soft photons 4,5 : the two soft photons 6,7,8,9,10,11 : the final state of the event with b an $\bar{b}$ quark, a couple quark and anti-quark, the electron (positron) we are looking for with index 10 and the correspondent neutrino.

So we have simply plotted the distribution of the electron with index 10 and we found out that the angular distribution has the asymmetry we are looking for.
Now we are going to run our script on lxplus looping all the data.

In addition we discover that we have not always the resonant state in the simulation, and sometimes we have a top without an antitop (or viceversa); so we don't know very well how the simulation was made. We don't care to this fact very much for the moment.

2.5   31/07/2015

Whizard ntuples
The result of the plot with all date is visible in figure 2.4.

We have then compared this distribution with the one calculated analytically (figure 2.5), and, by making the ratio of the two histograms we obtained something really flat (figure 2.6), so the two distributions agree. There are some spike in the ratio of the two histograms near the boundary of the allowed energy. This is due to the fact that probably the top mass and the W mass were chosen approximately (we found out that the top mass were chosen to be 174 GeV), and the maximum energy of the electrons depends a lot on the top mass.

If we better analyze the ratio by zooming on the distribution in the flat region, we can see that there are many fluctuations that are due to the little quantity of data.

In the end we can say that our work can start, because our data take in account the polarization effect.
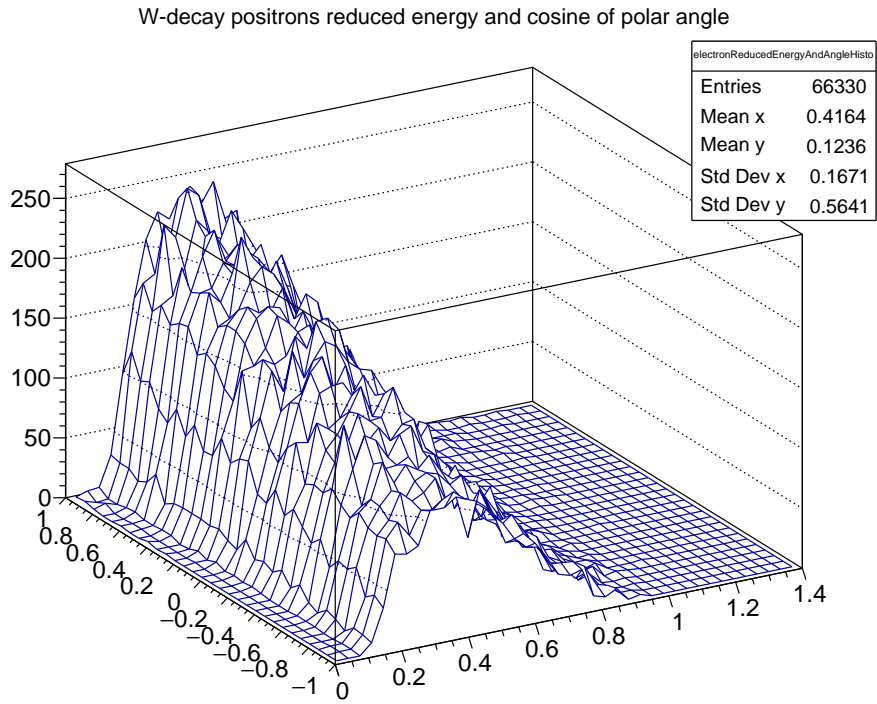
W-decay positrons reduced energy and cosine of polar angle

electronReducedEnergyAndAngleHisto	
Entries	66330
Mean x	0.4164
Mean y	0.1236
Std Dev x	0.1671
Std Dev y	0.5641

