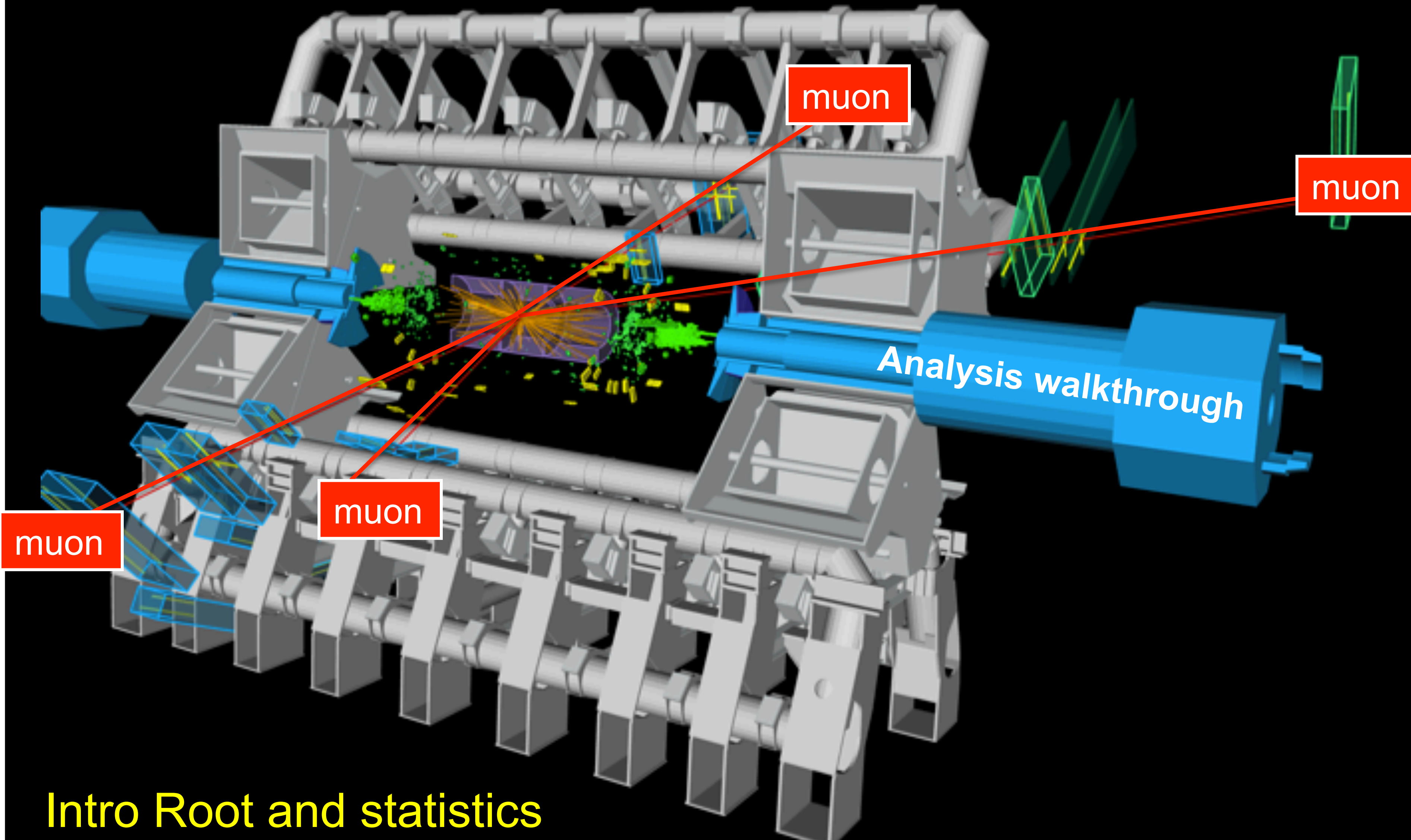
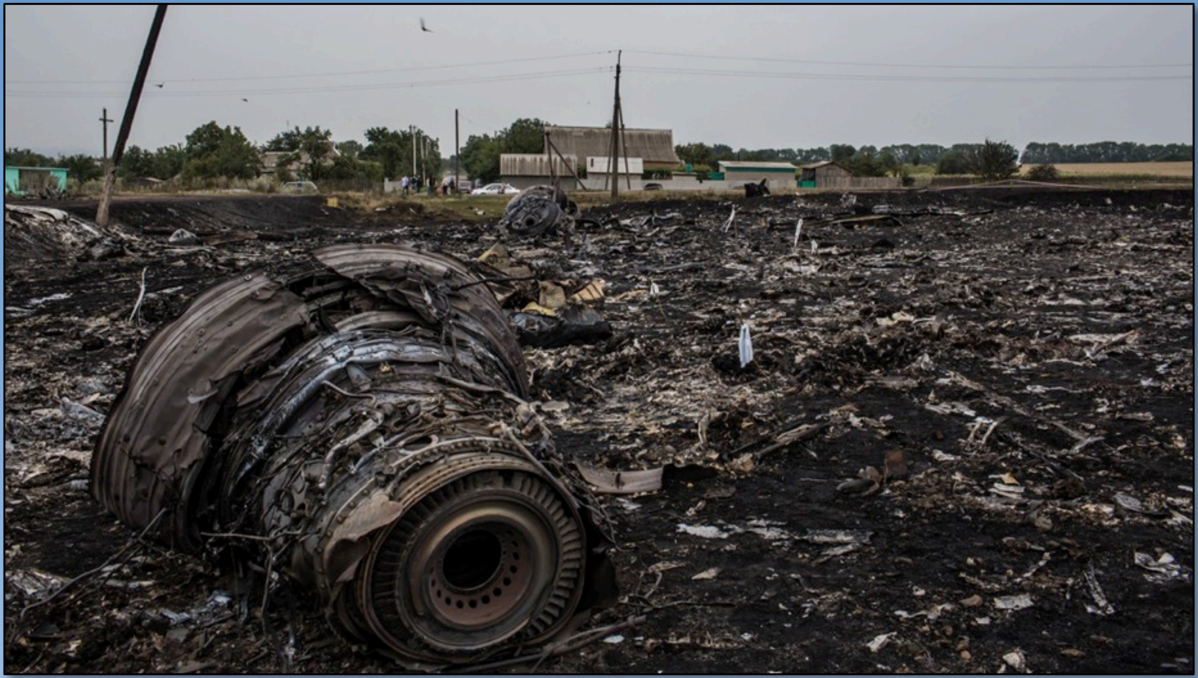
HASCO summerschool, July 2014
(Göttingen)



Intro Root and statistics

Ivo van Vulpen (UvA/Nikhef)



July 17th 2014
MH17 plane crash

298 people died
(193 Dutch)

HASCO school is more than just learning about the Standard Model

International collaboration and research is not about politics. On the contrary

This does NOT mean that I like this!

on the contrary!

Computing at HASCO school

Intro Root / statistics

Ivo van Vulpen
(Amsterdam)

Friday 25/7
- 11:00-12:30 Lecture
- 16:30-18:30 Hands-on

Data preparation

Riccardo Di Sipio
(Bologna)

Monday 28/7
- 11:30-12:30 Lecture
- 14:00-16:00 Hands-on

Computing at HASCO school

Intro Root / statistics

Ivo van Vulpen
(Amsterdam)

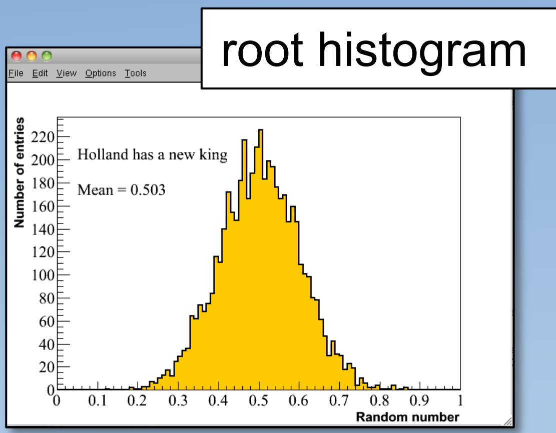
Lecturer at university of Amsterdam
- Introduction Python programming
- Particle physics (bachelor)
- Higgs (master/PhD)

Researcher at Nikhef
- ATLAS (Higgs physics)

Goals for today

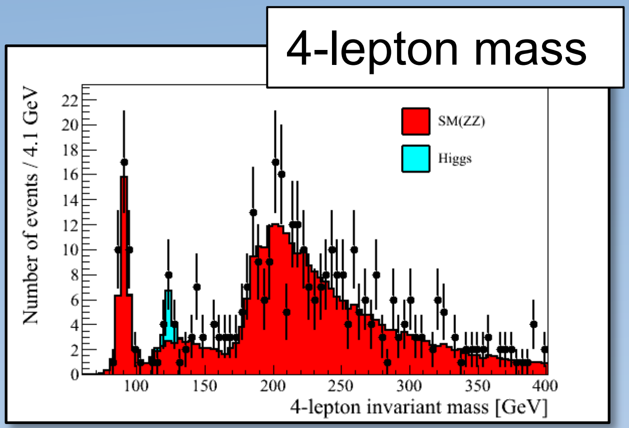
Note: background in computing for HASCO students is VERY different

[A] Root and data manipulation



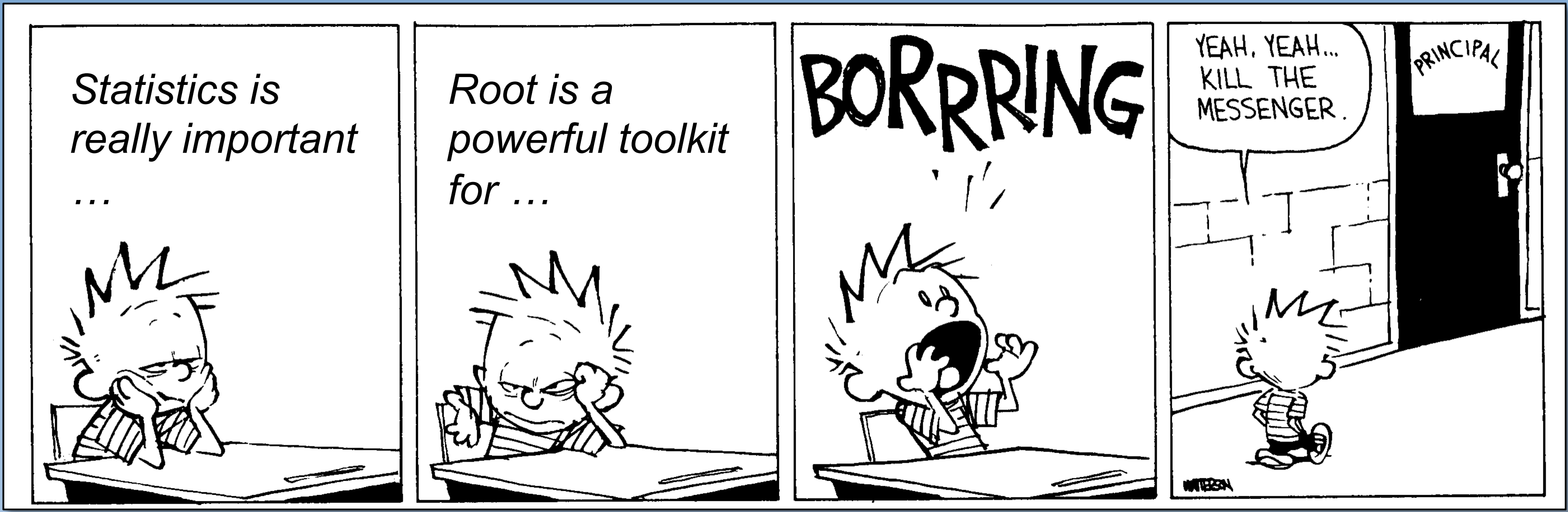
1) Manipulate histograms from a root file
2) Write and compile your own Root macro

[B] Statistics and data analysis



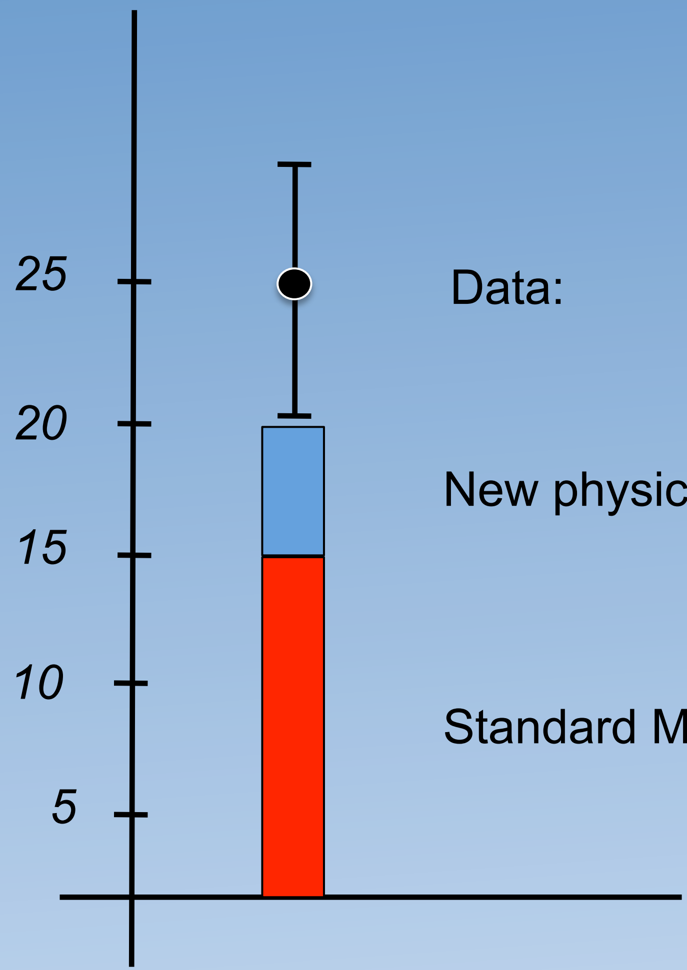
1) Do a likelihood fit yourself bkg. estimation in signal region
2) Definition of significance optimize a search window

A short lecture on Root and statistics



Note: will be 95% of your work. Master this and you can focus on physics

Example of ‘simple’ statistics



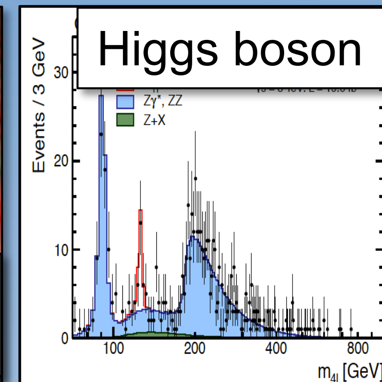
Data: $N_{data} = 25$

New physics: $N_{sig} = 5$

Standard Model: $N_{bgr} = 15$

What is the significance ?

Statistics is everywhere in science and industry



- Many mysteries, folklore, buzz-words, bluffing etc.
  - → **you** need to master it to summarize your analysis
  - → do **not** just ‘do what everybody else / your supervisor does‘

- RooFit, BAT, TMVA, BDT’s are excellent and very powerful tools
  Understand the basics → **then** ask RooFit to do complex stuff

“Do the basics yourself at least once”

Results from any scientific or business study:

$$\text{Result} = X \pm \Delta X$$

Particle mass, cross-section, excluded cross-section, numerical integral, probability of bankruptcy … or becoming a millionair

Make sure you know how you extracted this

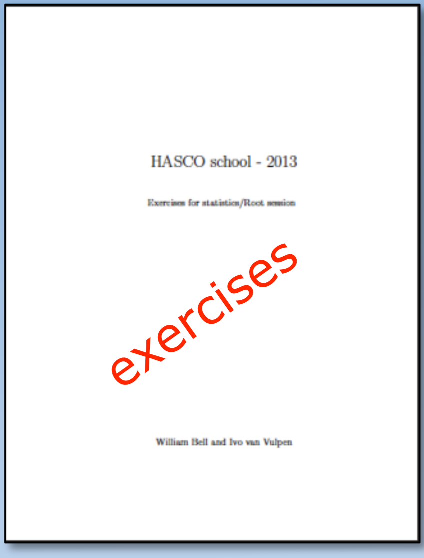
Root/Statistics sessions

Goal: 1) have you learn Root basics and manipulate histogram data
2) do a Likelihood fit and extract a number and it's error
3) understand what a significance is

Friday:

Lecture 1: Introduction to Root and statistics-exercises

Exercises: Hands-on computing exercises


HASCO school - 2013

Exercises for statistics/Root session

exercises

William Bell and Ivo van Vulpen

Note: - exam questions will cover Root/statistic concepts (i.e. no coding)
- if you do not know any C++ or are a Root-expert already let me know

Basic material for the Root examples:

1) Download tarball: **HascoRootStatisticsCode.tgz**

2) Unpack everything: **tar –vzxf HascoRootStatisticsCode.tgz**

Directory /RootExamples/

- **a)** Example0*.C (* = 0,1,2,3,4,5) — All *.C-files in this presentation
- **b)** Code for Ntuple production and reading
- **c)** rootlogon.C — some standard Root settings

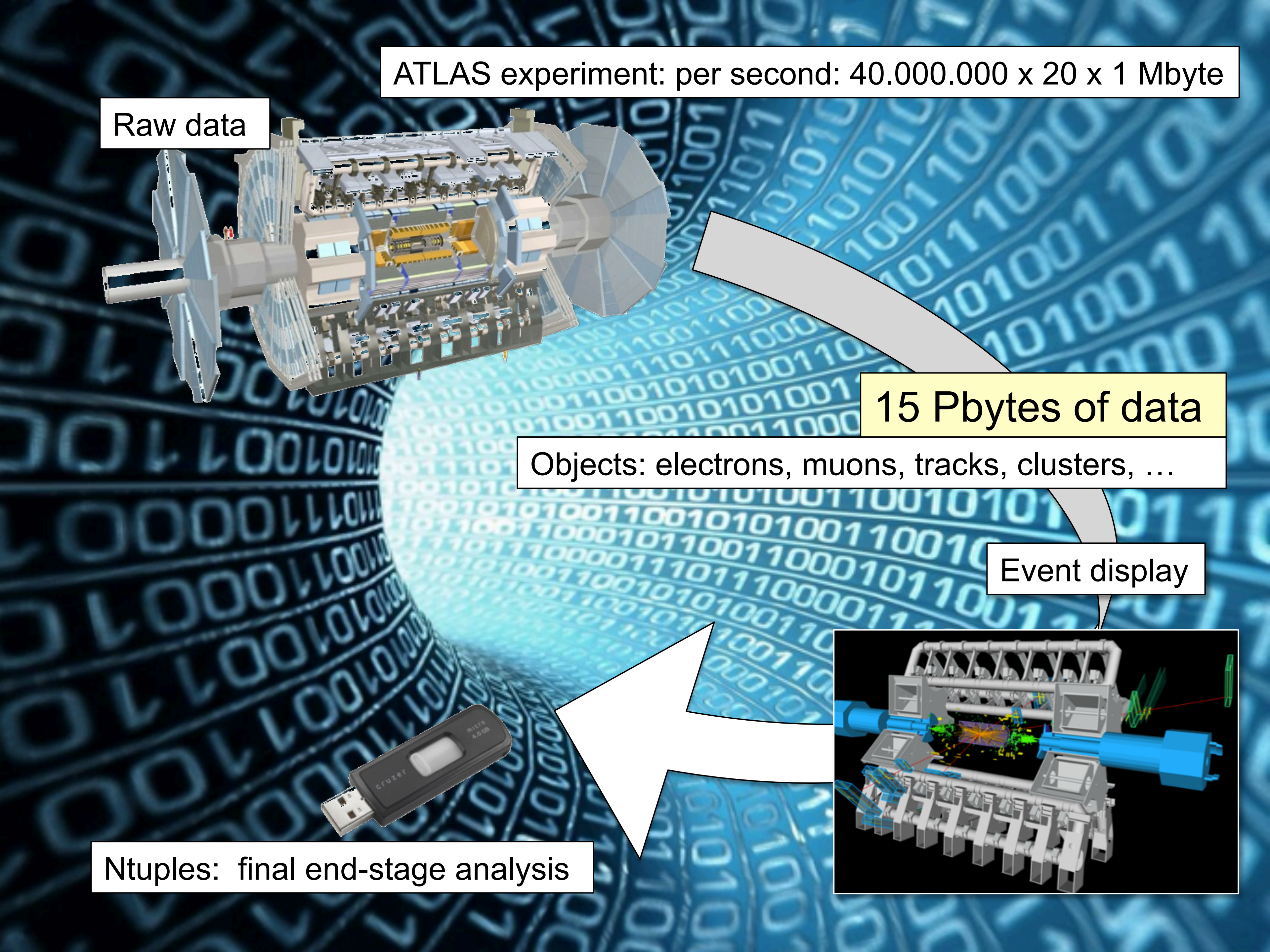
Directory /Exercises/

- **a)** Histograms_fake.root
  4 histograms of the 4-lepton invariant mass (H125, H200, ZZ, data)
- **b)** Hasco_skeleton.C
  skeleton code (different levels, as minimal as possible). Your code !
- **c)** rootlogon.C
  some standard Root settings

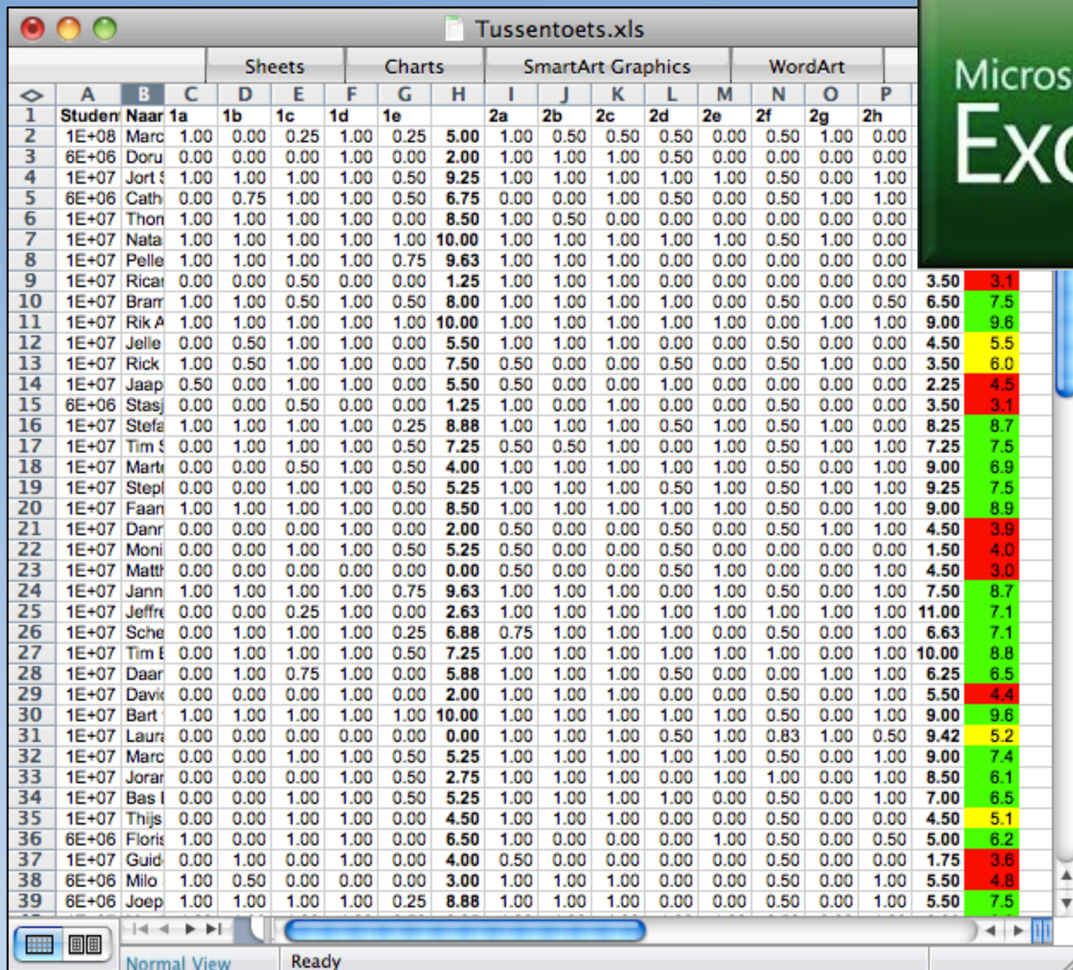
Root

Don't worry when you do not know C++ or Root already

… well ok, worry a bit, but not too much


ATLAS experiment: per second: 40.000.000 x 20 x 1 Mbyte
Raw data
15 Pbytes of data
Objects: electrons, muons, tracks, clusters, …
Event display
Ntuples: final end-stage analysis