Figura 2.4: Energy-angle distribution of the electrons simulated by Whizard

smcross

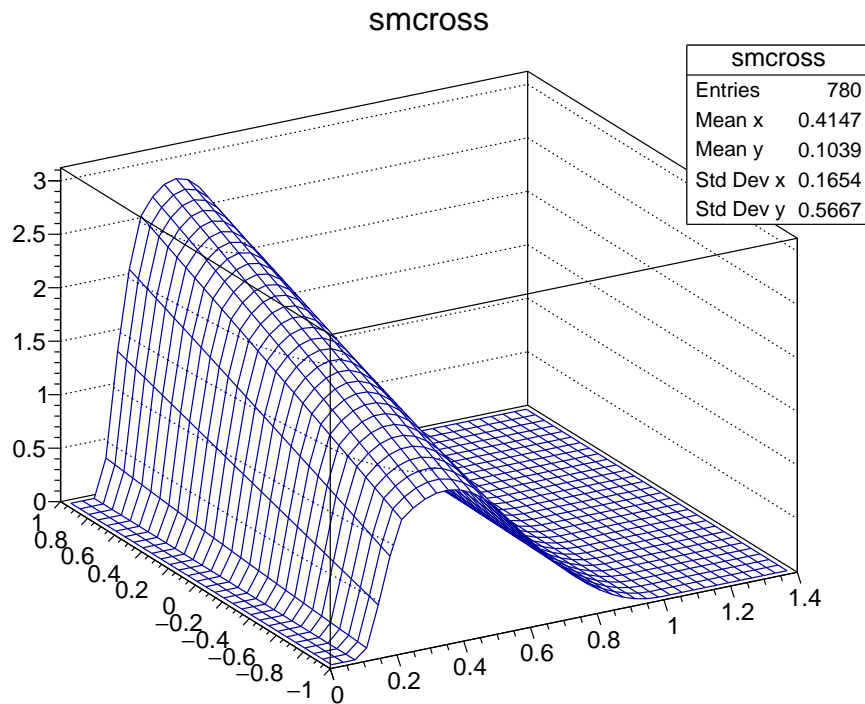| smcross	
Entries	780
Mean x	0.4147
Mean y	0.1039
Std Dev x	0.1654
Std Dev y	0.5667

Figura 2.5: Energy-angle distribution of the electrons calculated analytically

2.6 Other leptons simulations

We have just started to stage, copy and Marlin also the files containing one muon or tau.
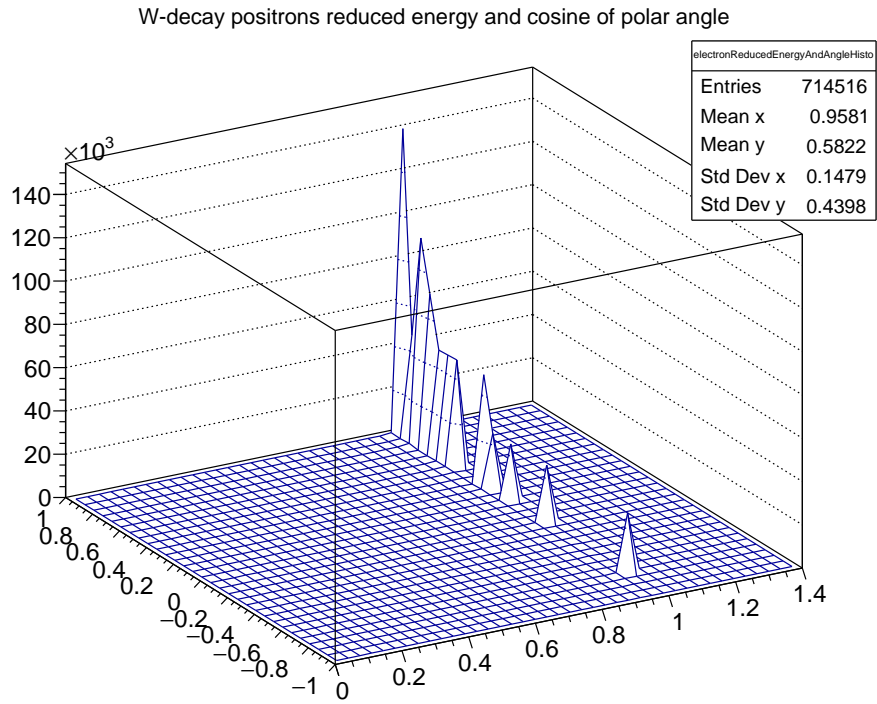
W-decay positrons reduced energy and cosine of polar angle

Figura 2.6: Ratio of the distribution simulated and analytic

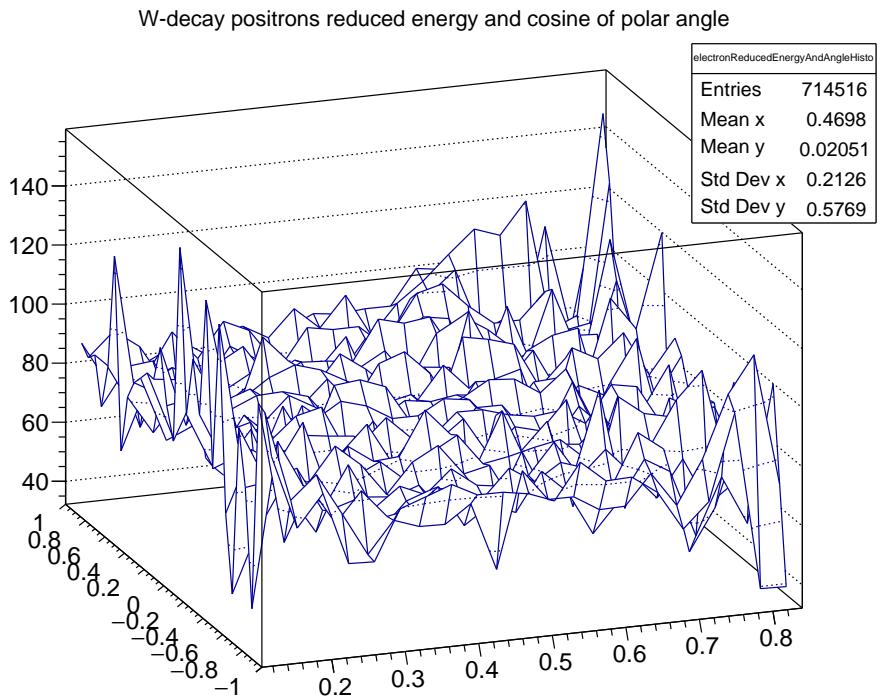W-decay positrons reduced energy and cosine of polar angle

Figura 2.7: Zoom of the ratio of the distribution simulated and analytic

3 Week 03

3.1 03/08/2015

Other leptons Whizard simulations

We have copied and Marlin all the other simulations files with one leptons (muons or electrons) in the final state.

We would like to analyze the energy-angle distribution to understand if the boundary effect are due to some diagrams which interferes (only in the electrons case) or if they are due to something else.

Likelihood fit

After a lesson about the likelihood and how to fit an histogram by finding the minimum of the likelihood (thanks to Patrick), we started fitting our energy-angle distribution for the electrons with a model $S_0 + af_1$, where S_0 is the standard model distribution and f_1 is a BSM correction.

We have started from two histograms of S_0 and f_1 with 200 bins in each dimension and reduced energy between 0.114246 and 1, and $cos\theta$ between -1 and 1 (obviously).

We have then normalized the distribution so we can use them as probability density function. So we have reproduced our distribution with these settings and use the likelihood formula:

$$L = -\sum_{events} log(S_0 + af_1) = -\sum_{bins} N_{bin} log(S_0 + af_1)$$

$$L' = -\sum N \frac{f_1}{S_0 + af_1} and L'' = \sum N \frac{f_1^2}{(S_0 + af_1)^2}$$

where N is the number of entries in the selected bin of the montecarlo histogram, and S_0 and f_1 are the numbers of the entries in the histograms of the probability density function.

Then we could use the tangent method to find iteratively the minimum value (that we expect to be 0).

$$a_0 = a - \frac{L'(a)}{L''(a)}$$

and after all we can estimate the error with

$$\sigma = \sqrt{\frac{1}{L''(a_0)}}$$

We have found some errors while calculating the Likelihood and its derivatives cause to the fact that S_0 and f_1 became very near to 0 in some points. We don't now how to solve yet.

3.2 04/08/2015

TChain

We have just discovered a new method to process a lot of Root files containing tree without using glob. We simply need to define a TChain object:

mychain = TChain(MyTreeName) where MyTreeName is the name of the trees inside the root files.

Then we can add all the root files to the chain by using: mychain.Add(*.root)

and then we can loop on all the events dealing mychain as a tree.

Likelihood fit part 2

We found the problem in the code: strangely, if we look to the S_0 and f_1, we find that the bins (0,*) and (*,0) contains exactly 0 entries. So during the calculation of the derivatives we find several problems in dividing by zero. We solve the problem by simply don't consider these bins. The result is pretty strange because, starting with an initial value of a=0.1 we obtain a result of $a = 0.2 \pm 0.01$ with

a first derivative very close to zero. This result is only quite reasonable because we expect something close to zero, with 0 into the errorbar.

After a little time we found out that the first bins (*,0) and (0,*) and the last bins (*,201) and (201,*) are the overflow and the underflow of the histograms, so we need to use the method GetBinContent on the bin between 1 and 200 in both directions. The result hasn't changed and we tried to plot the marginal distributions. But this result is not satisfactory.

3.3 05/08/2015

Physics lesson at Gigi's home

We have attended a physics lesson at Gigi's home about how to do physics at e+e- colliders.

3.4 06/08/2015

Leptons tree and effective invariant mass

We have created 6 trees (one for each possible lepton in the final state) with only the variables we need. In particular there are the energy (and reduced energy), the three component of the momentum, the polar angle, and the invariant mass of the two electrons after radiating photons. This invariant mass is the effective energy at which the collisions happen, and the mean is 363.7 GeV.