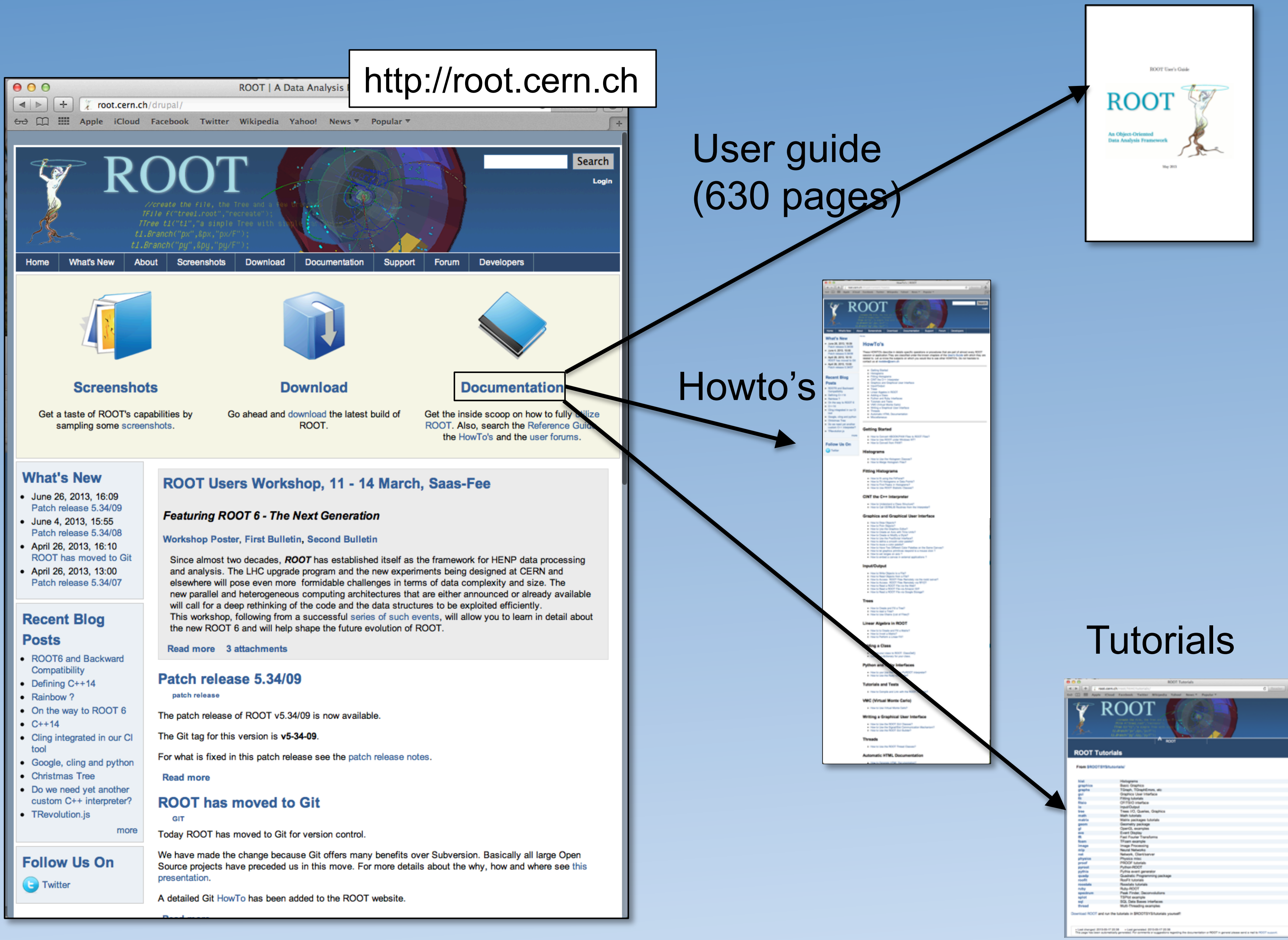
Excell or ascii files will not do



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P		
1	Studen	Naar	1a	1b	1c	1d	1e		2a	2b	2c	2d	2e	2f	2g	2h		
2	1E+08	Marc	1.00	0.00	0.25	1.00	0.25	5.00	1.00	0.50	0.50	0.50	0.00	0.50	1.00	0.00		
3	6E+06	Doru	0.00	0.00	0.00	1.00	0.00	2.00	1.00	1.00	1.00	0.50	0.00	0.00	0.00	0.00		
4	1E+07	Jort	1.00	1.00	1.00	1.00	0.50	9.25	1.00	1.00	1.00	1.00	1.00	0.50	0.00	1.00		
5	6E+06	Cath	0.00	0.75	1.00	1.00	0.50	6.75	0.00	0.00	1.00	0.50	0.00	0.50	1.00	1.00		
6	1E+07	Thon	1.00	1.00	1.00	1.00	0.00	8.50	1.00	0.50	0.00	0.00	0.00	0.00	0.00	0.00		
7	1E+07	Nata	1.00	1.00	1.00	1.00	1.00	10.00	1.00	1.00	1.00	1.00	1.00	0.50	1.00	0.00		
8	1E+07	Pelle	1.00	1.00	1.00	1.00	0.75	9.63	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00		
9	1E+07	Ricar	0.00	0.00	0.50	0.00	0.00	1.25	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00	3.50	3.1
10	1E+07	Bram	1.00	1.00	0.50	1.00	0.50	8.00	1.00	1.00	1.00	1.00	0.00	0.50	0.00	0.50	6.50	7.5
11	1E+07	Rik A	1.00	1.00	1.00	1.00	1.00	10.00	1.00	1.00	1.00	1.00	1.00	0.00	1.00	1.00	9.00	9.6
12	1E+07	Jelle	0.00	0.50	1.00	1.00	0.00	5.50	1.00	1.00	1.00	0.00	0.00	0.50	0.00	0.00	4.50	5.5
13	1E+07	Rick	1.00	0.50	1.00	1.00	0.00	7.50	0.50	0.00	0.00	0.50	0.00	0.50	1.00	0.00	3.50	6.0
14	1E+07	Jaap	0.50	0.00	1.00	1.00	0.00	5.50	0.50	0.00	0.00	1.00	0.00	0.00	0.00	0.00	2.25	4.5
15	6E+06	Stasj	0.00	0.00	0.50	0.00	0.00	1.25	1.00	0.00	1.00	0.00	0.00	0.50	0.00	0.00	3.50	3.1
16	1E+07	Stefa	1.00	1.00	1.00	1.00	0.25	8.88	1.00	1.00	1.00	0.50	1.00	0.50	1.00	0.00	8.25	8.7
17	1E+07	Tim	0.00	1.00	1.00	1.00	0.50	7.25	0.50	0.50	1.00	0.00	1.00	0.50	1.00	1.00	7.25	7.5
18	1E+07	Marti	0.00	0.00	0.50	1.00	0.50	4.00	1.00	1.00	1.00	1.00	1.00	0.50	0.00	1.00	9.00	6.9
19	1E+07	Stepl	0.00	0.00	1.00	1.00	0.50	5.25	1.00	1.00	1.00	0.50	1.00	0.50	1.00	1.00	9.25	7.5
20	1E+07	Faan	1.00	1.00	1.00	1.00	0.00	8.50	1.00	1.00	1.00	1.00	1.00	0.50	0.00	1.00	9.00	8.9
21	1E+07	Danr	0.00	0.00	0.00	1.00	0.00	2.00	0.50	0.00	0.00	0.50	0.00	0.50	1.00	1.00	4.50	3.9
22	1E+07	Moni	0.00	0.00	1.00	1.00	0.50	5.25	0.50	0.00	0.00	0.50	0.00	0.00	0.00	0.00	1.50	4.0
23	1E+07	Matth	0.00	0.00	0.00	0.00	0.00	0.00	0.50	0.00	0.00	0.50	1.00	0.00	0.00	1.00	4.50	3.0
24	1E+07	Jann	1.00	1.00	1.00	1.00	0.75	9.63	1.00	1.00	1.00	0.00	1.00	0.50	0.00	1.00	7.50	8.7
25	1E+07	Jeffre	0.00	0.00	0.25	1.00	0.00	2.63	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	11.00	7.1
26	1E+07	Sche	0.00	1.00	1.00	1.00	0.25	6.88	0.75	1.00	1.00	1.00	0.00	0.50	0.00	1.00	6.63	7.1
27	1E+07	Tim E	0.00	1.00	1.00	1.00	0.50	7.25	1.00	1.00	1.00	1.00	1.00	1.00	0.00	1.00	10.00	8.8
28	1E+07	Daar	0.00	1.00	0.75	1.00	0.00	5.88	1.00	1.00	1.00	0.50	0.00	0.00	1.00	1.00	6.25	6.5
29	1E+07	Davi	0.00	0.00	0.00	1.00	0.00	2.00	1.00	1.00	1.00	0.00	0.00	0.50	0.00	1.00	5.50	4.4
30	1E+07	Bart	1.00	1.00	1.00	1.00	1.00	10.00	1.00	1.00	1.00	1.00	1.00	0.50	0.00	1.00	9.00	9.6
31	1E+07	Laura	0.00	0.00	0.00	0.00	0.00	0.00	1.00	1.00	1.00	0.50	1.00	0.83	1.00	0.50	9.42	5.2
32	1E+07	Marc	0.00	0.00	1.00	1.00	0.50	5.25	1.00	1.00	1.00	1.00	1.00	0.50	0.00	1.00	9.00	7.4
33	1E+07	Jorar	0.00	0.00	0.00	1.00	0.50	2.75	1.00	1.00	1.00	0.00	1.00	1.00	0.00	1.00	8.50	6.1
34	1E+07	Bas I	0.00	0.00	1.00	1.00	0.50	5.25	1.00	1.00	1.00	1.00	0.00	0.50	0.00	1.00	7.00	6.5
35	1E+07	Thijs	0.00	0.00	1.00	1.00	0.00	4.50	1.00	1.00	1.00	0.00	0.00	0.50	0.00	0.00	4.50	5.1
36	6E+06	Floris	1.00	0.00	1.00	1.00	0.00	6.50	1.00	0.00	0.00	0.00	1.00	0.50	0.00	0.50	5.00	6.2
37	1E+07	Guid	0.00	1.00	0.00	1.00	0.00	4.00	0.50	0.00	0.00	0.00	0.00	0.50	0.00	0.00	1.75	3.6
38	6E+06	Milo	1.00	0.50	0.00	0.00	0.00	3.00	1.00	1.00	1.00	0.00	0.00	0.50	0.00	1.00	5.50	4.8
39	6E+06	Joep	1.00	1.00	1.00	1.00	0.25	8.88	1.00	1.00	1.00	0.00	0.00	0.50	0.00	1.00	5.50	7.5

ROOT

An Object-Oriented
Data Analysis Framework

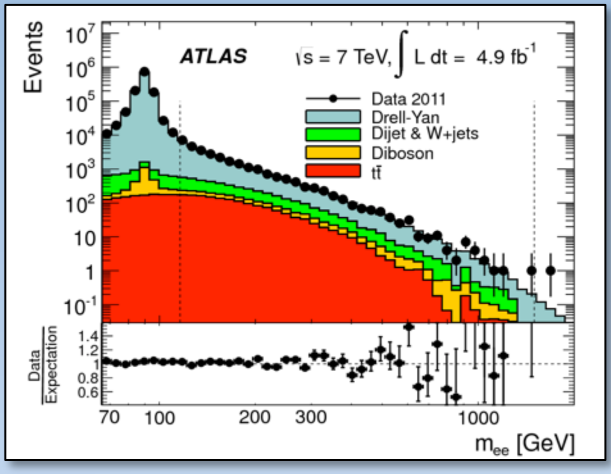

http://root.cern.ch
User guide
(630 pages)
Howto's
Tutorials
ROOT | A Data Analysis
root.cern.ch/drupal/
Apple iCloud Facebook Twitter Wikipedia Yahoo! News Popular
ROOT
Search
Login
//create the file, the Tree and a few branches
TFile f("tree1.root","recreate");
TTree t1("t1","a simple Tree with simple variables");
t1.Branch("px",&px,"px/F");
t1.Branch("py",&py,"py/F");
Home
What's New
About
Screenshots
Download
Documentation
Support
Forum
Developers
Screenshots
Get a taste of ROOT's capabilities by sampling some screenshots.
Download
Go ahead and download the latest build of ROOT.
Documentation
Get the inside scoop on how to fully utilize ROOT. Also, search the Reference Guide, the HowTo's and the user forums.
What's New
June 26, 2013, 16:09
Patch release 5.34/09
June 4, 2013, 15:55
Patch release 5.34/08
April 26, 2013, 16:10
ROOT has moved to Git
April 26, 2013, 13:00
Patch release 5.34/07
Recent Blog Posts
ROOT6 and Backward Compatibility
Defining C++14
Rainbow ?
On the way to ROOT 6
C++14
Cling integrated in our CI tool
Google, cling and python
Christmas Tree
Do we need yet another custom C++ interpreter?
TRevolution.js
more
Follow Us On
Twitter
ROOT Users Workshop, 11 - 14 March, Saas-Fee
Featuring ROOT 6 - The Next Generation
Workshop Poster, First Bulletin, Second Bulletin
Since almost two decades, ROOT has established itself as the framework for HENP data processing and analysis. The LHC upgrade program and the new experiments being designed at CERN and elsewhere will pose even more formidable challenges in terms of data complexity and size. The new parallel and heterogeneous computing architectures that are either announced or already available will call for a deep rethinking of the code and the data structures to be exploited efficiently.
This workshop, following from a successful series of such events, will allow you to learn in detail about the new ROOT 6 and will help shape the future evolution of ROOT.
Read more 3 attachments
Patch release 5.34/09
patch release
The patch release of ROOT v5.34/09 is now available.
The Git tag for this version is v5-34-09.
For what is fixed in this patch release see the patch release notes.
Read more
ROOT has moved to Git
GIT
Today ROOT has moved to Git for version control.
We have made the change because Git offers many benefits over Subversion. Basically all large Open Source projects have preceded us in this move. For more details about the why, how and where see this presentation.
A detailed Git HowTo has been added to the ROOT website.
ROOT
An Object-Oriented Data Analysis Framework
ROOT
HowTo's
ROOT
ROOT Tutorials

Root:

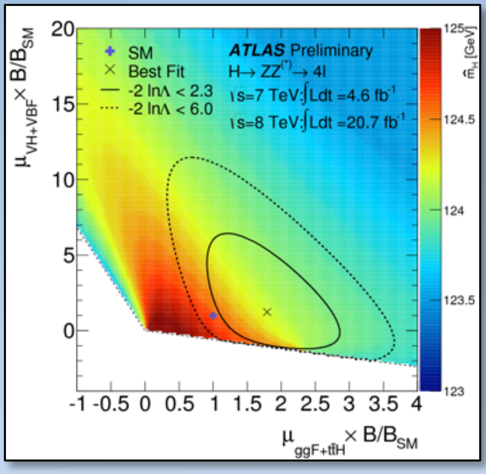
[1] HEP toolkit: C++ based analysis framework
[2] Data analysis: storing, manipulating
[3] Modeling: model building, fitting, hypothesis testing
[4] Visualisation: 1 to multi-dimensional

Root [4]: data visualisation

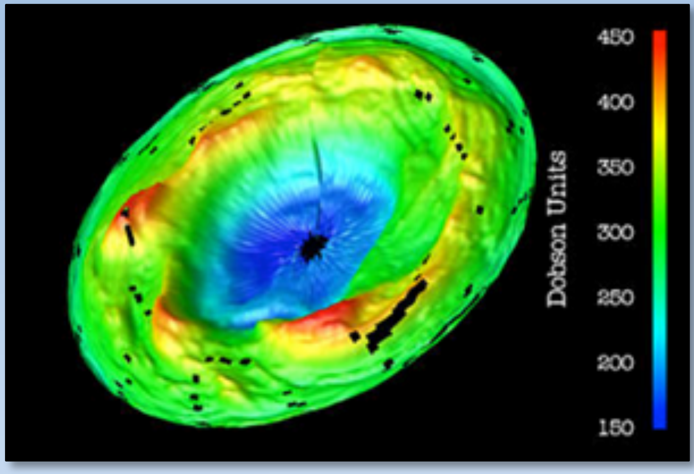
1-dimensional



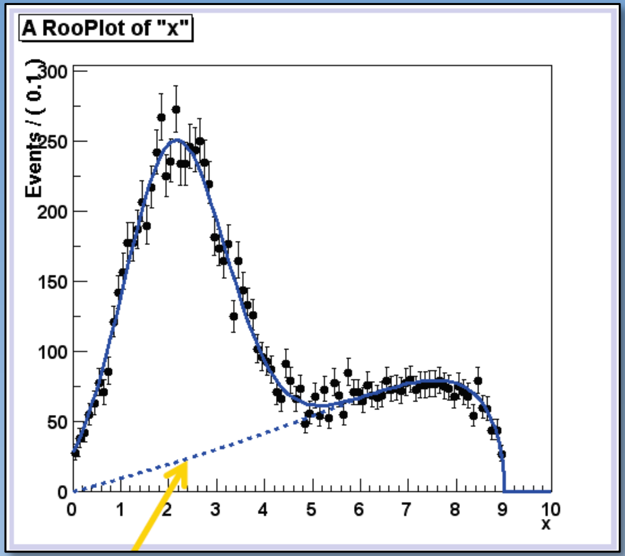
2-dimensional



3-dimensional



Root [3]: data modeling & hypothesis testing



Modeling & (numerical) convolutions

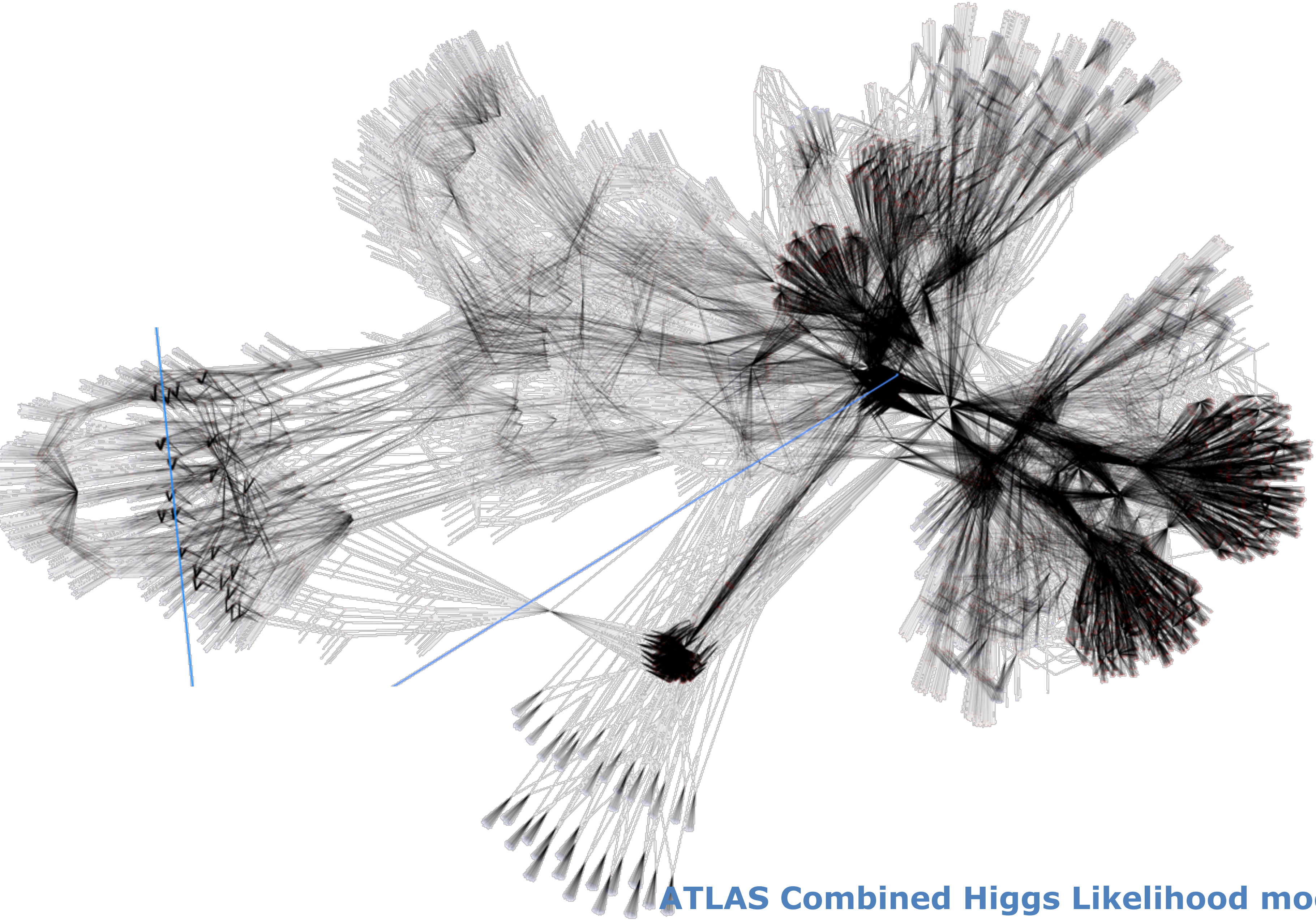
Statistics tools: Fitting, hypothesis testing, test-statistic, toy-MC, significance, limits, profiling

Multi-dimensional analyses: complex tools: Neural nets, Boosted decision trees etc.

→ **We'll do our own fit in the exercise session**

Root [1,2]: mathematics & physics tools

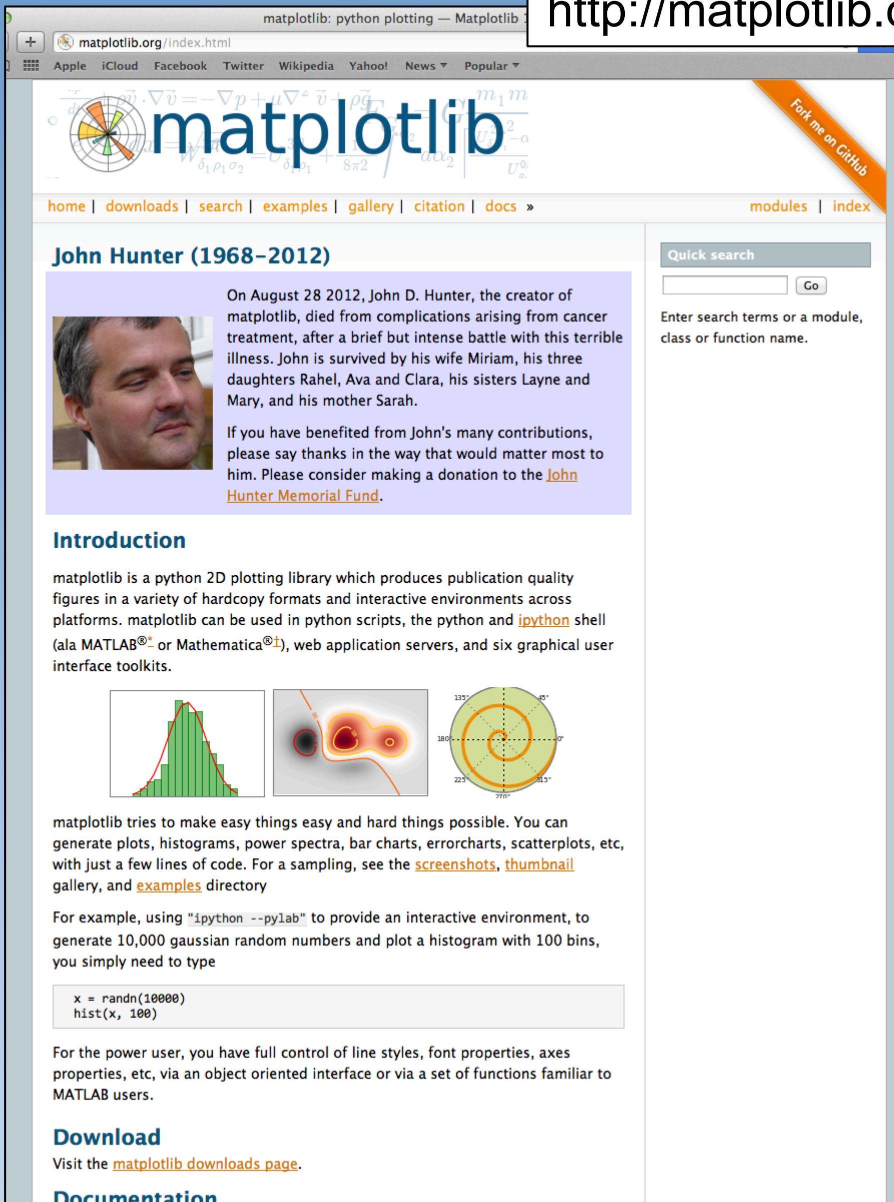
- Types: int, float, double, vectors, matrices, … but also Lorentz vectors etc.
  you can of course build your own class/object/structure
- Random numbers, matrix manipulation, PDG information, boosts
- Geant4, …

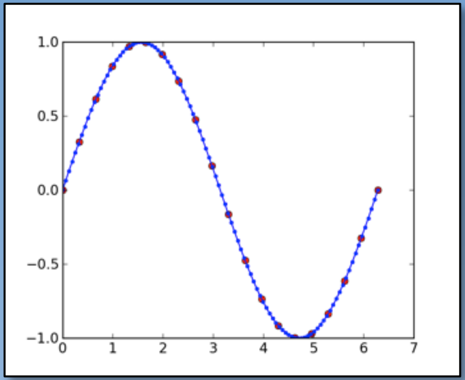


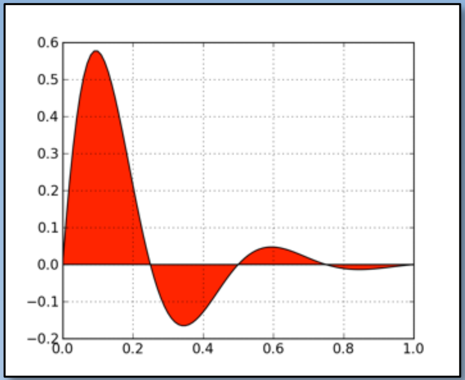
ATLAS Combined Higgs Likelihood model
(W. Verkerke)

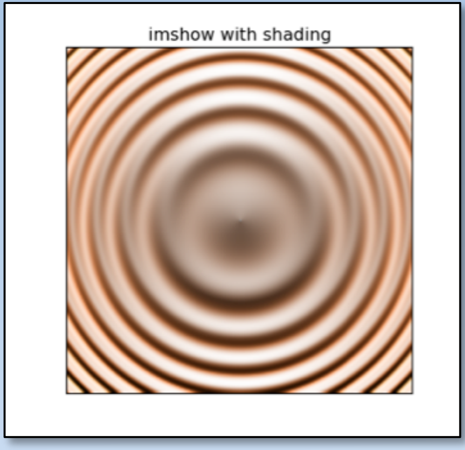
Data analysis and visualisation in Python (Matplotlib)

http://matplotlib.org







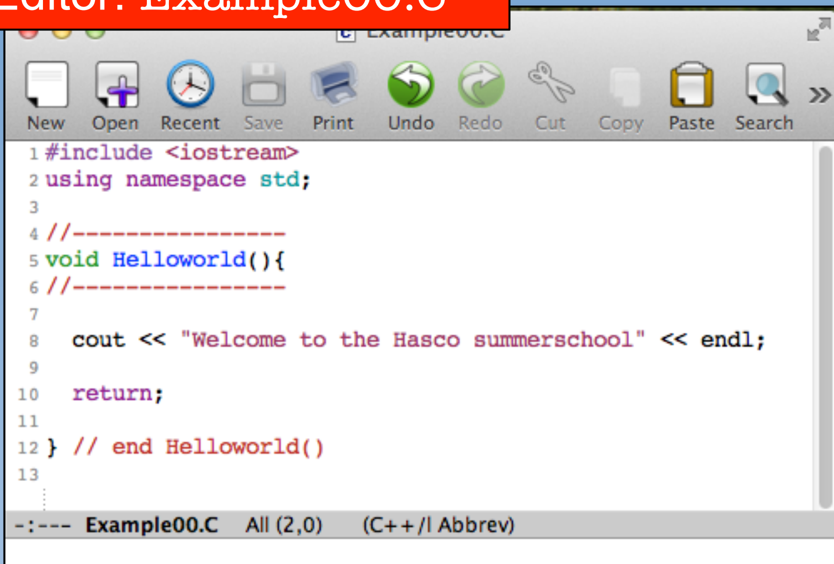


https://www.enthought.com/downloads/

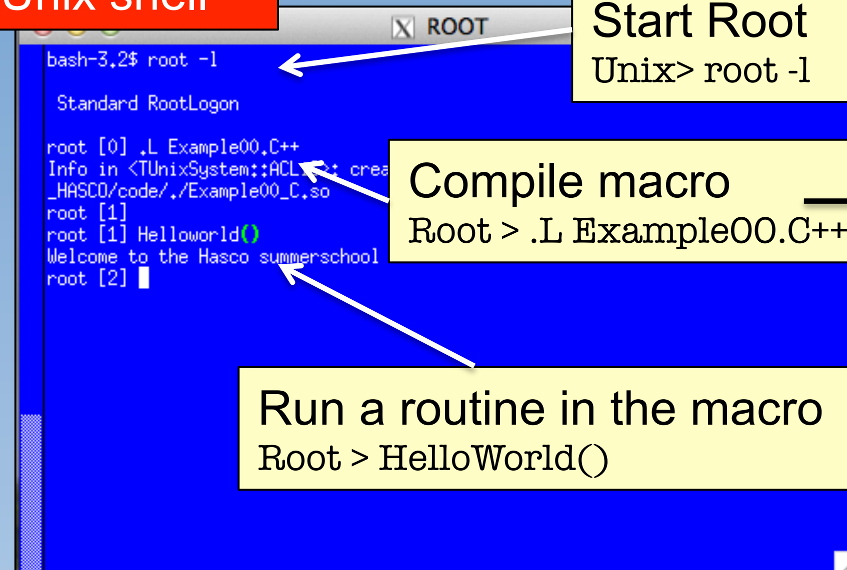
Running your first Root macro

This '++' 'compiles' your macro. Spot mistakes

Editor: Example00.C



Unix shell



Start Root
Unix> root -l

Compile macro
Root > .L Example00.C++

Run a routine in the macro
Root > HelloWorld()

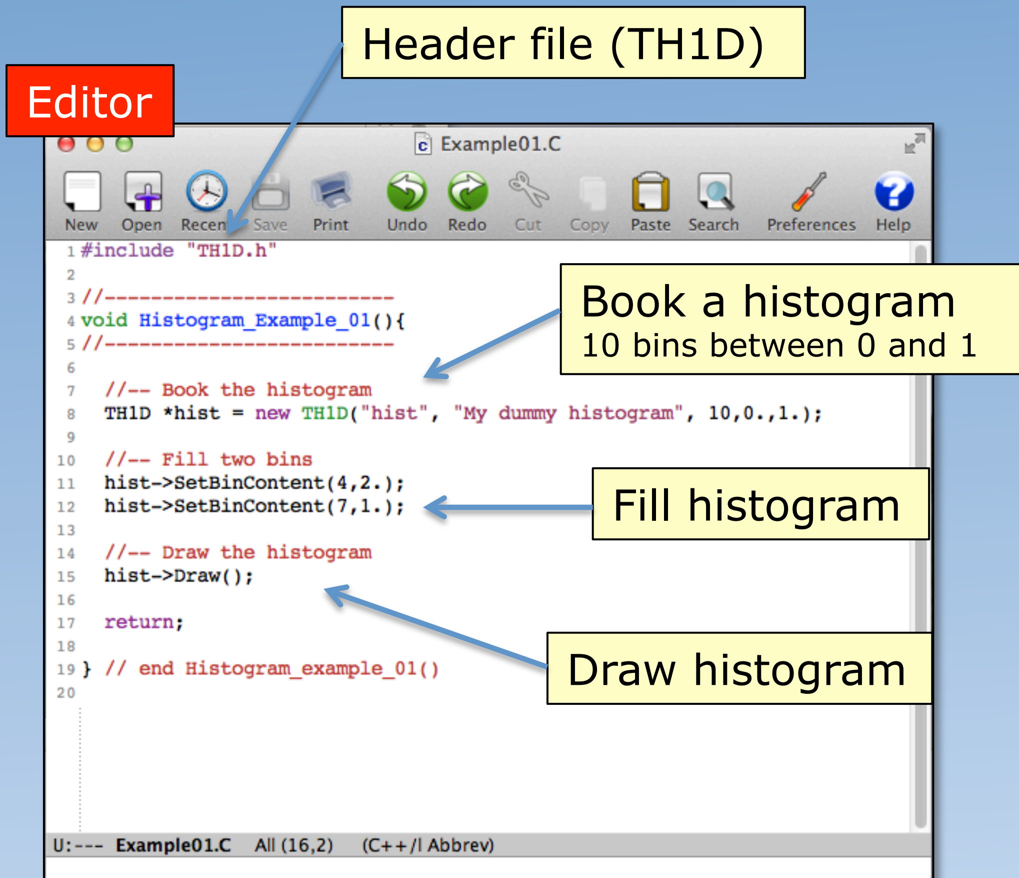
Note: When you start root it automatically processes the file **rootlogon.C**
This is a file where you can set all your default settings

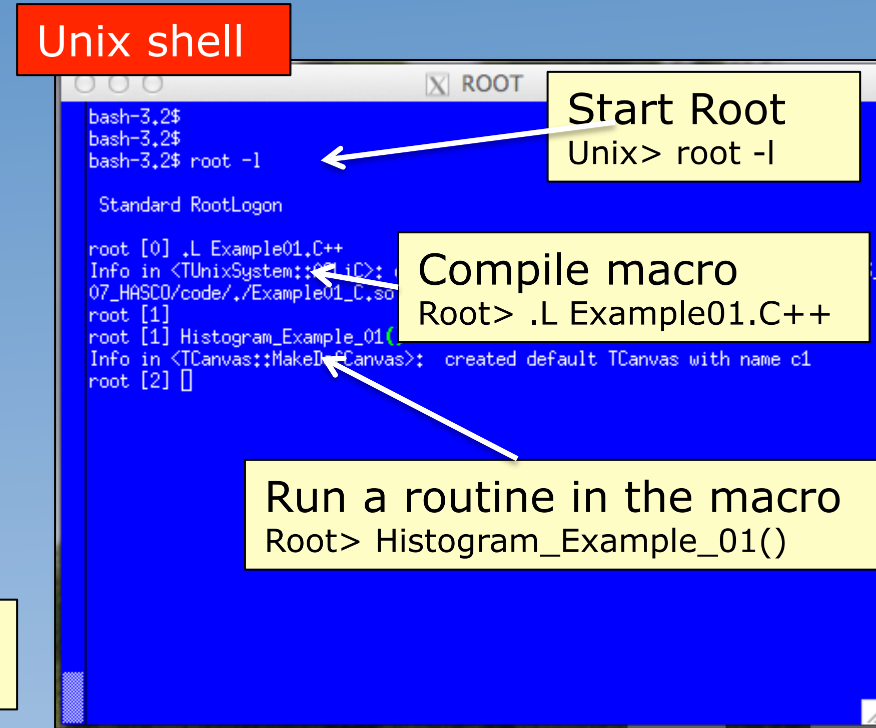
Histograms

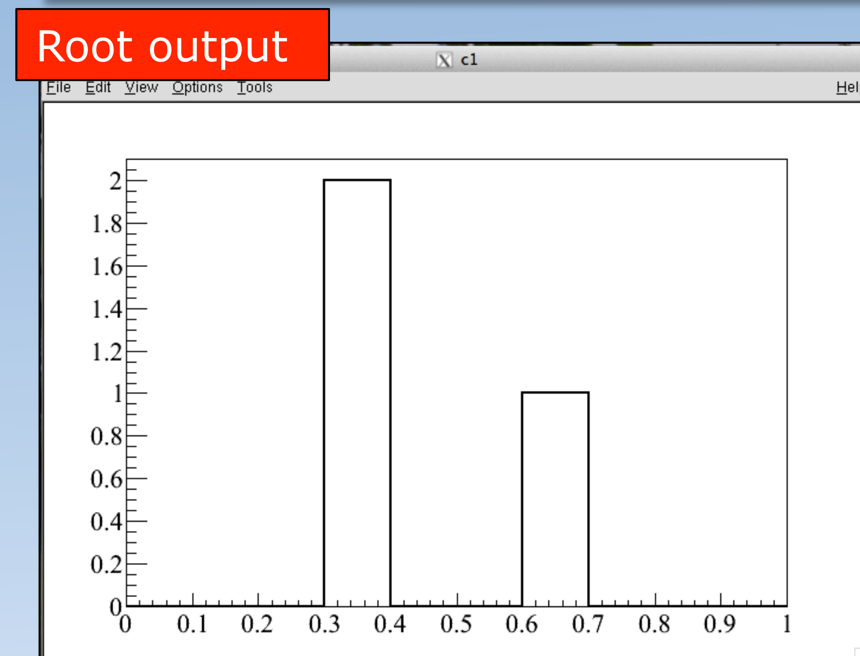
You'll use these during the exercise session

Histogram_Example_01.C

"plot a histogram on screen"

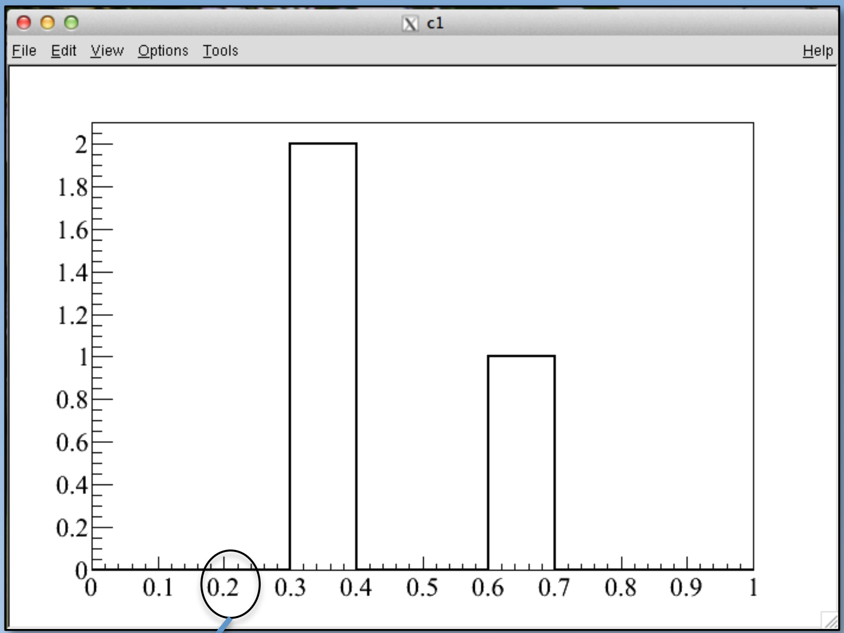






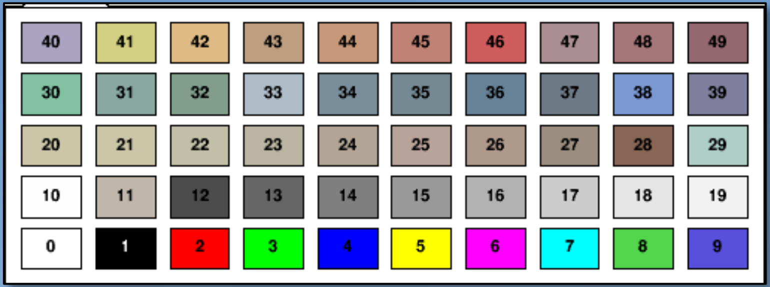
Inspect and style histograms

TH1:
http://root.cern.ch/root/html/TH1.html



Details:

hist->SetLabelColor()
hist->SetLabelFont()
hist->SetLabelOffset()
hist->SetLabelSize()



Style:

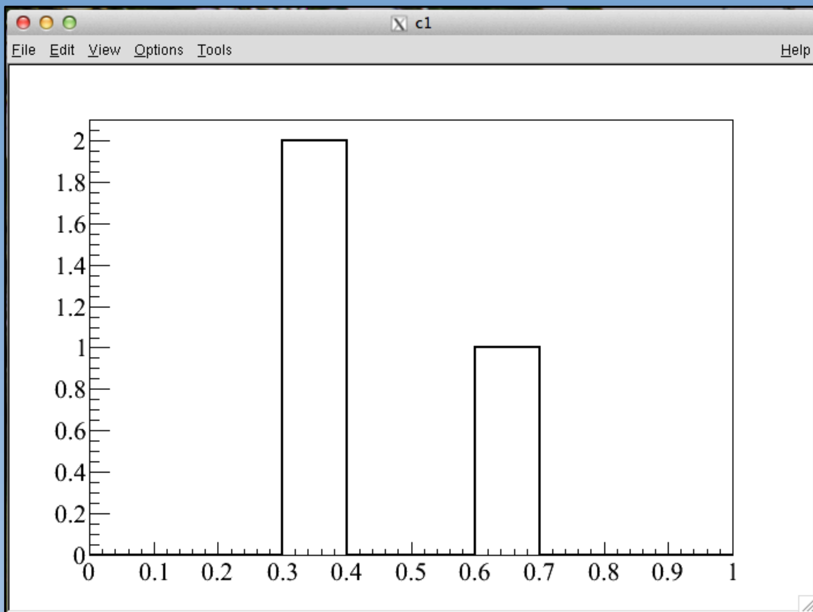
hist->SetFillColor(i_{color})
hist->SetFillStyle()
hist->SetLineColor()
hist->SetLineStyle()
hist->SetLineWidth
…

Inspect:

hist->GetRMS()
hist->GetMean()
hist->GetSumOfWeights()
hist->GetMaximumBin()
hist->GetBinContent(i_{bin})
hist->GetBinCenter(i_{bin})
hist->Integrate(…)
hist->Write(…)
…

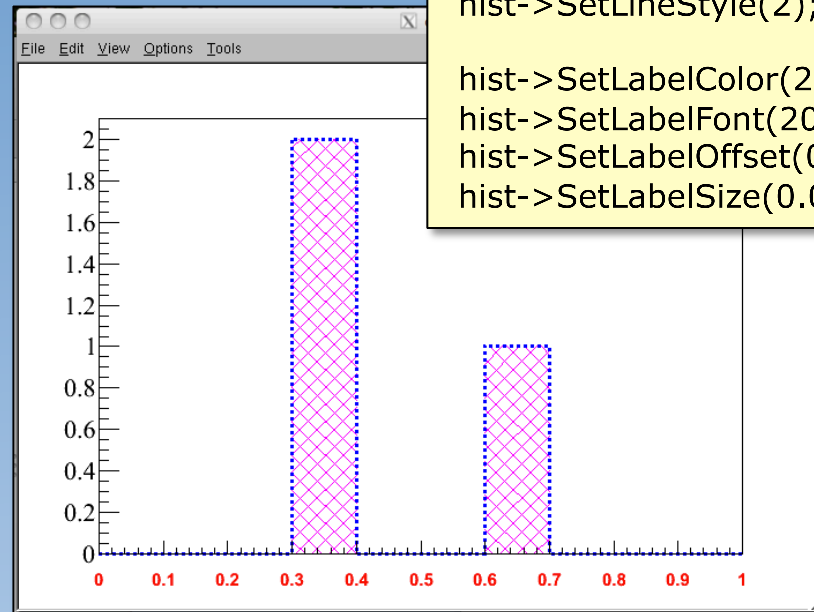
You'll need these in the exercises

Standard



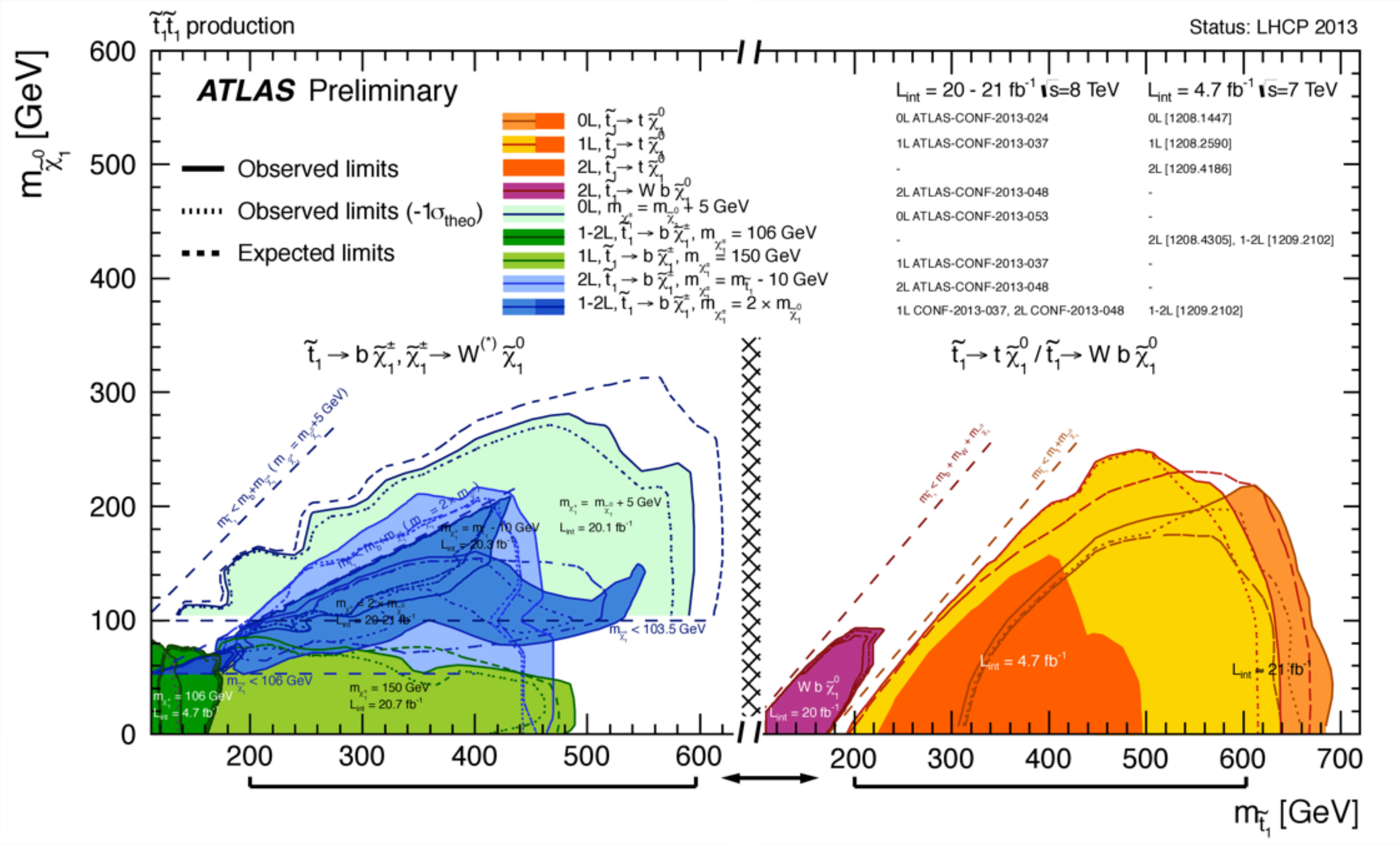
Your style

```
hist->SetFillColor(6);
hist->SetFillStyle(3544);

hist->SetLineColor(4);
hist->SetLineWidth(3);
hist->SetLineStyle(2);

hist->SetLabelColor(2);
hist->SetLabelFont(200);
hist->SetLabelOffset(0.03);
hist->SetLabelSize(0.035);
```



Note:
- beauty is in the eye of the beholder
- take some time to make the plot clear
- default settings in rootlogon.C

Summarizing your measurement is important ... and not easy



Tip: analysis takes O(1 year). Take > 1 minute for final plot

Histogram_Example_02.C

"write histogram to file"

TFile

Example02.C

```
#include "TH1D.h"
#include "TFile.h"

#include <iostream>
using namespace std;

//------------------------
void Histogram_example_02(){
//------------------------
//----------------------------------------
//-- Goal: o book histogram and fill 2 bins
//         o write histogram to file
//----------------------------------------

  //-- Book the histogram
  TH1D *hist = new TH1D("hist", "My dummy histogram", 10,0.,1.);

  //-- Fill two bins
  hist->SetBinContent(4,2.);
  hist->SetBinContent(7,1.);

  //-- Draw the histogram
  hist->Draw();

  //-- Save the histogram in a file
  TFile *myfile = new TFile("MySummaryFile.root", "recreate");
  hist->Write();
  myfile->Close();

  cout << " I just wrote the histogram to a file" << endl;

  return;

} // end Histogram_example_02()
```

-:--- Example02.C All (18,20) (C++/l Abbrev)

TFile:
http://root.cern.ch/root/html/TFile.html

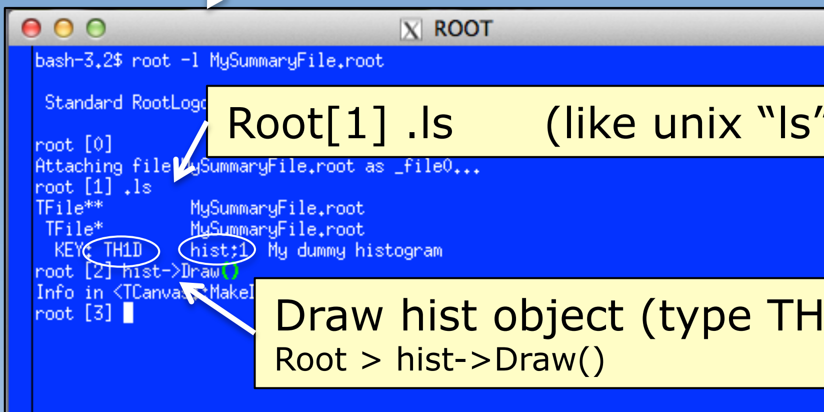
open a file called MySummaryFile.root, write histogram to it and close the file