3.5 07/08/2015

Corrections to the likelihood

Together with Patrick, we realized that the likelihood we defined on Monday 03/08/2015 was not correct. In fact, the normalization factor that turns the cross sections into the probabilities depends on the parameter a, and we have also to consider the probability that our MonteCarlo generator produced N events, if the mean value of events is μ. Let S_{tot}, f_{tot} and N_{tot} be the sum of the contents of the bins in our range of study (we may want to make some cuts in the values of $\cos\theta$ and the reduced energy x) for the Standard Model cross section, the correction-to-SM cross sections, and the number of events generated by Whizard. Let the luminosity (integrated over the experiment time) be K. Then, the probability of having N events produced in our range of study is:

$$\frac{\mu^N e^{-\mu}}{N!} \quad \rightarrow \quad \frac{e^{\frac{-(N-\mu)^2}{2\mu}}}{\sqrt{2\pi\mu}}$$

where $N = KS_{tot}$, $\mu = K(S_{tot} + af_{tot})$ and we have used the fact that $\mu \gg 1$[1]. Therefore, the likelihood is:

$$L = \left(\prod_{bins}\left(\frac{S_{bin} + af_{bin}}{S_{tot} + af_{tot}}\right)^{N_{bin}}\right)\frac{e^{\frac{-(N-\mu)^2}{2\mu}}}{\sqrt{2\pi\mu}}$$

We would like to maximize the likelihood, by minimizing the following quantity:

$$\mathcal{L} = -\log L - \frac{1}{2}\log(2\pi N_{tot}) =$$

$$= -\sum_{bin}\log(S_{bin} + af_{bin}) + N_{tot}\log(S_{tot} + af_{tot}) - \frac{1}{2}\log\mu + \frac{(\mu - N)^2}{2\mu} =$$

$$= -\sum_{bin}\log(S_{bin} + af_{bin}) + N_{tot}\log(S_{tot} + af_{tot}) + \frac{1}{2}\log\left(\frac{S_{tot} + af_{tot}}{S_{tot}}\right) + N_{tot}\frac{(af_{tot})^2}{2S_{tot}(S_{tot} + af_{tot})}$$

$$(3.1)$$

---

[1]Moreover, we have made the subtle (not true!) assumption that Whizard generates events following our same theoretical renormalization scheme.

We performed the one dimentional fits for both f_A^Z and f_B^Z corrections to SM, using the simulated distribution for the electrons. To avoid the deviations from our expected model, we considered only data with reduced energy lesser than 0.9; taking in account the acceptance of our experimental apparatus, we excluded from our fit the data with $|\cos\theta| > 0.9$. The results are equal to zero within 2σ for f_A^Z and 3σ for f_B^Z and they improved by using the 364 GeV theoretical distribution instead of the 365 one (and also if we used as reduced energy cut 0.8 instead of 0.9). For example, the value obtained for the f_A^Z coefficient, with the 364 GeV distribution, is $a = 0.62 \pm 0.39\%$. The likelihood as function of a, and the projections of the fitted and the simulated distributions are plotted in figures 3.1, 3.2 and 3.3
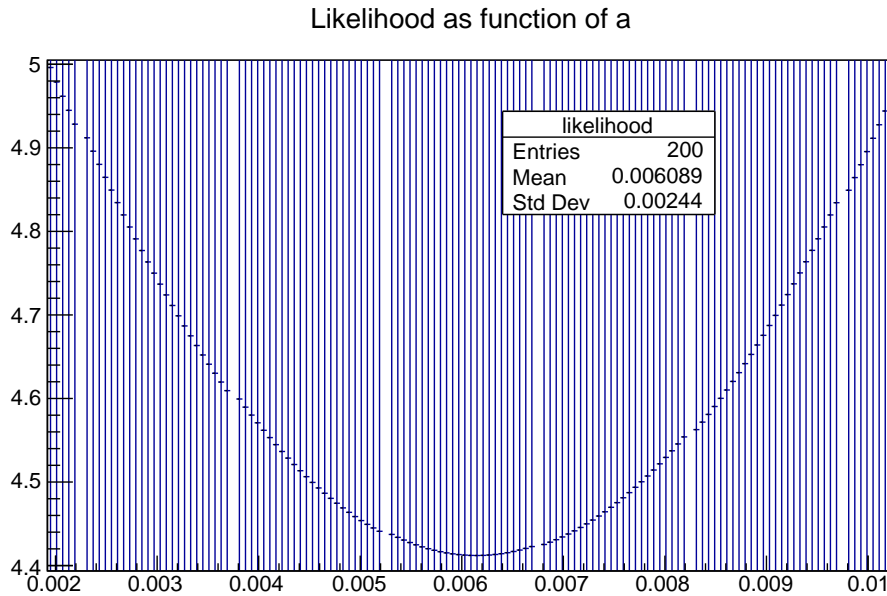

Figura 3.1: Likelihood as funcion of a for the one-dimentional fit of the electron distribution with f_A^Z BSM correction.