Histogram_Example_03.C

"read a histogram from a file and plot it"

Unix> root –l MySummaryfile.root

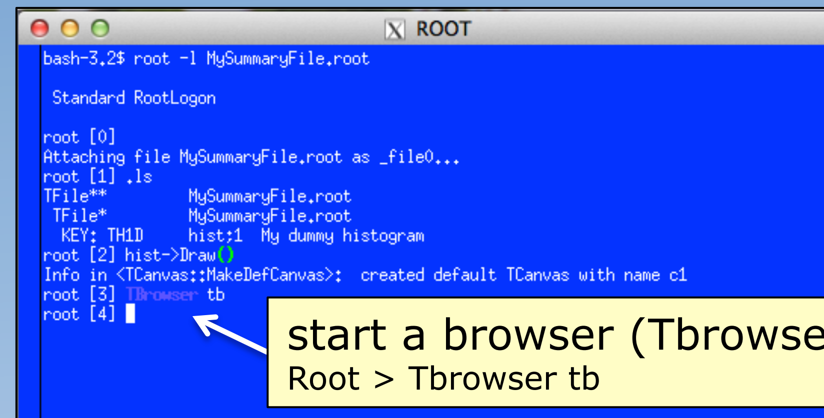
option 1: command line



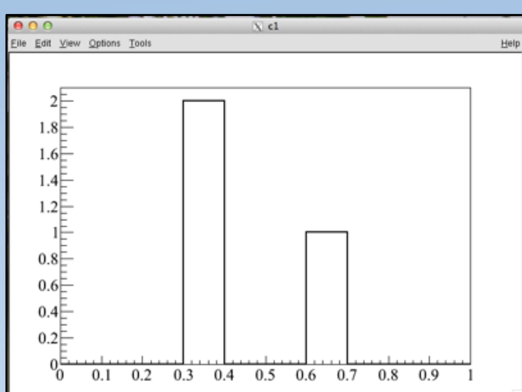
Root[1] .ls   (like unix "ls")

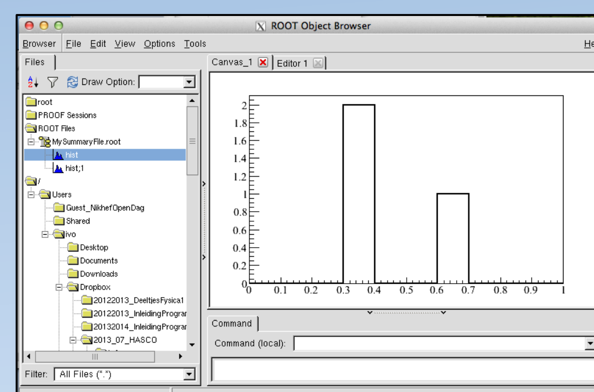
Draw hist object (type TH1D)
Root > hist->Draw()

option 2: browser / clicking



start a browser (Tbrowser)
Root > Tbrowser tb





Histogram_Example_03.C

"read a histogram from a file and plot it"

option 3: using a macro

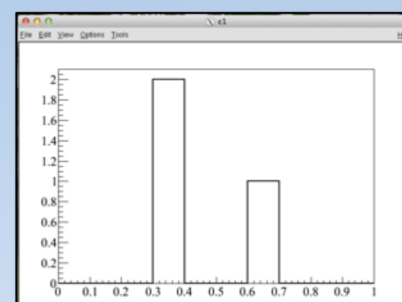
Example03.C

```
#include "TH1D.h"
#include "TFile.h"

#include <iostream>
using namespace std;



//-------------------------
void Histogram_example_03(){
//-------------------------
//---------------------------------------
//-- Goal: o read a histogram from a file
//         o make a copy and plot it
//---------------------------------------

  //-- open the file and make a copy of the histogram
  TDirectory* dir = gDirectory;
  TFile *file = new TFile("MySummaryFile.root", "READ");
  dir->cd();
  TH1D *hist_copy = (TH1D*) file->Get("hist")->Clone("hist_copy");
  file->Close();

  //-- Draw the histogram
  hist_copy->Draw();

  return;

} // end Histogram_example_03()
```

-:--- Example03.C All (32,0) (C++/l Abbrev)

→ *open the root file & make a copy of the histogram*

→ *draw histogram*



Histogram_Example_04.C

"Fill histogram with random numbers and add some text"

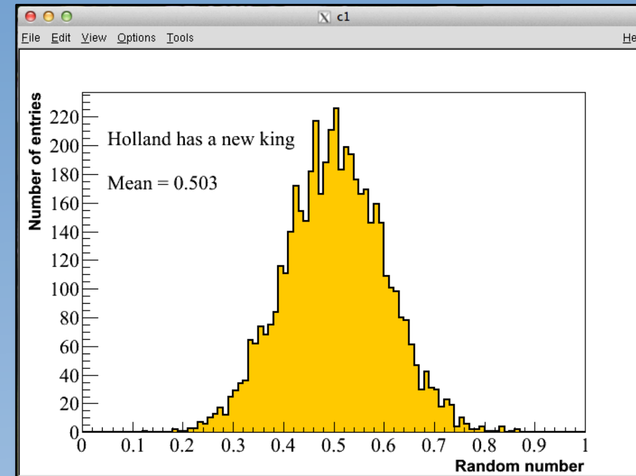
Example04.C

New Open Recent Save Print Undo Redo Cut Copy Paste Search Preferences Help

```
#include "TH1D.h"
#include "TColor.h"
#include "TLatex.h"
#include "TRandom3.h"

//--------------------------------------------
void Histogram_example_04(int Nrandom = 5000){
//--------------------------------------------
//------------------------------------------------------------------------
//-- Goal: o Fill a histogram with <Nrandom> numbers
//         o plot it and add some information
//------------------------------------------------------------------------

  //-- book histogram (100 bins between 0. and 1.)
  TH1D *hist = new TH1D("hist", "My dummy histogram", 100,0.,1.);

  //-- fill the histogram with Nrandom random numbers
  TRandom3 *R = new TRandom3();
  for(int i_random = 0; i_random < Nrandom ; i_random++ ){
    double random_number  = R->Gaus(0.5,0.1); // random number: center at 0.5 and sigma = 0.1
    hist->Fill(random_number);
  }

  //-- make histogram orange and draw it
  hist->SetFillColor(kOrange);
  hist->Draw();

  //-- add axes
  hist->SetXTitle("Random number");
  hist->SetYTitle("Number of entries");

  //-- add some text
  TLatex *text1 = new TLatex(0.05,200., "Holland has a new king");
  text1->SetTextSize(0.05);
  text1->Draw();

  //-- add some text (advanced)
  //-- plot the mean of the histogram at 75% of the hight of the most populated bin
  double mean = hist->GetMean();
  double most_populated_bin = hist->GetMaximumBin();
  double hight_most_populated_bin = hist->GetBinContent(most_populated_bin);
  TLatex *text2 = new TLatex(0.05,0.75*hight_most_populated_bin,Form("Mean = %5.3f",mean));
  text2->SetTextSize(0.05);
  text2->Draw();

  return;

} // end Histogram_example_04()
```

-:--- Example04.C Top (30,0) (C++/l Abbrev)

Beginning of buffer

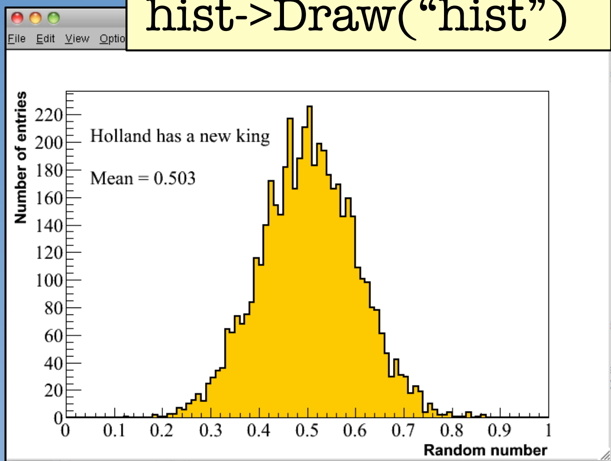


Fill histogram - random numbers

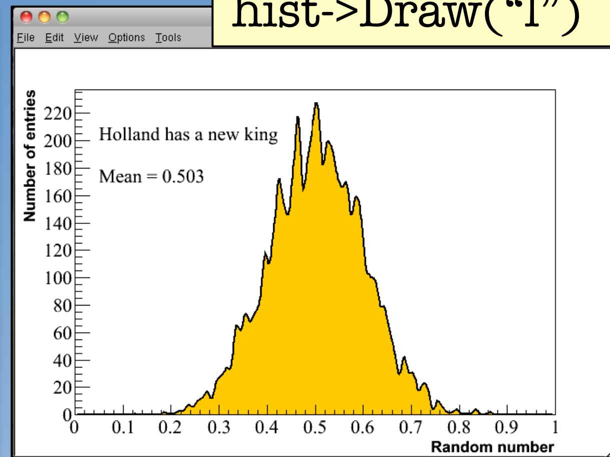
Give it your favorite color and add axes titles

Add some text in the plot

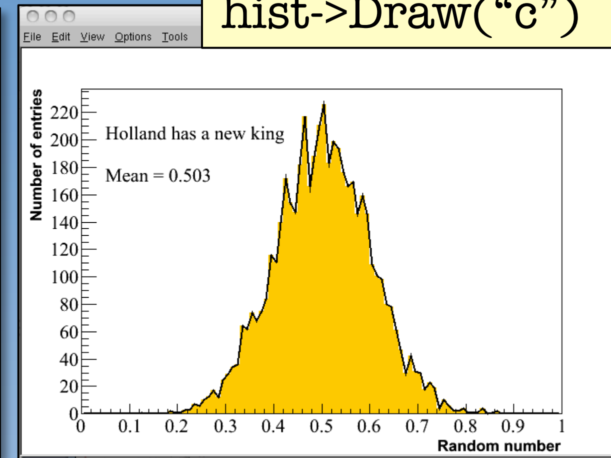
Plot the mean at 75% of the height of the maximum bin hist->Draw(“hist”)



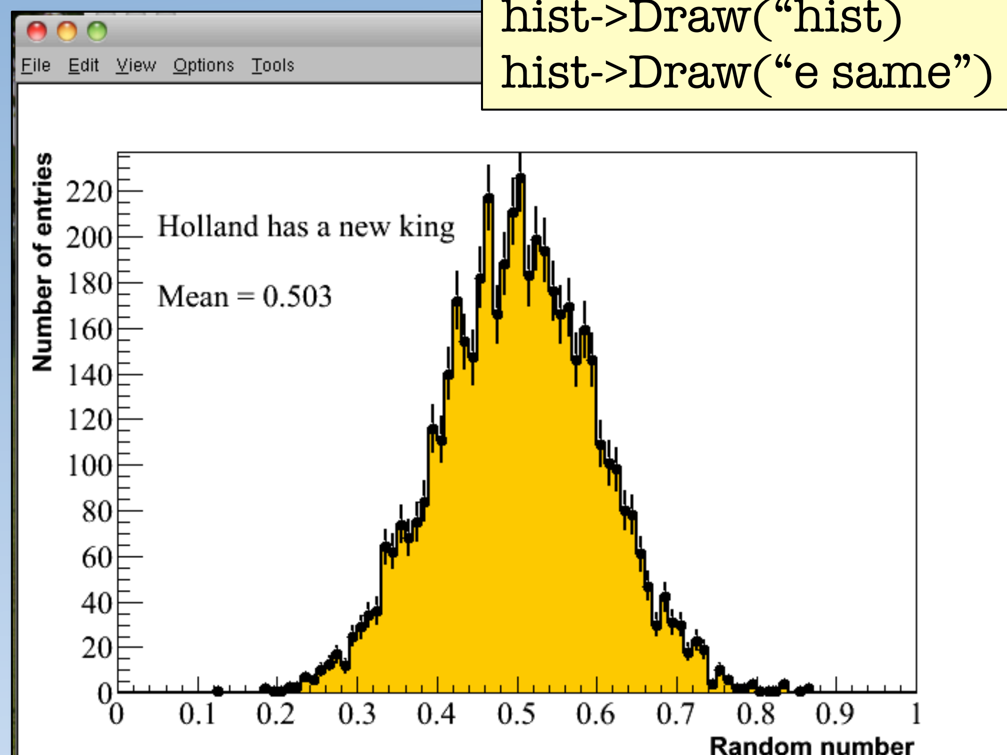
hist->Draw(“l”)



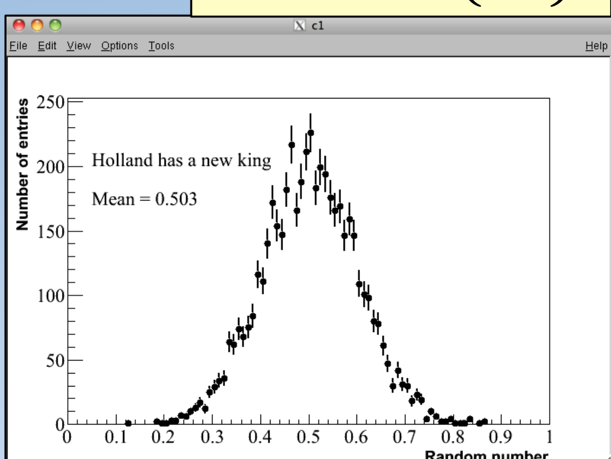
hist->Draw(“c”)



hist->Draw(“hist)
hist->Draw(“e same”)

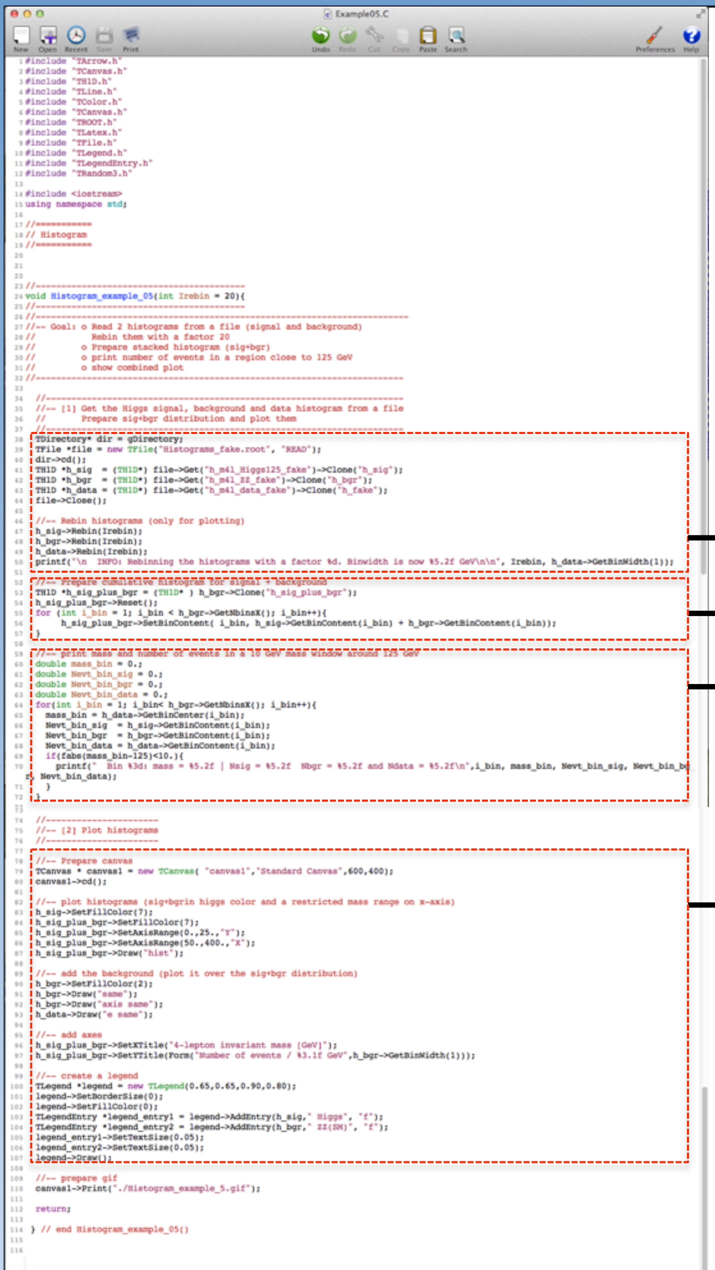


hist->Draw(“e”)

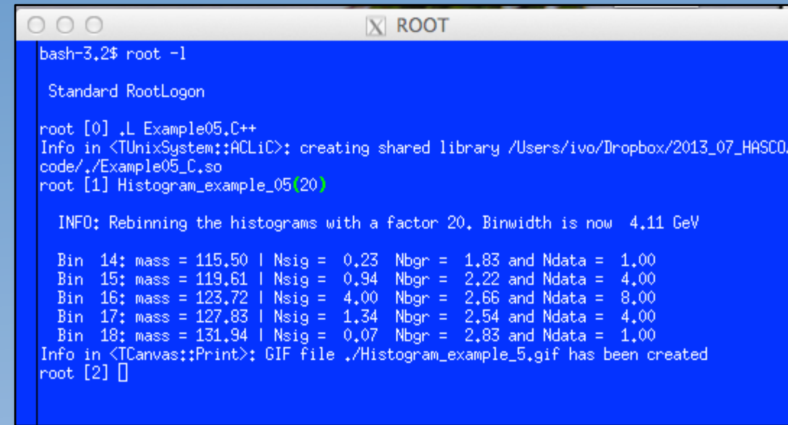


Histogram_Example_05.C

"Prepare a 'typical' Higgs discovery plot"



```cpp
#include "TArrow.h"
#include "TCanvas.h"
#include "TH1D.h"
#include "TLine.h"
#include "TColor.h"
#include "TCanvas.h"
#include "TROOT.h"
#include "TLatex.h"
#include "TFile.h"
#include "TLegend.h"
#include "TLegendEntry.h"
#include "TRandom3.h"

#include <iostream>
using namespace std;

//===========
// Histogram
//===========



//---------------------------------------
void Histogram_example_05(int Irebin = 20){
//---------------------------------------
//====================================================================
//-- Goal: o Read 2 histograms from a file (signal and background)
//            Rebin them with a factor 20
//          o Prepare stacked histogram (sig+bgr)
//          o print number of events in a region close to 125 GeV
//          o show combined plot
//====================================================================

  //==================================================================
  //-- [1] Get the Higgs signal, background and data histogram from a file
  //       Prepare sig+bgr distribution and plot them
  //==================================================================
  TDirectory* dir = gDirectory;
  TFile *file = new TFile("Histograms_fake.root", "READ");
  dir->cd();
  TH1D *h_sig  = (TH1D*) file->Get("h_m4l_Higgs125_fake")->Clone("h_sig");
  TH1D *h_bgr  = (TH1D*) file->Get("h_m4l_ZZ_fake")->Clone("h_bgr");
  TH1D *h_data = (TH1D*) file->Get("h_m4l_data_fake")->Clone("h_fake");
  file->Close();

  //-- Rebin histograms (only for plotting)
  h_sig->Rebin(Irebin);
  h_bgr->Rebin(Irebin);
  h_data->Rebin(Irebin);
  printf("\n  INFO: Rebinning the histograms with a factor %d. Binwidth is now %5.2f GeV\n\n", Irebin, h_data->GetBinWidth(1));

  //-- Prepare cumulative histogram for signal + background
  TH1D *h_sig_plus_bgr = (TH1D* ) h_bgr->Clone("h_sig_plus_bgr");
  h_sig_plus_bgr->Reset();
  for (int i_bin = 1; i_bin < h_bgr->GetNbinsX(); i_bin++){
       h_sig_plus_bgr->SetBinContent( i_bin, h_sig->GetBinContent(i_bin) + h_bgr->GetBinContent(i_bin));
  }

  //-- print mass and number of events in a 10 GeV mass window around 125 GeV
  double mass_bin = 0.;
  double Nevt_bin_sig = 0.;
  double Nevt_bin_bgr = 0.;
  double Nevt_bin_data = 0.;
  for(int i_bin = 1; i_bin< h_bgr->GetNbinsX(); i_bin++){
    mass_bin = h_data->GetBinCenter(i_bin);
    Nevt_bin_sig  = h_sig->GetBinContent(i_bin);
    Nevt_bin_bgr  = h_bgr->GetBinContent(i_bin);
    Nevt_bin_data = h_data->GetBinContent(i_bin);
    if(fabs(mass_bin-125)<10.){
      printf("  Bin %3d: mass = %5.2f | Nsig = %5.2f  Nbgr = %5.2f and Ndata = %5.2f\n",i_bin, mass_bin, Nevt_bin_sig, Nevt_bin_bgr, Nevt_bin_data);
    }
  }

  //---------------------
  //-- [2] Plot histograms
  //---------------------

  //-- Prepare canvas
  TCanvas * canvas1 = new TCanvas( "canvas1","Standard Canvas",600,400);
  canvas1->cd();

  //-- plot histograms (sig+bgrin higgs color and a restricted mass range on x-axis)
  h_sig->SetFillColor(7);
  h_sig_plus_bgr->SetFillColor(7);
  h_sig_plus_bgr->SetAxisRange(0.,25.,"Y");
  h_sig_plus_bgr->SetAxisRange(50.,400.,"X");
  h_sig_plus_bgr->Draw("hist");

  //-- add the background (plot it over the sig+bgr distribution)
  h_bgr->SetFillColor(2);
  h_bgr->Draw("same");
  h_bgr->Draw("axis same");
  h_data->Draw("e same");

  //-- add axes
  h_sig_plus_bgr->SetXTitle("4-lepton invariant mass [GeV]");
  h_sig_plus_bgr->SetYTitle(Form("Number of events / %3.1f GeV",h_bgr->GetBinWidth(1)));

  //-- create a legend
  TLegend *legend = new TLegend(0.65,0.65,0.90,0.80);
  legend->SetBorderSize(0);
  legend->SetFillColor(0);
  TLegendEntry *legend_entry1 = legend->AddEntry(h_sig," Higgs", "f");
  TLegendEntry *legend_entry2 = legend->AddEntry(h_bgr," ZZ(SM)", "f");
  legend_entry1->SetTextSize(0.05);
  legend_entry2->SetTextSize(0.05);
  legend->Draw();

  //-- prepare gif
  canvas1->Print("./Histogram_example_5.gif");

  return;

} // end Histogram_example_05()
```



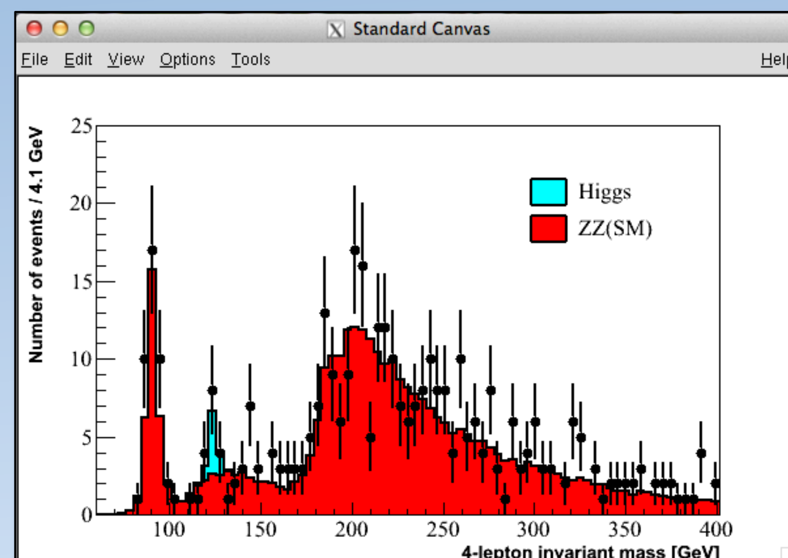
```
bash-3.2$ root -l

 Standard RootLogon

root [0] .L Example05.C++
Info in <TUnixSystem::ACLiC>: creating shared library /Users/ivo/Dropbox/2013_07_HASCO/code/./Example05_C.so
root [1] Histogram_example_05(20)

  INFO: Rebinning the histograms with a factor 20. Binwidth is now  4.11 GeV

  Bin  14: mass = 115.50 | Nsig =  0.23  Nbgr =  1.83 and Ndata =  1.00
  Bin  15: mass = 119.61 | Nsig =  0.94  Nbgr =  2.22 and Ndata =  4.00
  Bin  16: mass = 123.72 | Nsig =  4.00  Nbgr =  2.66 and Ndata =  8.00
  Bin  17: mass = 127.83 | Nsig =  1.34  Nbgr =  2.54 and Ndata =  4.00
  Bin  18: mass = 131.94 | Nsig =  0.07  Nbgr =  2.83 and Ndata =  1.00
Info in <TCanvas::Print>: GIF file ./Histogram_example_5.gif has been created
root [2]
```

→ *Read histograms from file. Also **rebin** them*

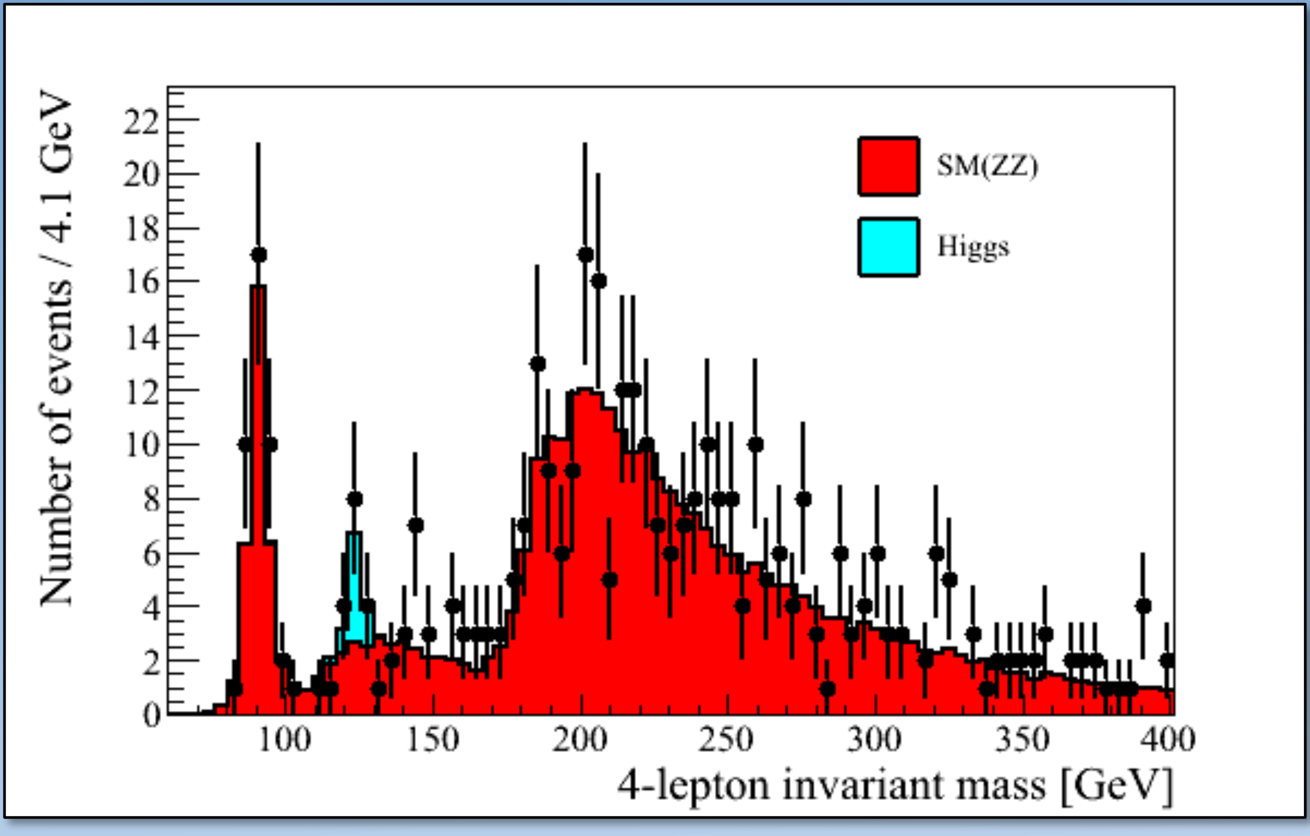
→ *Prepare cumulative histogram*

→ *Find #events (sig, bgr, data) around 125 GeV*

→ *plot*



Data-set for the exercises: 4 lepton mass



We'll try to squeeze as much info out of this distribution as we possibly can

Ntuples

You'll use something like this next week during Riccardo's lectures

Typical "event" consists of several (complex) objects

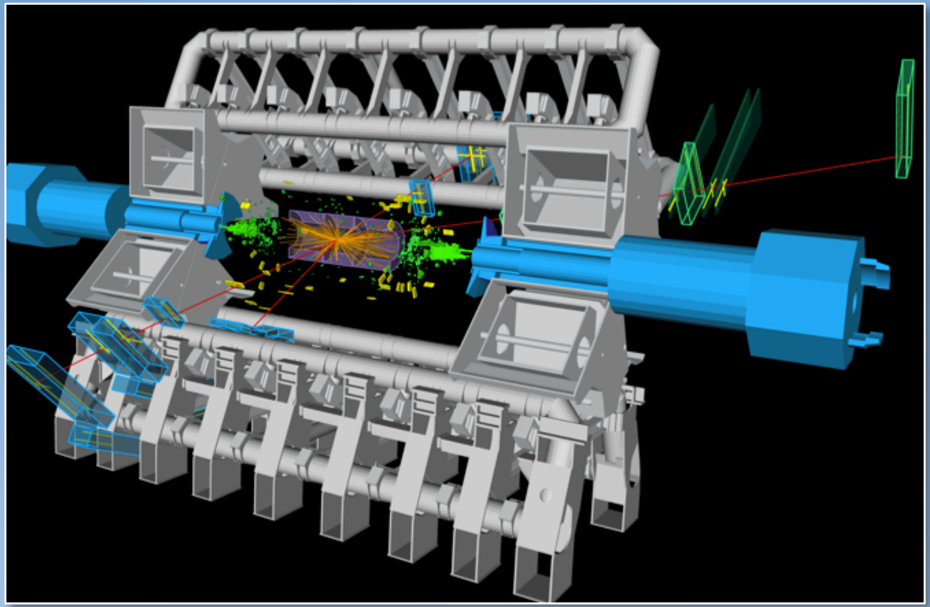
Single variables:
Event number, run number, lumi block, time, date, #muons, #tracks, #clusters, trigger info, …

Tracks:
Vector-like objects: (d_0, z_0, φ_0, θ, q/p), but also isolation, particle type, cluster links, …

Muons, electrons, jets
Muon algorithm types, track, energy, isolation corrections, …

Calorimeter clusters:
Pattern, elements, offsets, …

Jets:
Clusters, tracks, b-tag info, mass, …



Complex data structures and librairies

↓

Simplest 'flat' form is an ***Ntuple***

often the endpoint of an physics analysis

Ntuple_create.C

"Prepare a simple Ntuple and save in file"

```
#include "TH1D.h"
#include "TFile.h"
#include "TNtuple.h"
#include "TRandom3.h"

//------------------------
void NtupleCreation(){
//------------------------

  //-- Open a file
  TFile *file = new TFile("MySimpleNtuple.root","RECREATE");

  //-- Create an Ntuple
  TNtuple *ntuple = new TNtuple("ntuple", "my ntuple file", "x:y:z");

  //-- Fill the ntuple with 1000 random variables x,y,z
  double x, y,z;
  TRandom3 *R = new TRandom3();
  int Nevent = 1000;
  for(int i_event = 0; i_event < Nevent ; i_event++ ){
    x = R->Gaus(50.,5.);
    y = R->Gaus(50.,10.);
    z = R->Gaus(50.,15.);
    ntuple->Fill(x,y,z);
  }

  //-- Save the file
  file->Write();

  return;

} // end NtupleCreation()
```

```
Root > .L Ntuple_create.C++
Root > NtupleCreation()
```

Open the file: MySimpleNtuple.root

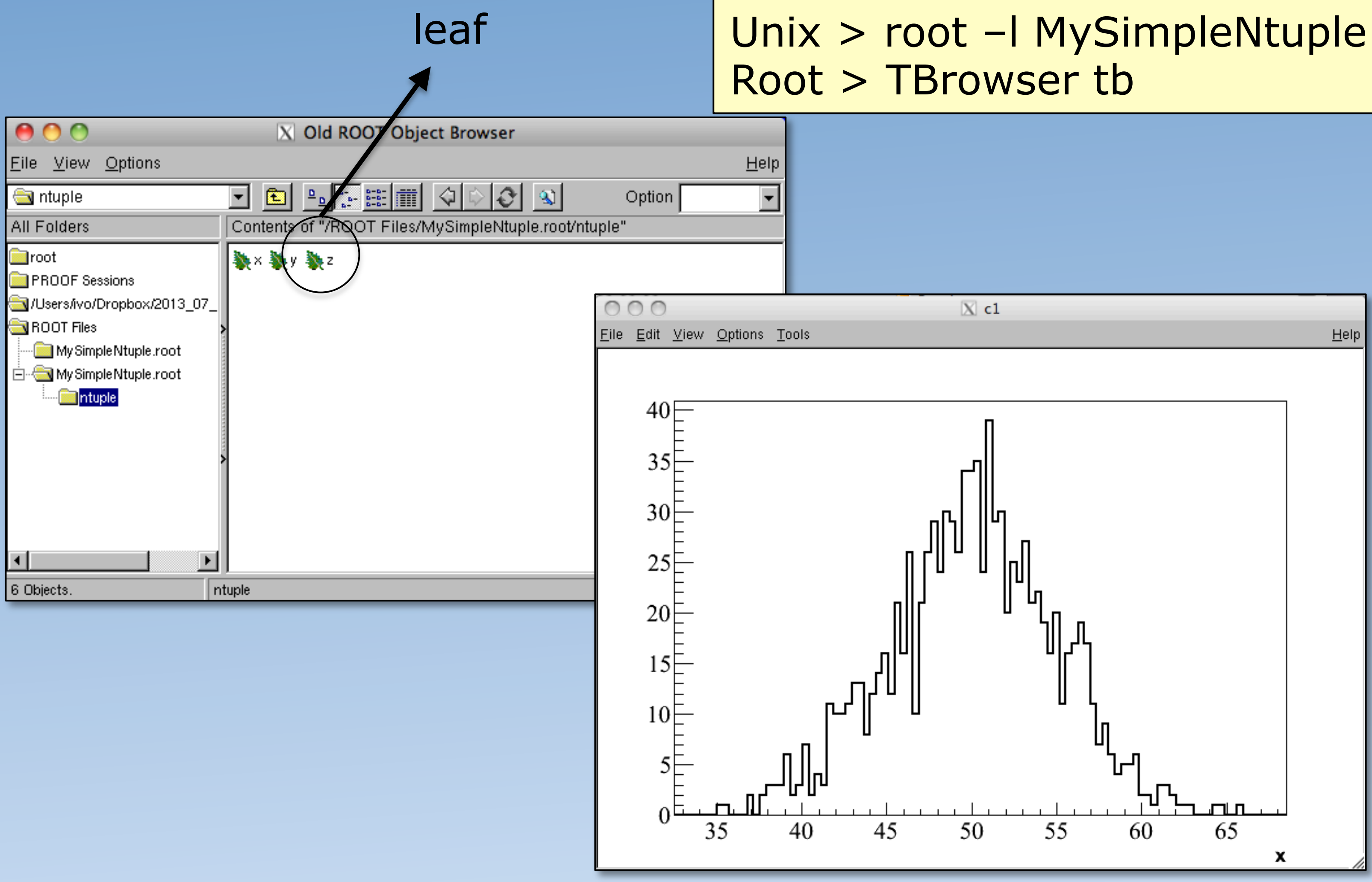
Create the Ntuple

Fill the Ntuple

Write the file

Looking at an Ntuple: browser

Unix > root –l MySimpleNtuple.root
Root > TBrowser tb



Looking at an Ntuple: MakeClass

```
Unix > root –l MySimpleNtuple.root
Root > .ls
Root > ntuple->MakeClass("MyNtupleAnalysis")
```

Header file: MyNtupleAnalysis.h

```
#ifndef MyNtupleReader_h
#define MyNtupleReader_h

#include <TROOT.h>
#include <TChain.h>
#include <TFile.h>

class MyNtupleReader {
public :
   TTree          *fChain;   //!pointer to the analyzed TTree or TChain
   Int_t           fCurrent; //!current Tree number in a TChain

   // Declaration of leaf types
   Float_t         x;
   Float_t         y;
   Float_t         z;

   // List of branches
   TBranch        *b_x;   //!
   TBranch        *b_y;   //!
   TBranch        *b_z;   //!

   MyNtupleReader(TTree *tree=0);
   virtual ~MyNtupleReader();
   virtual Int_t    Cut(Long64_t entry);
   virtual Int_t    GetEntry(Long64_t entry);
   virtual Long64_t LoadTree(Long64_t entry);
   virtual void     Init(TTree *tree);
   virtual void     Loop();
   virtual Bool_t   Notify();
   virtual void     Show(Long64_t entry = -1);
};

-:---  MyNtupleReader.h    8% (31,0)    (C/l Abbrev)
```

→ *Variables & types*

→ *GetEntry()*

→ *Loop()*

C-file: MyNtupleAnalysis.C

```
#define MyNtupleReader_cxx
#include "MyNtupleReader.h"
#include <TH2.h>
#include <TStyle.h>
#include <TCanvas.h>

void MyNtupleReader::Loop()
{
//   In a ROOT session, you can do:
//      Root > .L MyNtupleReader.C
//      Root > MyNtupleReader t
//      Root > t.GetEntry(12); // Fill t data members with entry number 12
//      Root > t.Show();       // Show values of entry 12
//      Root > t.Show(16);     // Read and show values of entry 16
//      Root > t.Loop();       // Loop on all entries
//

   if (fChain == 0) return;

   Long64_t nentries = fChain->GetEntriesFast();

   Long64_t nbytes = 0, nb = 0;
   for (Long64_t jentry=0; jentry<nentries;jentry++) {
      Long64_t ientry = LoadTree(jentry);
      if (ientry < 0) break;
      nb = fChain->GetEntry(jentry);   nbytes += nb;
      // if (Cut(ientry) < 0) continue;
   }
}

-:---  MyNtupleReader.C   All (29,1)    (C++/l Abbrev)
```

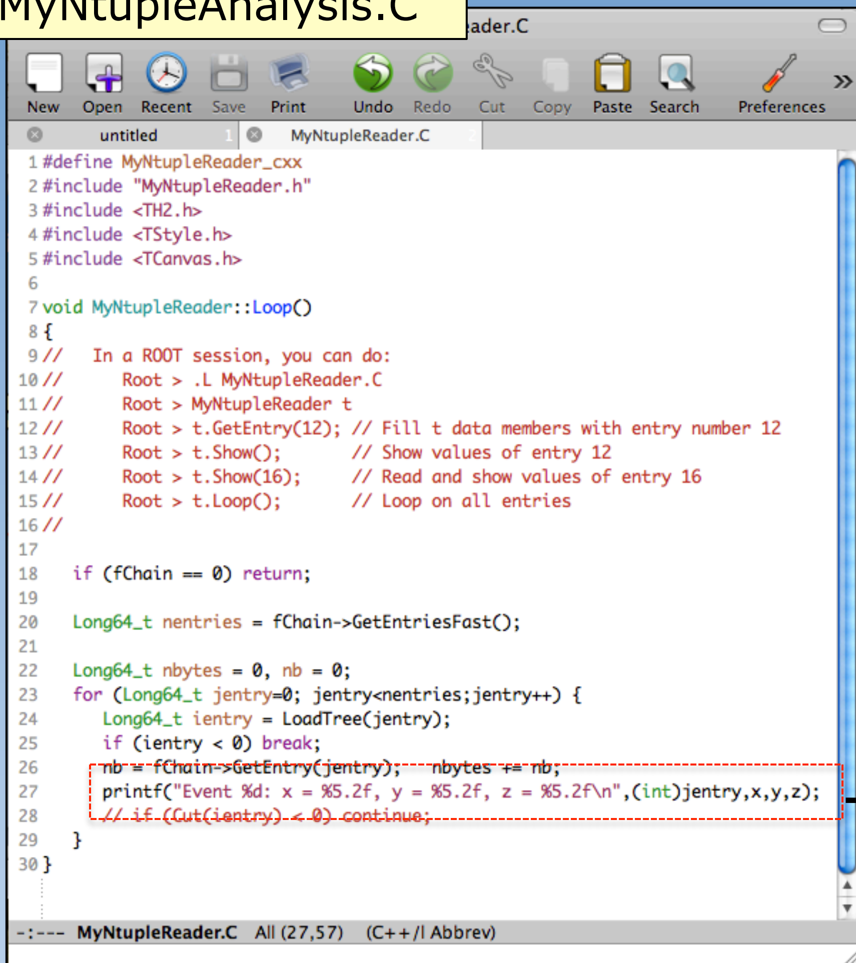
→ *Loop()*

→ *Loop over entries*

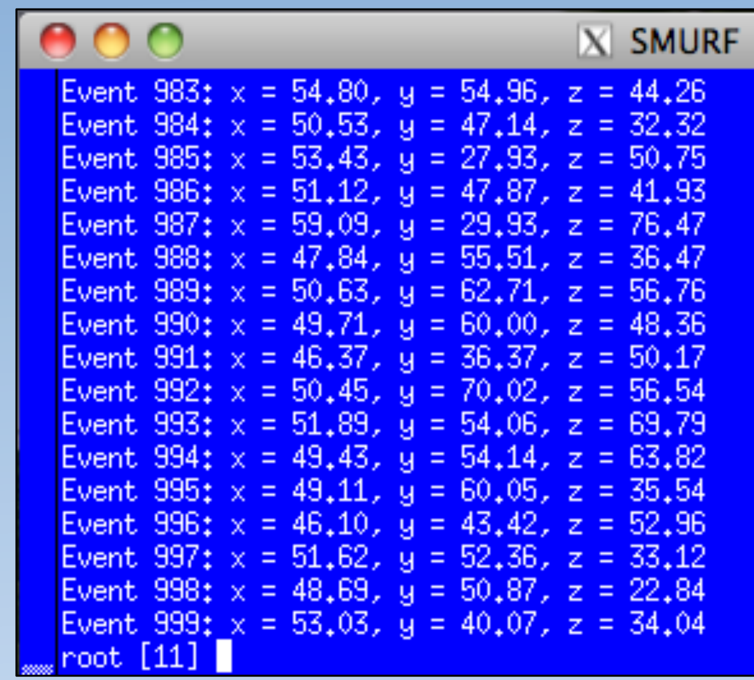
→ *Somewhere a link to MySimpleNtuple.root*

Running your code

MyNtupleAnalysis.C



Unix > root
Root > .L MyNtupleReader.C++
Root > MyNtupleReader mnr
Root > mnr.Loop()



ONE extra line to MyNtupleAnalysis.C:

printf("Event %d: x = %5.2f, y = %5.2f, z = %5.2f\n",(int)jentry,x,y,z);

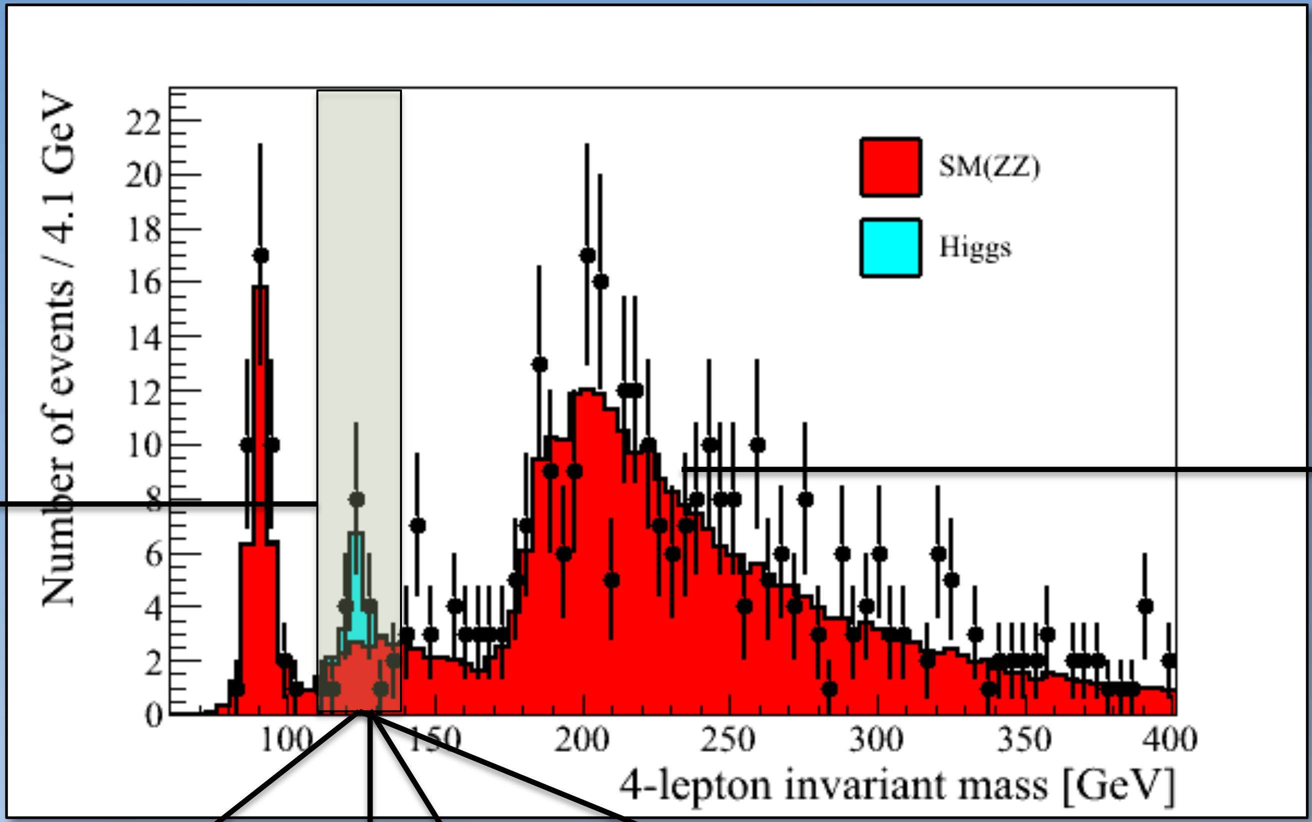
Statistics

Statistics

1) Hypotheses testing & search:
Significance (definition and optimization)

2) Measurement:
Side-band (likelihood) fit to estimate background

Data-set for the exercises: 4 lepton mass



Cross-section measurement

Exclusion

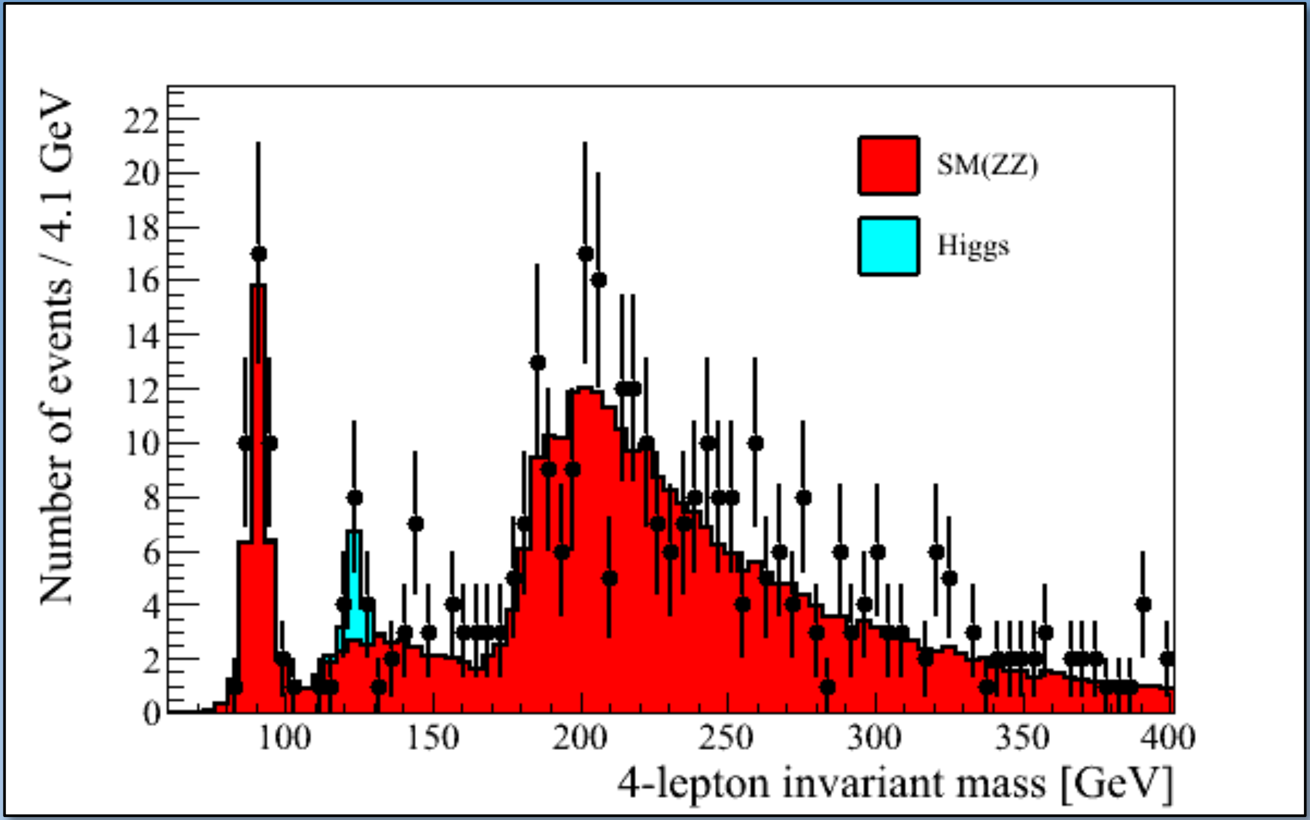
Significance optimization

Mass measurement

Test statistic (Toy-MC)