Fit 2D

We have tried to perform the fit with two corrections together. The generalization is simple: in 1D case we have that

$$L'(a) = (a - a_0)L''(a_0)$$

that could be rewritten as

$$a_0 = a - L'(a)/L''(a_0) \simeq a - L'(a)/L''(a)$$

and could be iterated.

in multidimensional case we have that

$$\frac{\partial L}{\partial a_i} = \frac{\partial^2 L}{\partial a_i \partial a_k}(a_k - a0_k)$$

which is a linear system. Then we could solve the system and iterate the process.

The inverse of the $\frac{\partial^2 L}{\partial a_i \partial a_k}$ matrix in the minimum point is the covariance matrix.

We have performed this fit with all the electron/positron data of the montecarlo. The result is:

$$a = \begin{pmatrix} 0.005 \\ -0.08 \end{pmatrix}$$

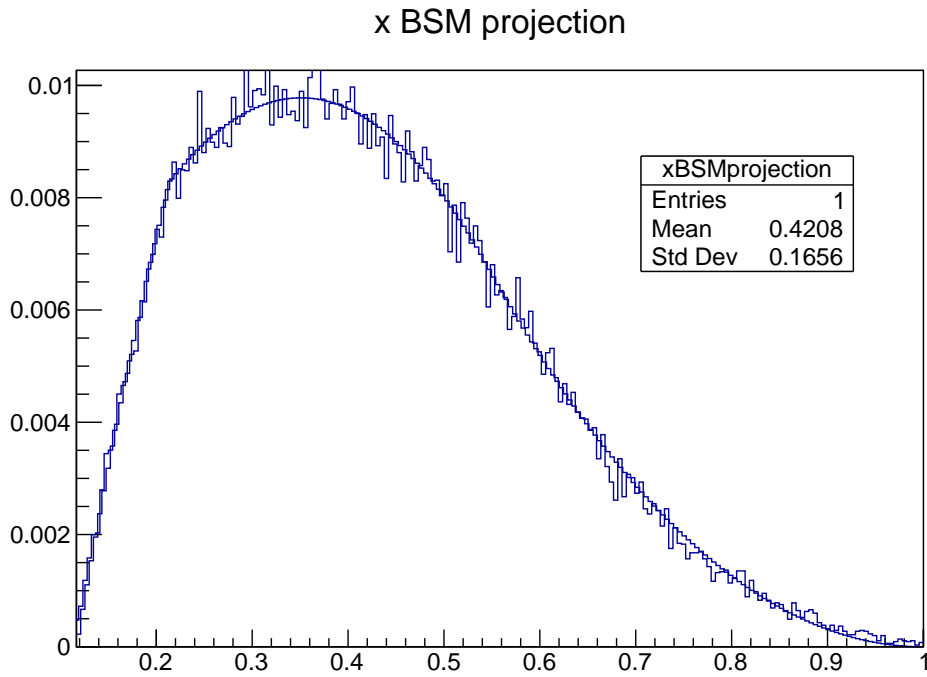$$cova = \begin{pmatrix} 1.5e-05 & 2.6e-05 \\ 2.6e-05 & 1.16e-03 \end{pmatrix}$$

Figura 3.2: Electrons distribution both simulated and fitted considering the f_A^Z BSM correction, projected on the reduced energy axis.
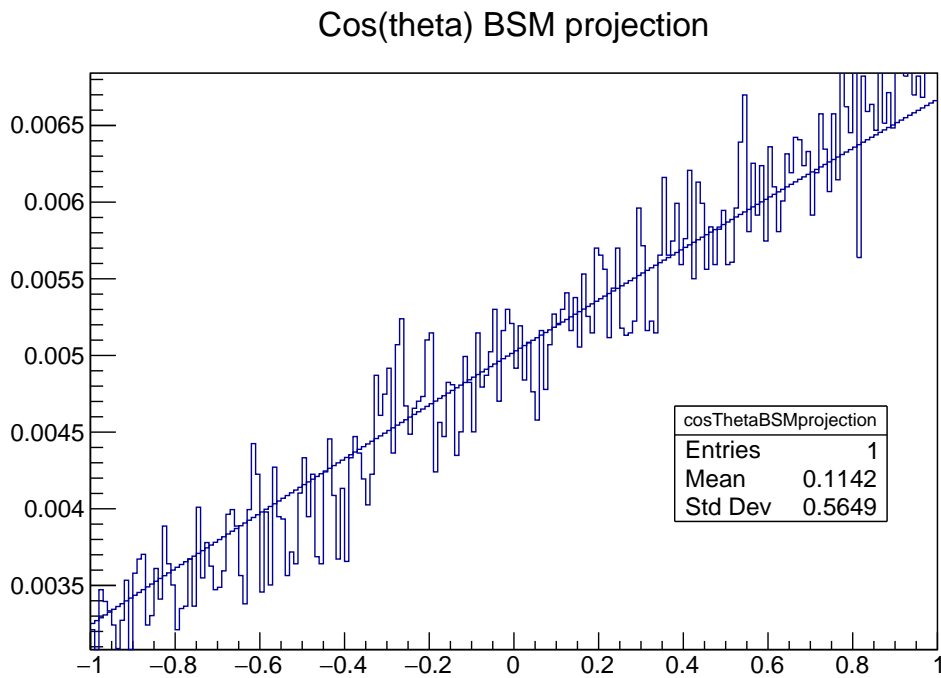