Data-driven background estimate (likelihood fit)

Data-set for the exercises: 4 lepton mass

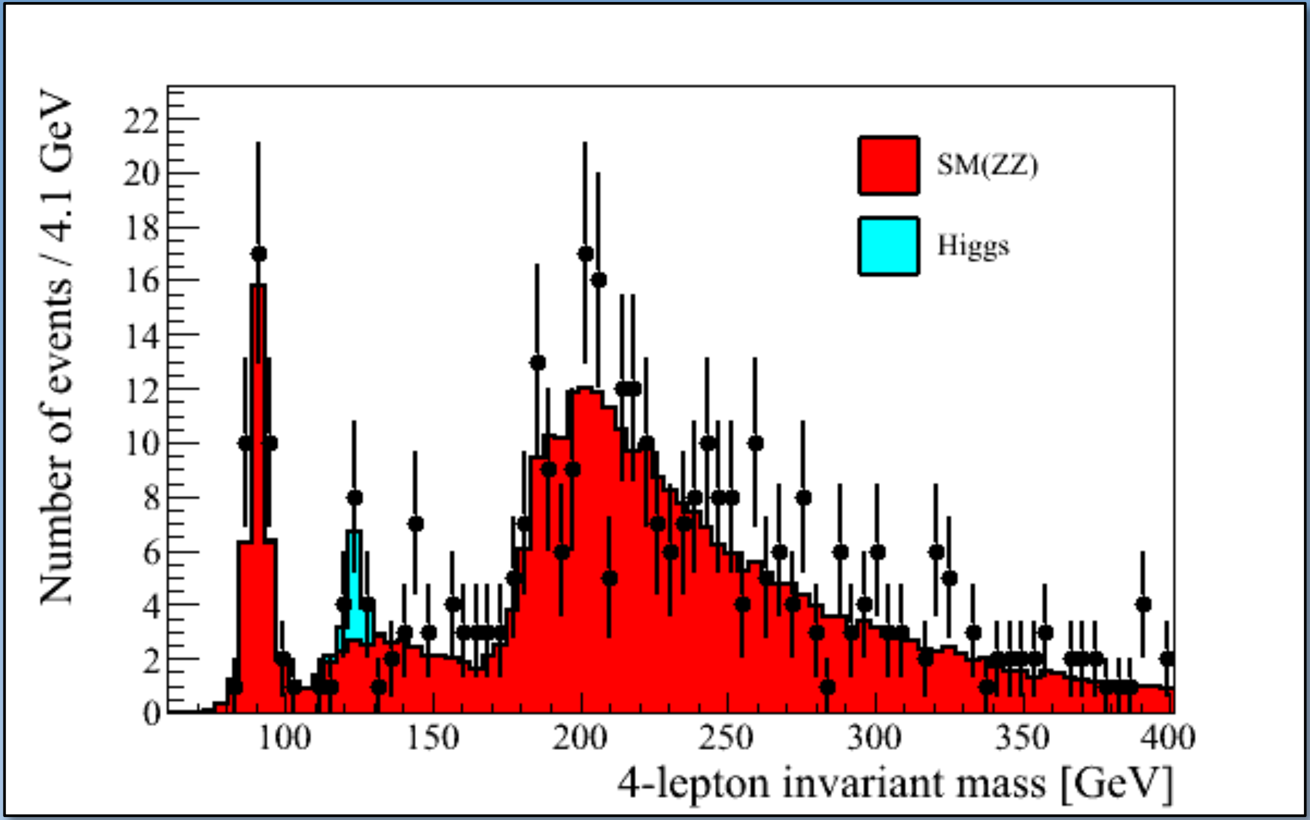


Full set of exercises:

https://www.nikhef.nl/~ivov/Talks/2013_03_21_DESY_HiggsExercises.pdf

Today we will only do a few of them

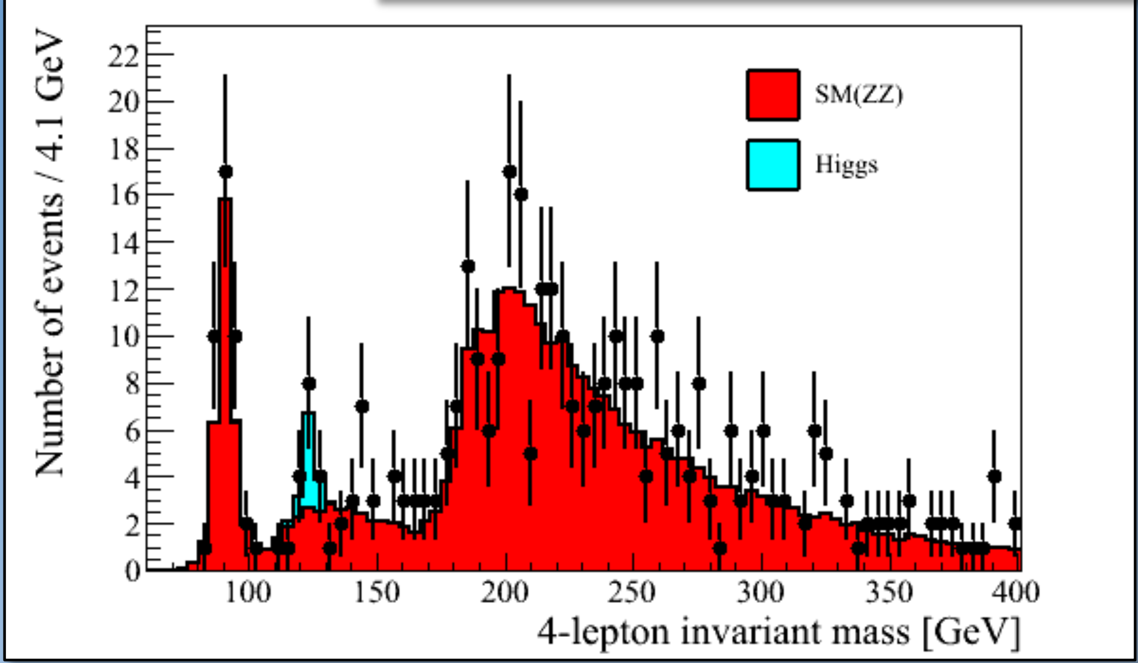
Data-set for the exercises: 4 lepton mass



Note: - Original histograms have 200 MeV bins

- This is fake data

In the 1st exercise we use re-binned histograms (4 GeV bins) our fake 4-lepton mass distribution



Set 1: Measurements

1. Data-driven background estimate (sideband likelihood fit + Poisson)

Optional:

2. Higgs signal cross-section measurement
3. *Optional: simultaneous mass + signal cross-section measurement*

Set 2: hypothesis testing

1. Significance optimization (counting)

Optional:

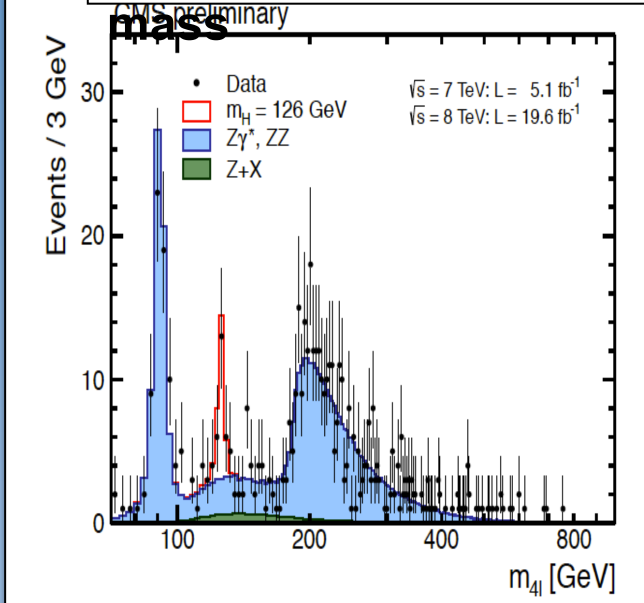
2. Compute test statistic (beyond counting)
3. Toy-MC and test statistic distribution
4. Interpretation: discovery
5. Interpretation: exclusion

Statistics

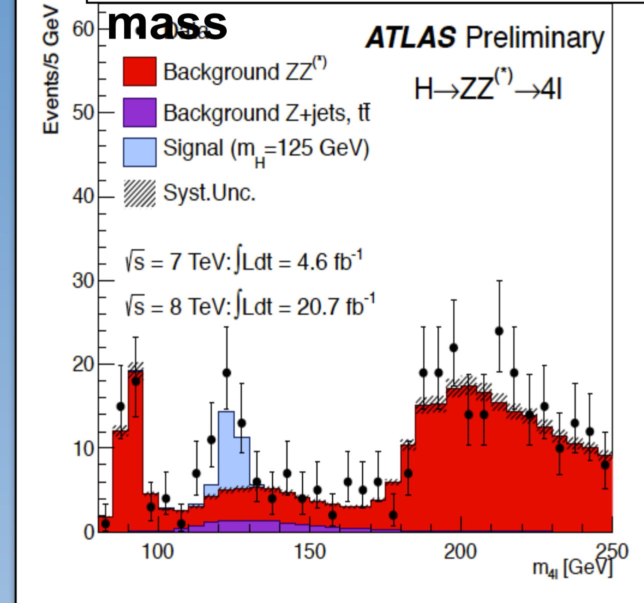
1) Hypotheses testing & search:
Significance (definition and optimization)

2) Measurement:
Side-band (likelihood) fit to estimate background

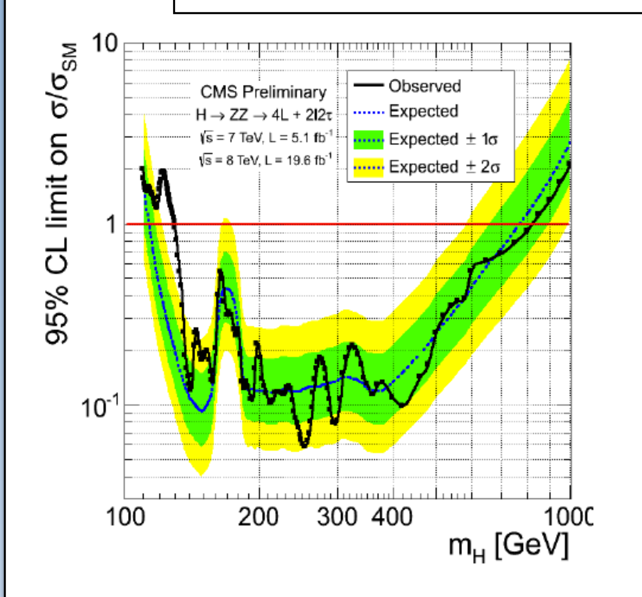
CMS 4 lepton invariant mass



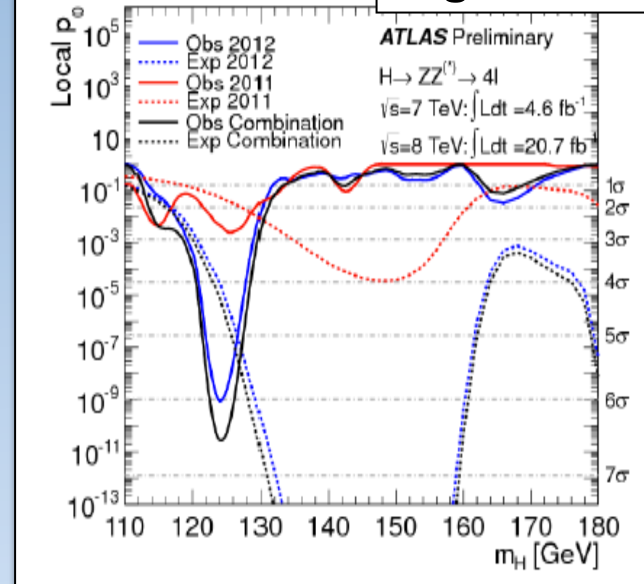
ATLAS 4 lepton invariant mass



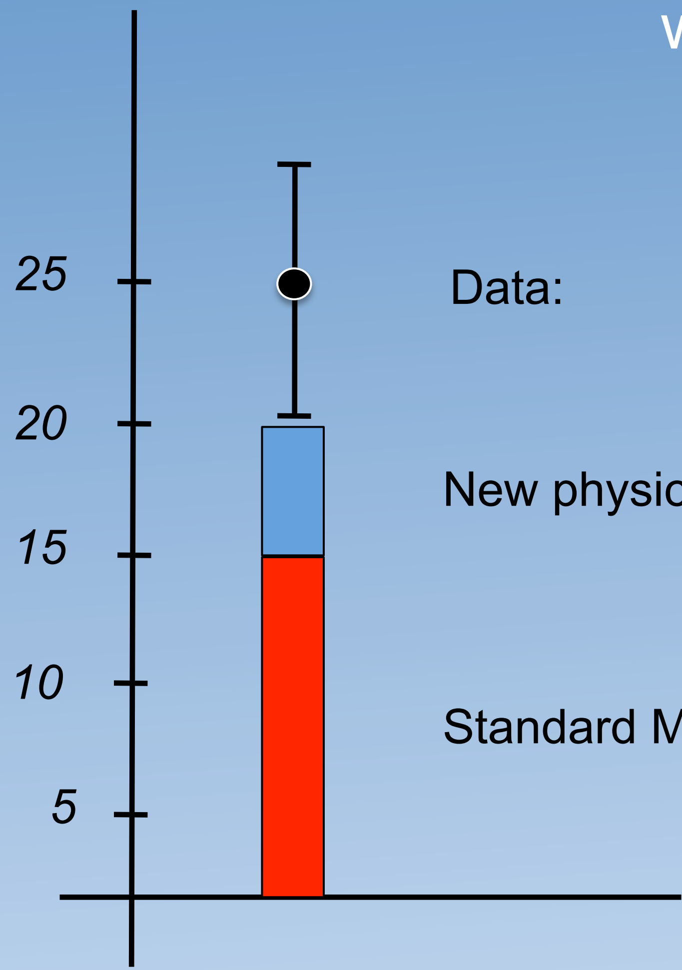
Excluded cross-sections



Significances



General remark : what is the significance ?



Data: $N_{data} = 25$

New physics: $N_{sig} = 5$

Standard Model: $N_{bgr} = 15$

Significance: probability to observe N events (or even more) under the background-only hypothesis

Observed significance:

$$\int_{25}^{\infty} \text{Poisson}(N \mid 15)\, dN = 0.0112 \quad \leftarrow \textit{p-value}$$

$$= 2.28 \text{ sigma} \quad \leftarrow \textit{significance}$$

Expected significance:

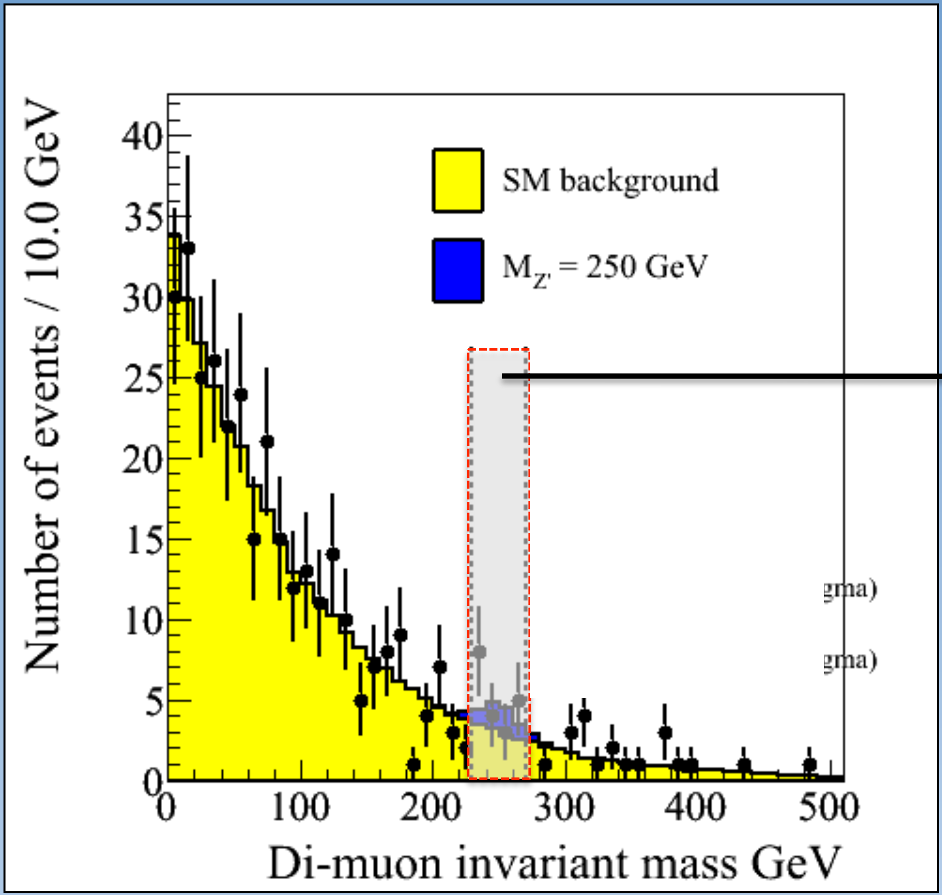
$$\int_{20}^{\infty} \text{Poisson}(N \mid 15)\, dN = 0.1248$$

$$= 1.15 \text{ sigma}$$

Discovery if p-value < 2.87×10^{-7}

→ 39 events

Counting events in a mass window



Standard Model

SM	10
Higgs	5
Data	12

Ok, now what ?

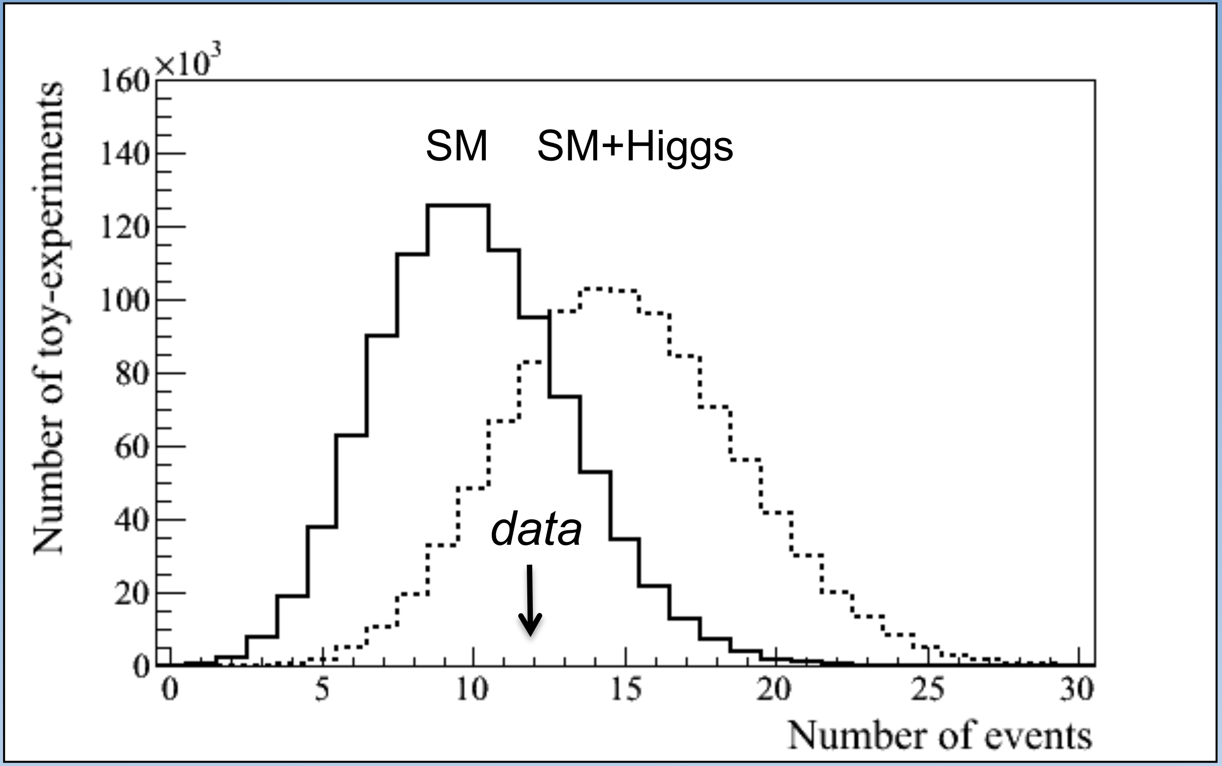
Significance: probability to observe N events (or even more) under the background-only hypothesis

Standard Model

SM	10
Higgs	5
Data	12

Ok, now what ?