Figura 3.3: Electrons distribution both simulated and fitted considering the f_A^Z BSM correction, projected on the $\cos\theta$ axis.

Plot the 2D ellipse corresponding to a given confidence level

For a likelihood-fit with 2 or more dimensions, the area of the ellipse with axis equal to one sigma for each dimension does not correspond to the 68 % confidence level. The right way is to find the value of a parameter k for which the area under $k \cdot \sigma$ is the confidence level given (for instance 68 %).

For 2D gaussian one can use the formula $k^2 = -2ln(1 - Pr)$ where Pr is the given probability.

In addition here is a table with some k-factor for multidimensional gaussian: "Click here"

So, we have to plot the ellipse

$$\left(\frac{x}{\sigma_x}\right)^2 + \left(\frac{y}{\sigma_y}\right)^2 = k^2$$

To plot the ellipse in ROOT we used the TEllipse class, which draw an ellipse on the canvas. It's important to set the option myellipse.SetFillStyle(0) which draw a transparent ellipse. To add the axis we draw two lines by using a TGraph, which automatically add the axis to the canvas. A TGraph is a set of two (or more) arrays of coordinates of points and can be constructed in this way: mygraph=TGraph(numberOfPoints,x[i],y[i]). The final result is plotted in figure 3.4, in which you can see the three ellipse corresponding to the 68% CL, 95% CL, 99% CL. As we can see our result agrees with 0 with 99% CL.
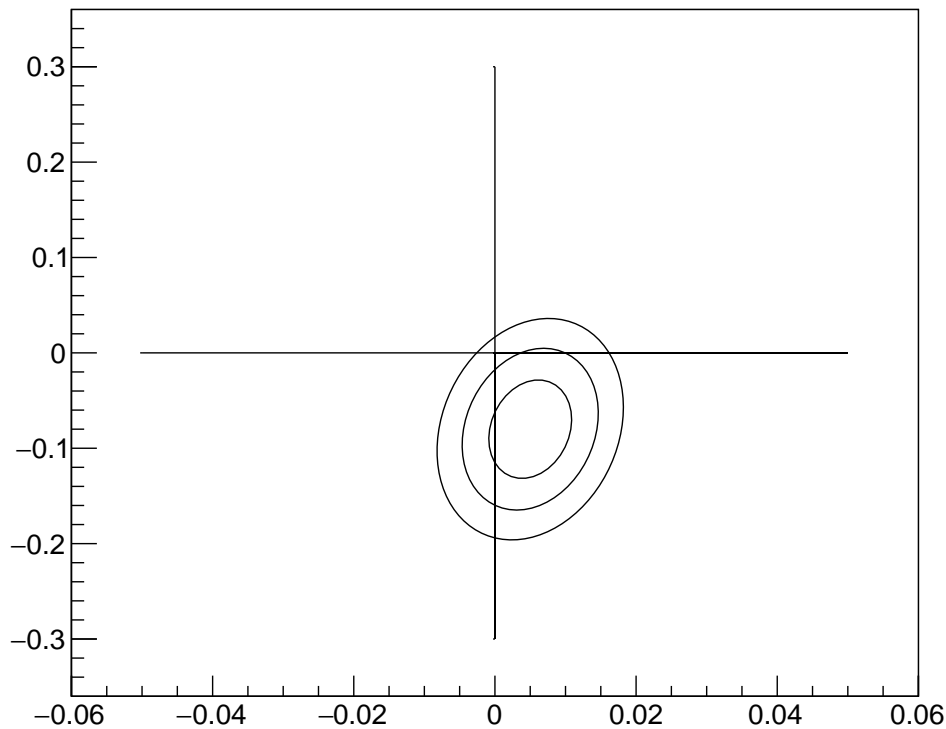

Figura 3.4: Electrons distribution both simulated and fitted considering the f_A^Z BSM correction, projected on the $\cos\theta$ axis.

This distance from 0 is a theoretical problem due to the different theoretical re-normalization scheme used for calculating Feynman Diagrams in the two cases: Whizard simulation and our theoretical distribution.

Fit in ROOT

To fit an histogram we can use the method called Fit which performs the fit with some options (like maximizing the likelihood) but we need the analytic formula (we cannot fit an histogram with another histogram).

There are also other tools to perform fit like Roofit.

NB: To do during the next weeks

It could be interesting to reproduce the likelihood fit and the ellipse plot adding the muons. In addition we could newly draw the ellipse using the variable g_l and g_r. They are defined in Patrick's slides.

3.6   09/08/2015

Particle distance in phase space

Now we need to find the reconstructed particle that corresponds to our lepton. We think that the best way is looping on all the reconstructed charged particle finding the particle that minimize this distance in phase space:

$$\sqrt{(E - E_{mc})^2 + (\vec{p} - \vec{p_{mc}})^2}$$

where E_{mc} and p_{mc} are the energy and momentum of our lepton.

We have just written a script which fills an histogram with the type, the distance and the difference in energy between the reconstructed particle and the montecarlo lepton. By trying this script with electron only 99 events I found that the reconstructed particle is mostly an electron, sometimes a photon (5 case) and in a couple of case is a negative pion. I think that the photon are that ones radiated by the electrons, and probably they must be added into the electron energy. A pion could be confused as an electron because it could happen that a charged pion interacting in the electromagnetic calorimeter became a neutral pion that decays immediately in two photons.

Screen and running processes in background

If you want to launch a command from a LxPlus shell and you don't want the command to be stopped when you close the connection you can use two ways:

- run in background: if you run the command by adding & at the end it will run in background. When this will finish you will get a string [process number] Done.

- using Screen: if you use screen (see the glossary) you can in parallel a lot of processes. If you disconnect you won't have these screens available unless you reconnect to the same computer (the number of the computer is written near lxplus in the shell). So you need to reconnect by using ssh -XY myname@lxplus1234.cern.ch

4   Week 04

4.1   10/08/2015

Searching among the reconstructed particles

We have just started to write some macros and to plot some histograms to understand how to find the electron coming from the primary vertex. The result are described in the next subsection.

4.2   11/08/2015

Meeting with Patrick and Patrizia

We have had a meeting in which we have talked about some things:

- Acceptance and efficiency: We need to produce some plots to understand what is the efficiency of our algorithm that finds the right electron. In addition we could understand which is the acceptance of the detector (in particular we expect that the acceptance is flat in the energy and have a plateau in cosTheta and two tails near 1 and -1.

- Bremsstrahlung correction: We need to correct the bremsstrahlung of the electrons by adding the energy of the photons radiated by the electron. One way is to select the photons in a cone around the electron. A better way is to select the photons radiated parallel to the electron track: we could select the vertex and the end point of the electron and we could add all the photons with angle theta and phi compatible with the electron.

- Sensitivity in measuring the energy and the Pt of the electron: we need to produce some plots to take in account the smearing produced by the uncertainty in measuring energy and transverse momentum.

- Start writing code with classes and function

New organization of the code

The code is now written using classes and function. We have defined a class Particle and a class Jet which contain the information we want in our object and some useful methods (like a correction which apply a bremsstrahlung correction to the electrons by adding the photons in a cone).

Finding the correct electron

We are searching for the best algorithm to find the correct a electron between the reconstructed particles. There are some ways to find this electron starting from the request that this electron is isolated. To see if the method is good or not we always plot the difference in energy and in angle between the montecarlo electron and the reconstructed electron.

- Pt rel: we loop on all the electrons, and, for each electron we select the jet respect to which the electron has the minimum Pt. The correct electron is, then, the electron with the highest minimum Pt rel.

- Pt rel to the closest jet: we loop on all the electrons and we search for the the closest jet, and the we calculate the Pt respect to that jet. The correct electron is, then, the electron with the highest Pt rel to the closest Jet.

- Energy in a cone: we loop on all the electron and we find that one with the minimum energy inside a cone with fixed angle (typically 10 degrees).

- Particle in a cone: we loop on all the electron and we find that one with the maximum distance from the closest particle (and we can add some cut, for instance we can require the closest charged particle or the closest particle different from a photon or an electron).

In addition we use the kinematic constrain (Energy lower then 13 GeV) to select only the electrons with an energy lower than 10 GeV (to take in account the experimental uncertainty. This have two effects: improve the efficiency and reduce the computation time.

Results:???