Poisson distribution



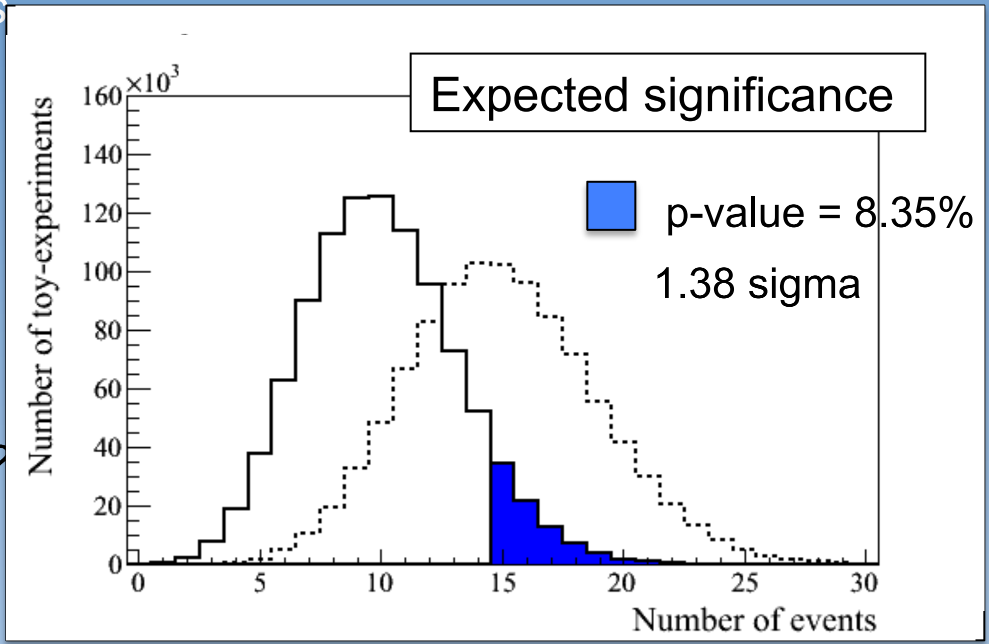
Interpretation

optimistic: discovery

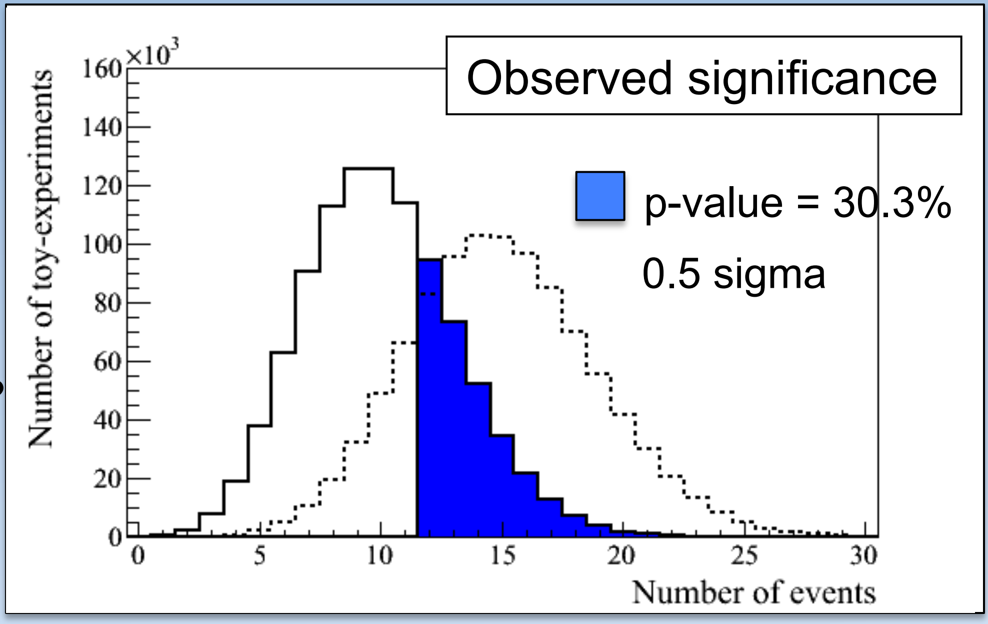
Discovery-aimed: p-value and significance

incompatibiliy with SM-only hypothesis

SM	10
Higgs	5
Data	12



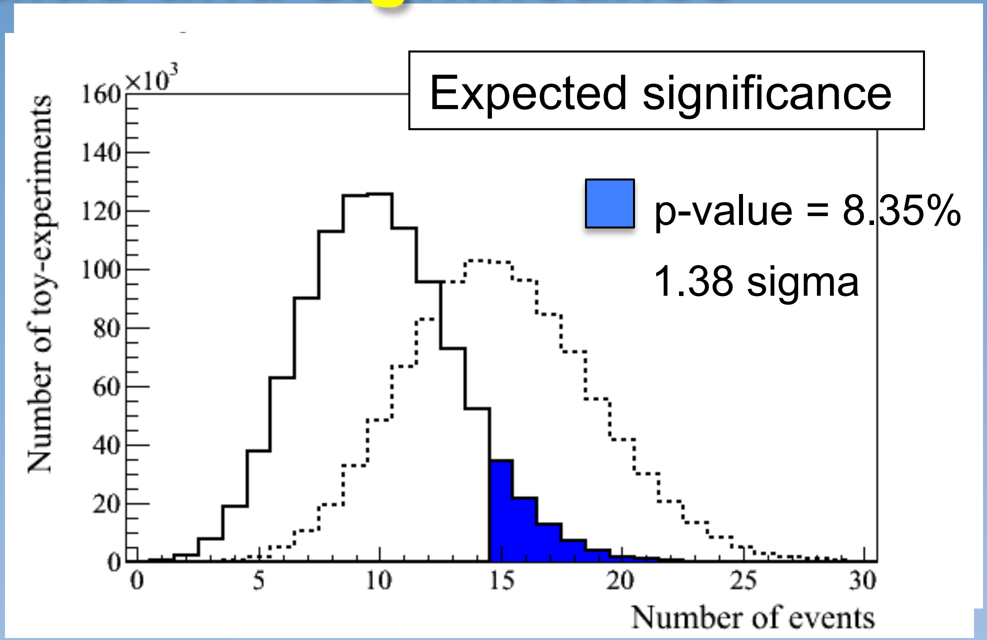
*1) What is the **expected** significance ?*



*2) What is the **observed** significance ?*

Discovery-aimed: p-value and significance

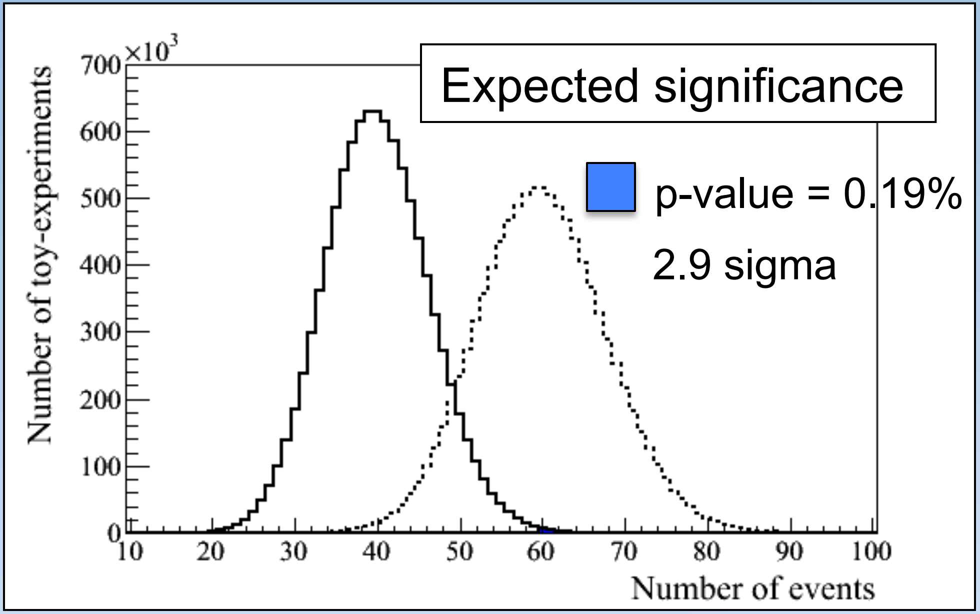
SM	10
Higgs	5



3) At what Lumi do you expect to be able to claim a discovery ?

3 times more LUMINOSITY

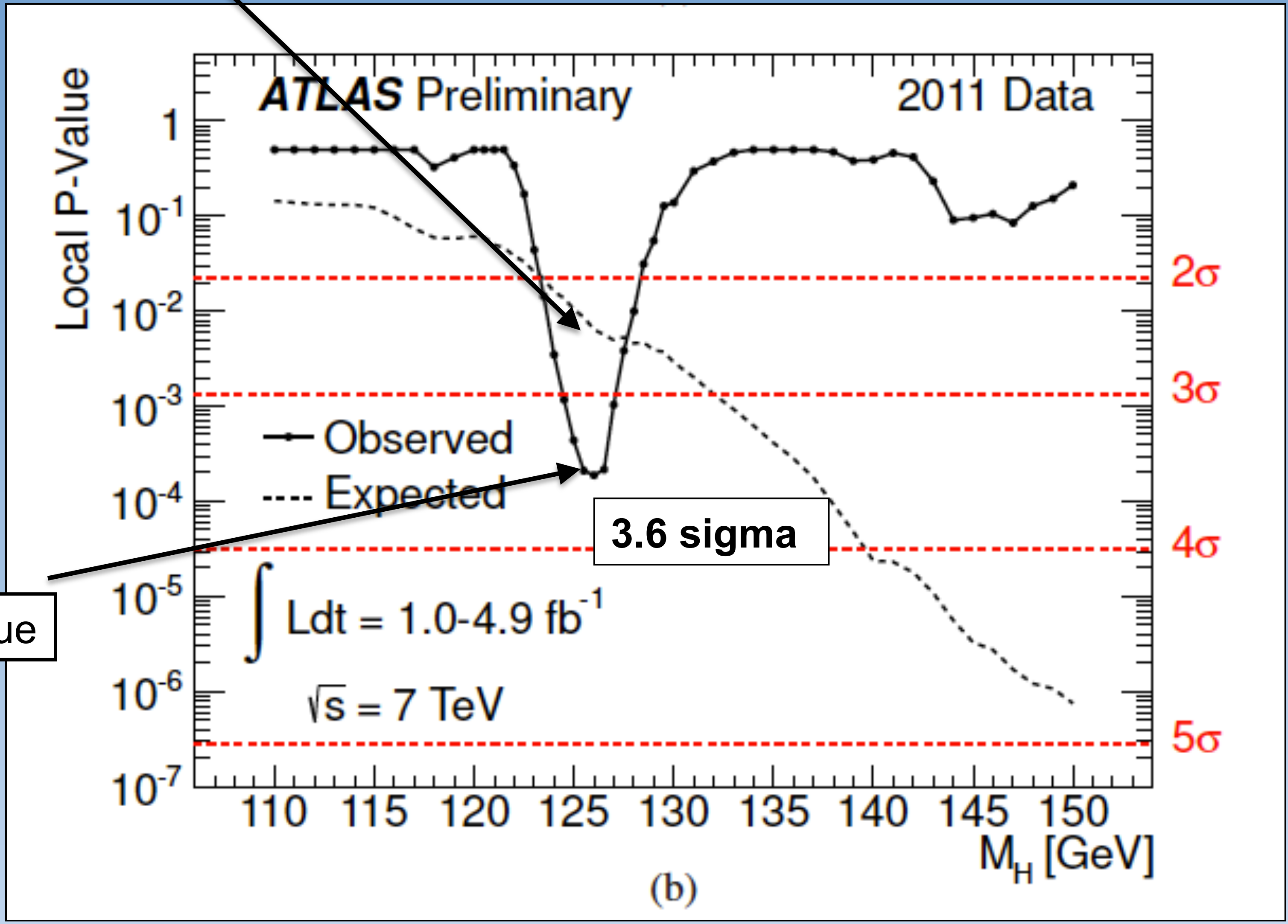
SM	30
Higgs	15



Discovery if p-value < 2.87×10^{-7}

Understanding the official ATLAS plot

exected p-value



(b)

3.6 sigma

observed p-value

Interpretation

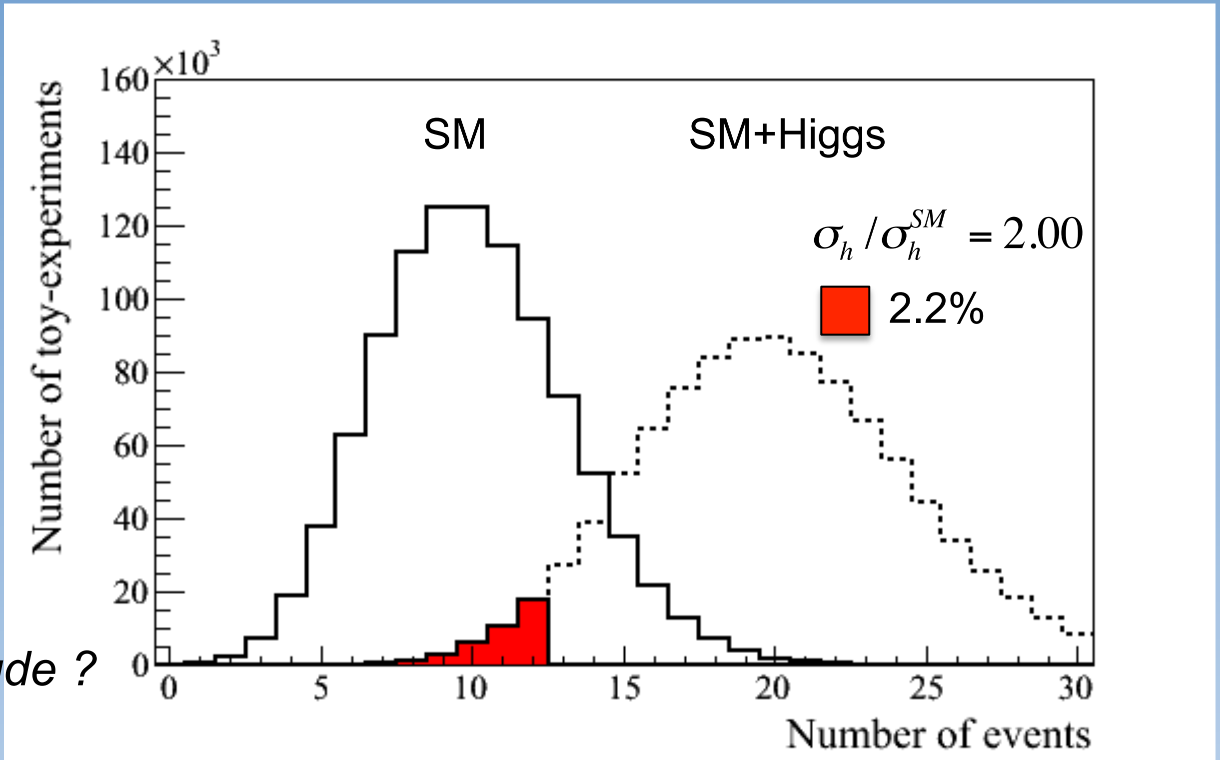
pessimistic: exclusion

When / how do you exclude a signal

Incompatibility with s+b hypothesis

Standard Model

SM	10
Higgs	5
Data	12



Can we exclude the SM+Higgs hypothesis ?

What σ_h/σ_h^{SM} can we exclude ?

Exclusion: probability to observe N events (or even less) under the signal + background hypothesis

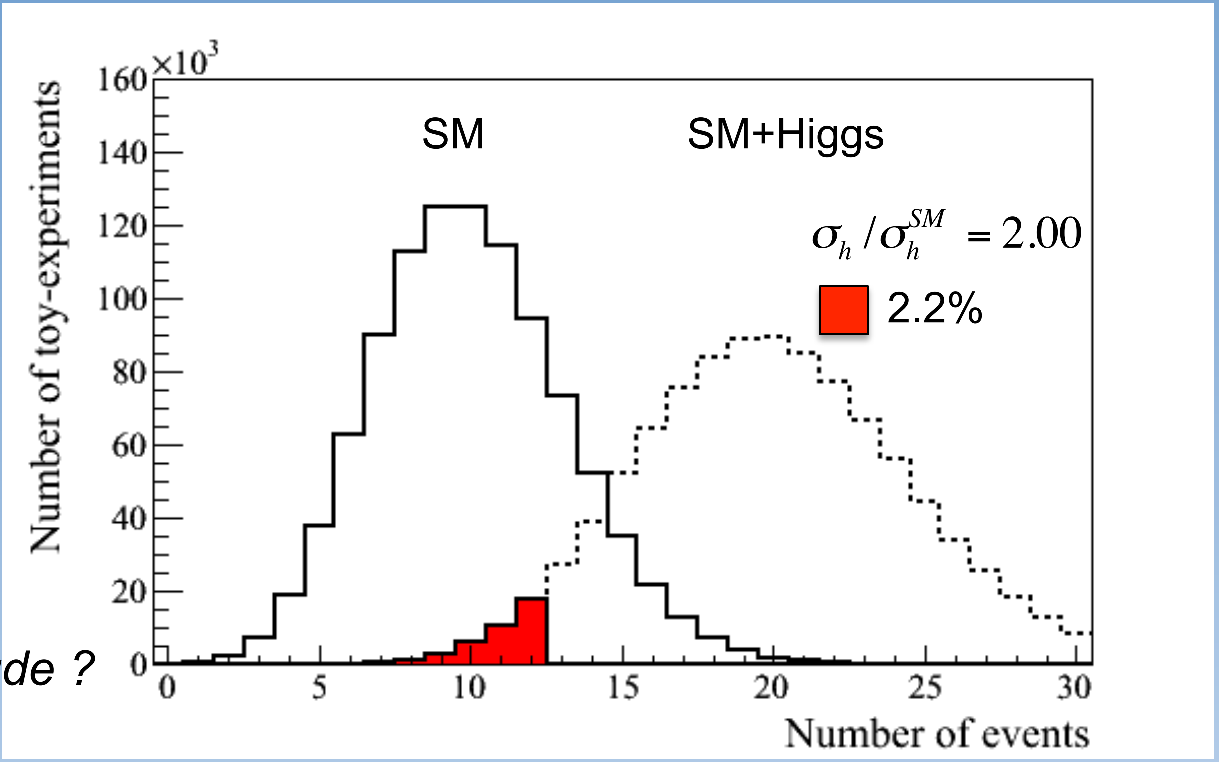
When / how do you exclude a signal

Standard Model

SM	10
Higgs	5
Data	12

Can we exclude the SM+Higgs hypothesis ?

What σ_h/σ_h^{SM} can we exclude ?



σ/σ_{SM}	SM	# data	SM+Higgs	
1.0	10	12	15.0	18.5 %
1.5	10	12	17.5	6.8%
2.0	10	12	20.0	2.2%

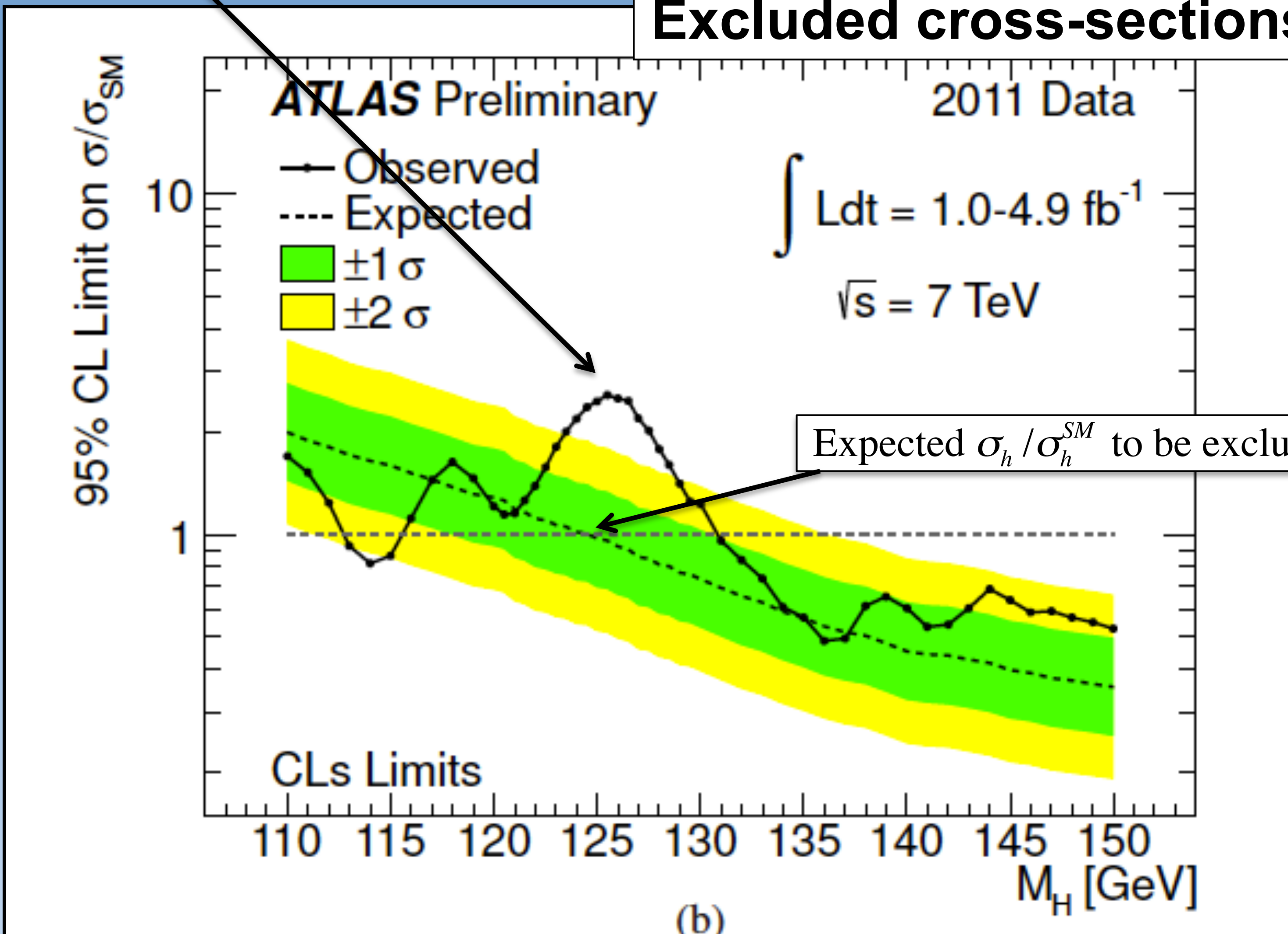
excluded

Expected exclusion ? Use mean SM instead of Ndata

Observed excluded cross-section, σ_h/σ_h^{SM} , = 1.64

Excluded cross-sections

Observed σ_h / σ_h^{SM} to be excluded



Expected σ_h / σ_h^{SM} to be excluded

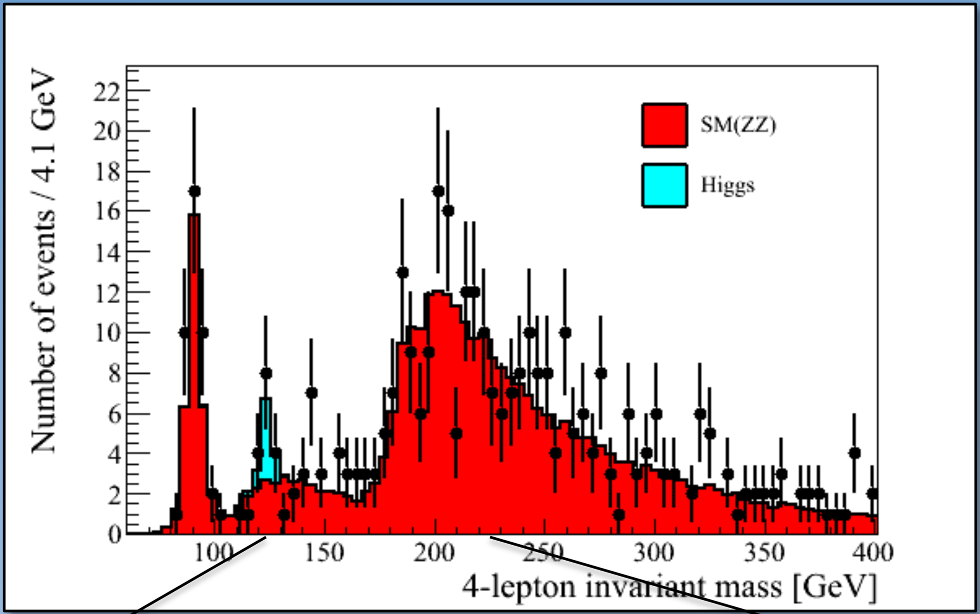
We will be optimistic today … discovery

Create the 4-lepton mass plot

```
root> .L Hasco_skeleton.C++
root> MassPlot(20)
```

↓ Rebin-factor

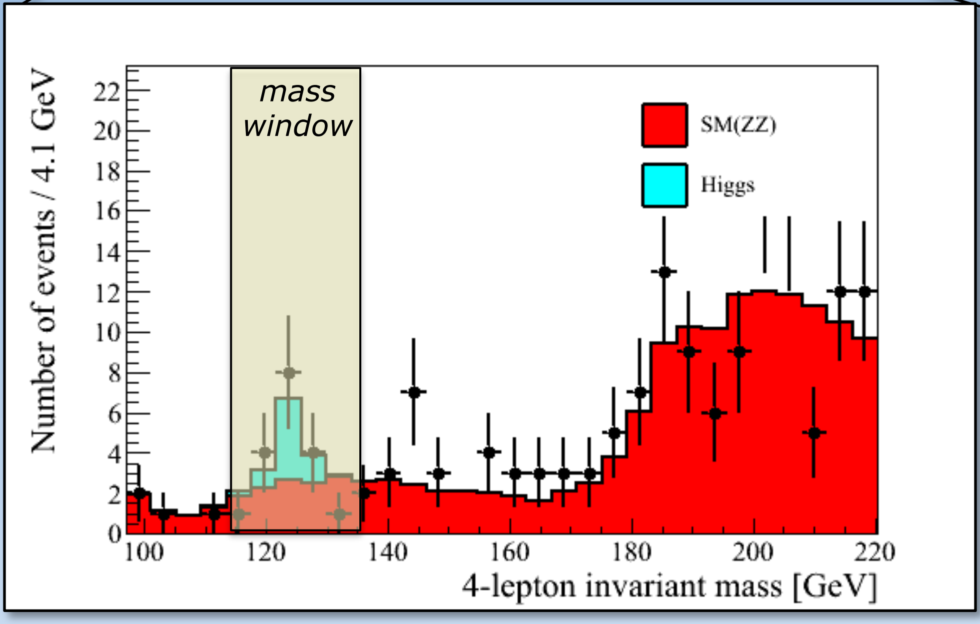
hist: h_bgr, h_sig, h_data



Summary in signal mass region
(using 200 MeV bins and 10 GeV window)