4.3   13/08/15

We changed our approach, restarting our study considering the muons sample. This makes our work simpler, because muons have less background than electrons and usually loose less energy before their detection. Besides, there are far fewer muons than electrons in every event.

Matching algorithm calibration We would like to find an algorithm to decide wether a reconstructed (RC) muon matches the W-decay Monte Carlo (W-MC) one. We decided to use, as discriminant variable, the angle between the relative momenta. To set the threshold value, we considered the events with three or more RC muons[2], where is more probable that there is a muon that matches the W-MC one. Using only these events, we filled the histogram of the angle of the RC muons from the W-MC muon. For each event, we used red colour for the nearest RC muon, and green colour for the others. The results are plotted in figures 4.2 and 4.1.
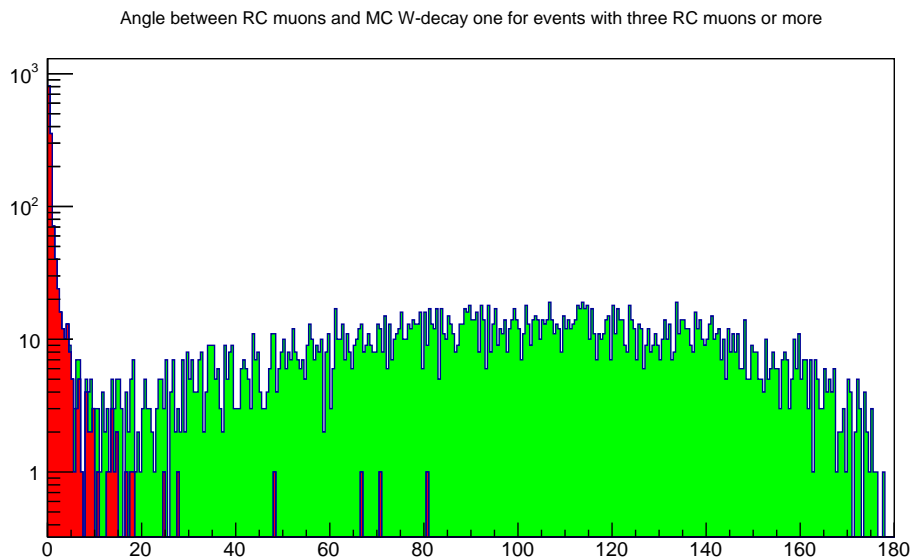
Angle between RC muons and MC W-decay one for events with three RC muons or more

Figura 4.1: YLogscale is used for a better visualisation.

Looking at figure 4.2, we decided to use 5.5^o as the maximum angle between two muons to say that they can match.

To sum up, we will say that a RC muon matches the W-MC muon if it is the closest RC muon to the W-MC one and the angle between the two is smaller than 5.5^o.

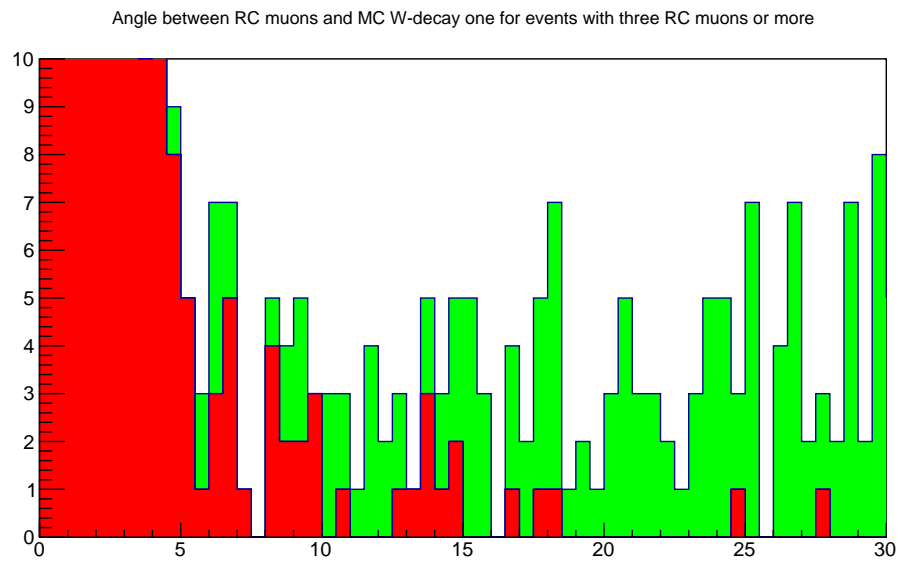[2]Which are about 2% of the events, whereas there are two RC muons in 20% of the events and no RC muons in 1% of cases.

Figura 4.2: Zoom near the cut.