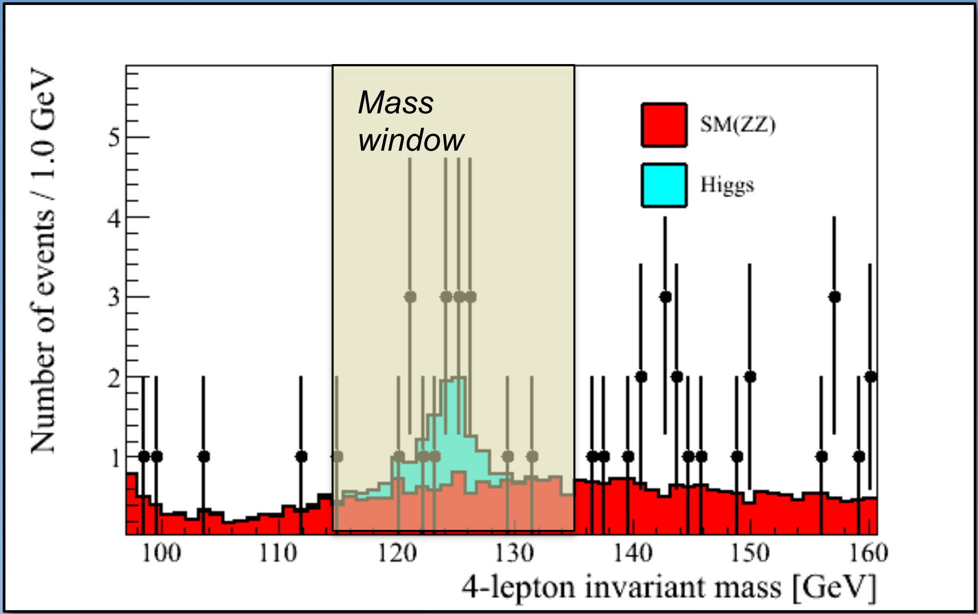
Ndata = 16
Nbgr = 6.42
Nsig = 5.96

Exercises: significance



Exercise 3

Optimizing the counting experiment



Code you could use:

```
IntegratePoissonFromRight()

Significance_Optimization()
```

Exercise 1: significance optimization of search window (Poisson counting)

3.1 Find the window that optimizes the expected significance

3.2 Find the window that optimizes the observed significance (never ever do this again)

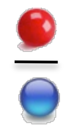
3.3 Find the window that optimizes the expected significance for 5x higher luminosity

3.4 At what luminosity do you expect to be able to make a discovery ?

More complex test statistics: beyond simple counting

Likelihood ratio:

$$X = -2\ln(Q), \text{ with } Q = \frac{L(\mu_s = 1)}{L(\mu_s = 0)} =$$



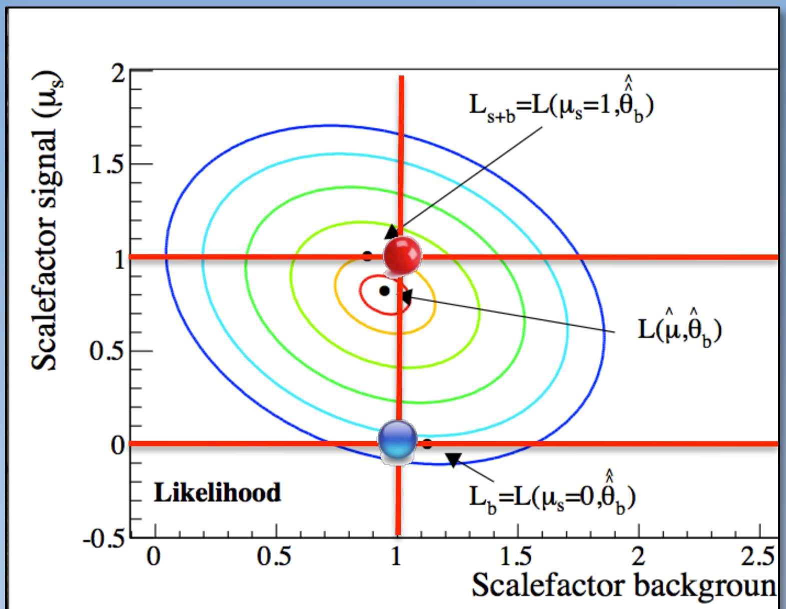
Tevatron-style:

$$X = -2\ln(Q), \text{ with } Q = \frac{L(\mu_s = 1, \hat{\hat{\theta}}_{(\mu_s=1)})}{L(\mu_s = 0, \hat{\hat{\theta}}_{(\mu_s=0)})}$$

LHC experiments:

$$X(\mu) = -2\ln(Q(\mu)), \text{ with } Q(\mu) = \frac{L(\mu, \hat{\hat{\theta}}(\mu))}{L(\hat{\mu}, \hat{\theta})}$$

2-dimensional fit (α and μ free)



Note:
α_{bgr} is just one of the nuissance parameters θ in a ‘real’ analysis

Let's stick to counting but make it a bit more realistic

In real life:

- you do not know the background level with absolute precision
- What is the best estimate and what is it's uncertainty ?

→ and how does it change your sensitivity ?

Statistics

1) Measurement:
- Likelihood fit

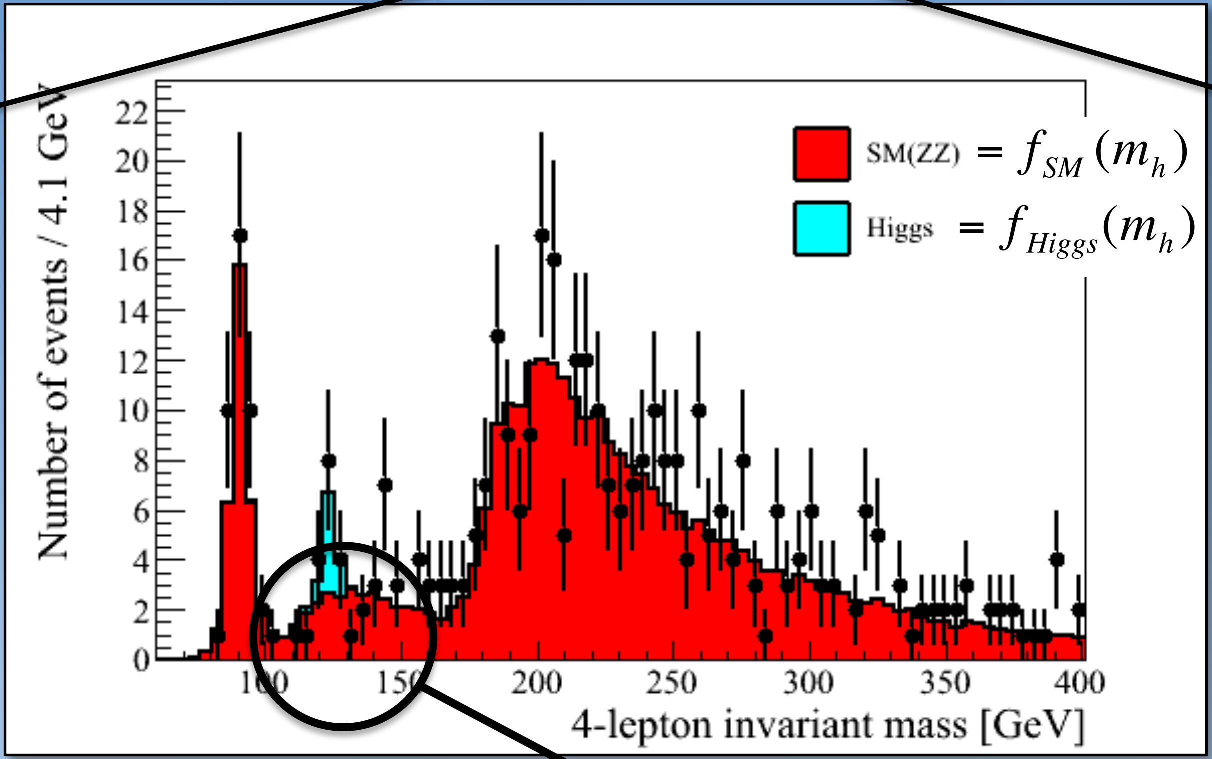
2) Hypotheses testing & search:
- Significance (definition and optimization)

- Test-statistic & pseudo-experiments (toy-MC)
- Exclusion/discovery

Marumi: $f(m_h) = \mu \times f_{Higgs}(m_h) + \alpha \times f_{SM}(m_h)$

Scale factor of the Higgs

Scale factor for background



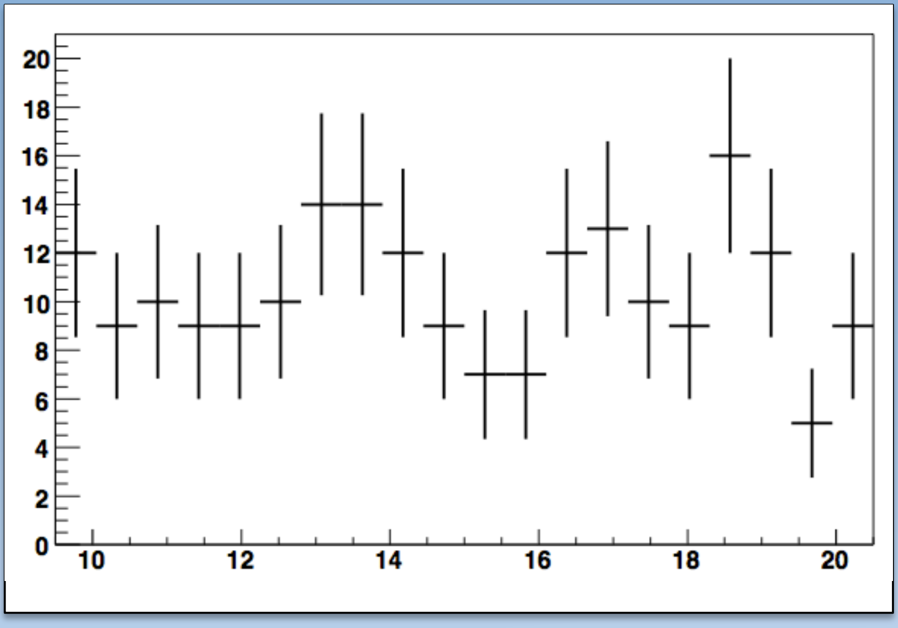
Goal: best background estimate: $N_{SM} \pm \Delta_{NSM}$?

1) Estimate $\alpha \pm \Delta\alpha$ in a background dominated region (mh>200)
2) Correct the MC estimate in signal region (from α)
3) Compute uncertainty on the background level (from $\Delta\alpha$)

Exercise 0

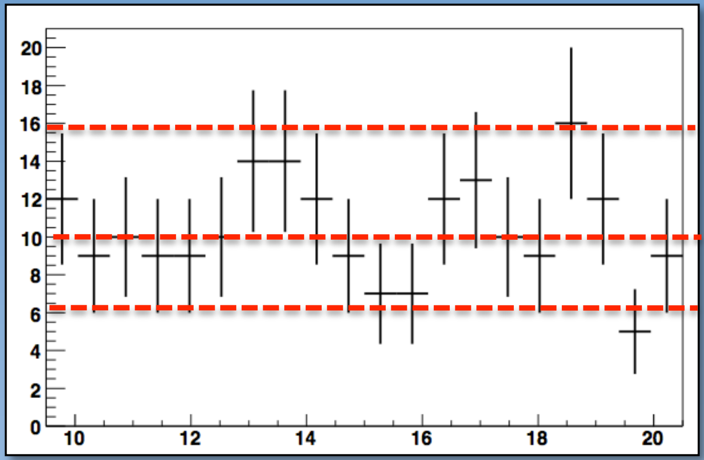
Produce a histogram on the screen

Exercise 0: reproduce this histogram on your screen, compute mean and create a gif file. Add text if you like.



Note: Look at the example macro’s: Histogram_Example_0*.C (*=0,1,2,3,4,5)

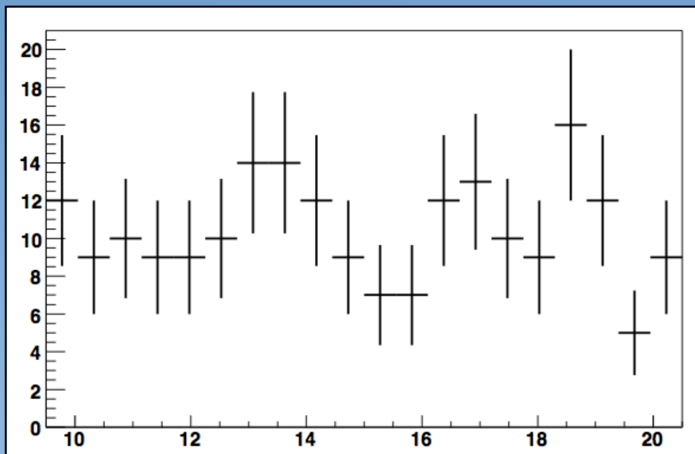
Fitting in 1 slide



You model: $f(x) = \lambda$

Try different values of λ
and for each one compute:

Fitting in 1 slide



You model: $f(x) = \lambda$

Try different values of λ and for each one compute the ***compatibility*** of the model with the data

χ^2-fit

Compatibility number :

$$\chi^2 = \sum_{bins} \frac{\left(N_{bin}^{data} - \lambda_{bin}^{expected}\right)^2}{N_{bin}^{data}}$$

Best value:
Value of λ that minimizes χ^2 (χ_{min}^2)

Errors:
Values of λ for which $\chi^2 = \chi_{min}^2 + 1$

Likelihood-fit

Compatibility number :

$$-2\log(L) = -2 \cdot \sum_{bins} \log(Poisson(N_{bin}^{data} \mid \lambda))$$

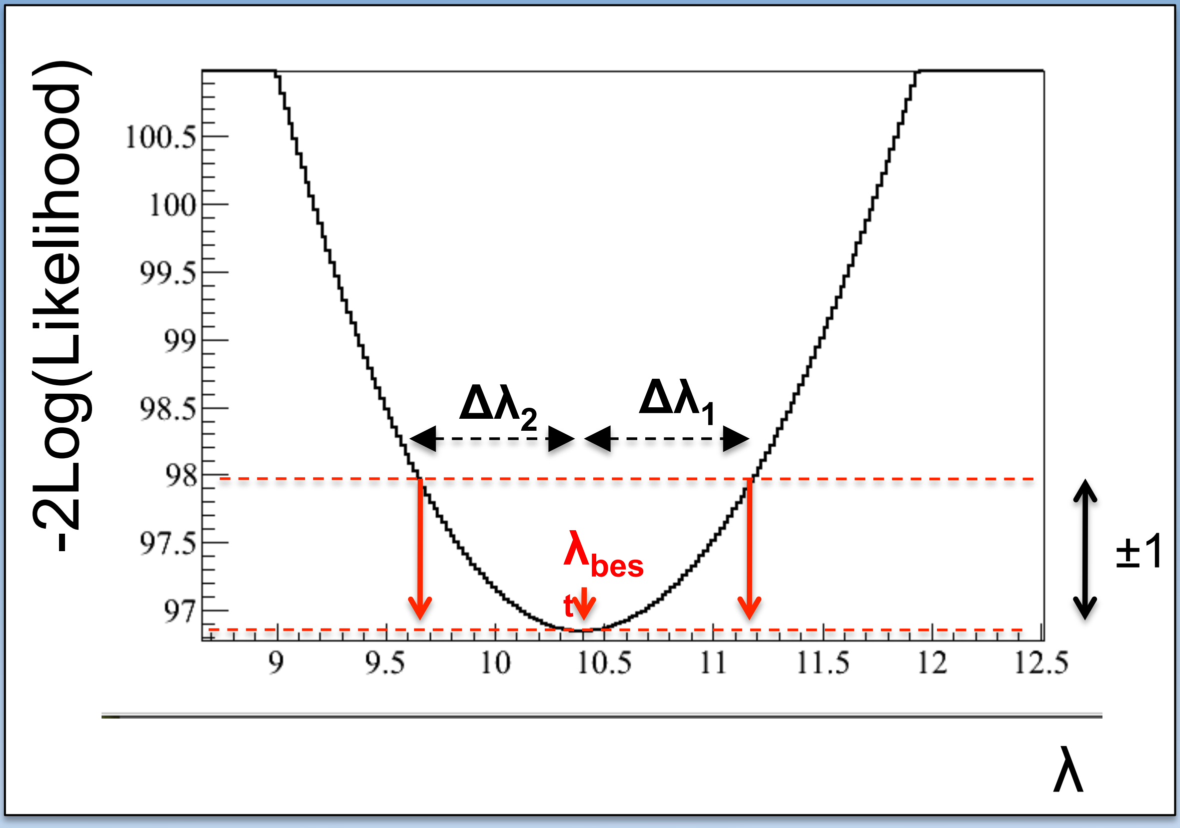
TMath::Poisson(Nevt_bin, λ)

Best value:
Value of λ that minimizes -2Log(L) ($-2\log(L)_{min}$)

Errors:
Values of λ for which 2Log(L) = ($-2\log(L)_{min}$) +1

Result from the fit

$$\text{result}: \quad \lambda = \lambda_{\text{best}}{}^{+\Delta\lambda_1}_{-\Delta\lambda_2}$$



Link to Lecture or d’Agostini on Monday

Probability $(data\|\lambda)$	$\neq$	Probability$(\lambda\|data)$
What we have computed *Likelihood*		What we want

The Poisson distribution

Binomial with n→∞, p →0 and np=λ

$$P(n \mid \lambda) = \frac{\lambda^n e^{-\lambda}}{n!}$$

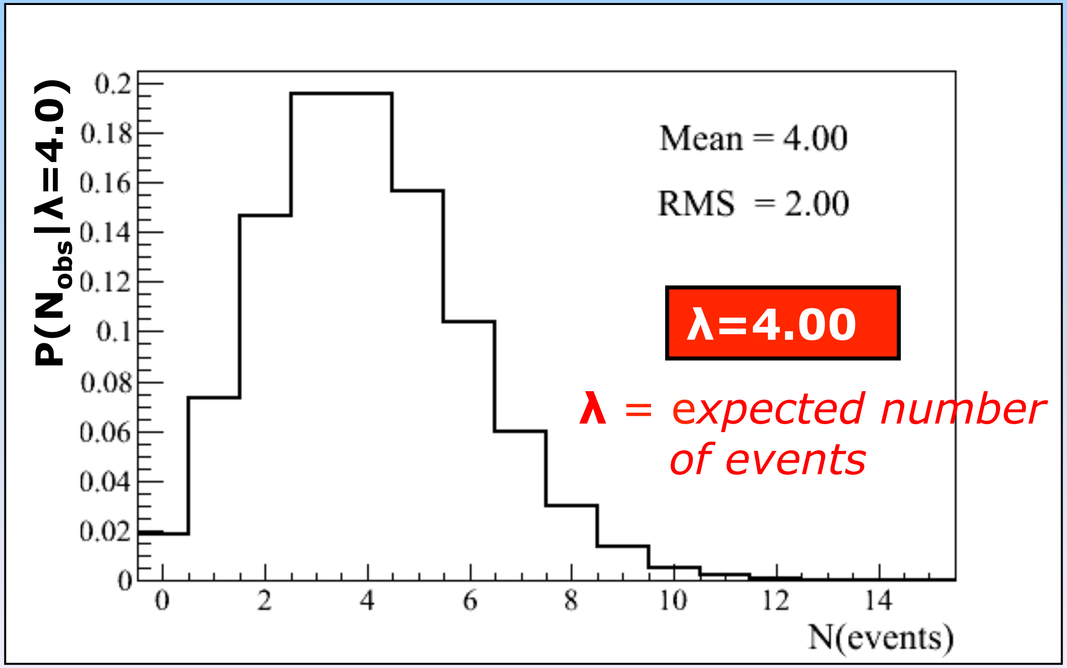
Probability to observe n events when λ are expected

Poisson distribution

$$P(0 \mid 4.0) = 0.01832$$
$$P(2 \mid 4.0) = 0.14653 \quad !$$
$$P(3 \mid 4.0) = 0.19537$$
$$P(4 \mid 4.0) = 0.19537$$
$$P(6 \mid 4.0) = 0.10420 \quad !$$

#observed — **varying**

λ hypothesis — **fixed**



λ = *expected number of events*

Ivo van Vulpen

Known λ (Poisson)

Binomial with n→∞, p →0 np=λ

$$P(n\,|\,\lambda) = \frac{\lambda^n e^{-\lambda}}{n!}$$

Poisson distribution

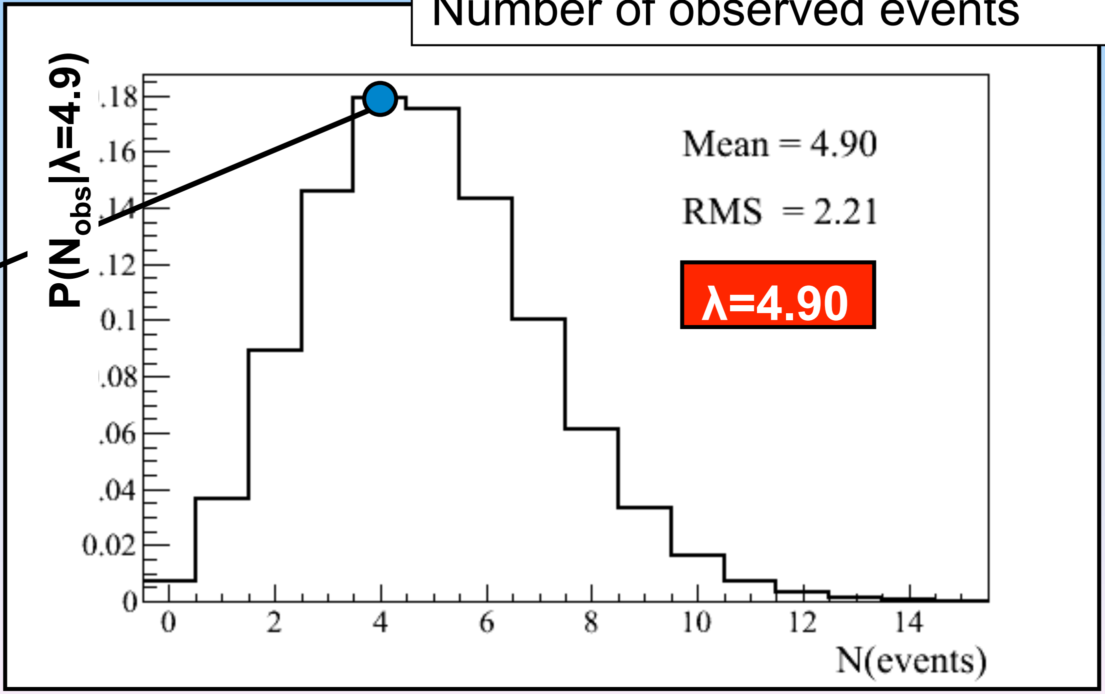
Probability to observe n events when λ are expected

$$P(0\,|\,4.9) = 0.00745$$
$$P(2\,|\,4.9) = 0.08940$$
$$P(3\,|\,4.9) = 0.14601$$
$$P(4\,|\,4.9) = 0.17887$$

#observed — **varying**

λ hypothesis — **fixed**

Number of observed events



the famous √N

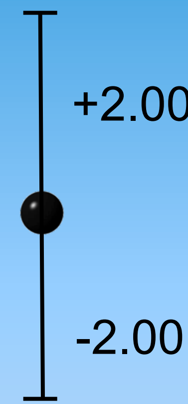
properties

(1) Mean: $\langle n \rangle = \lambda$

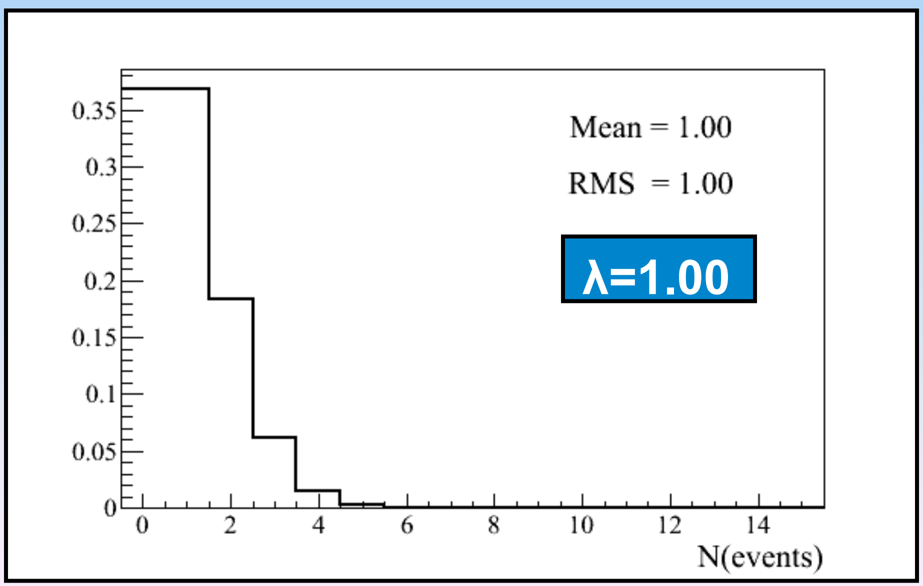
(2) Variance: $\langle (n - \langle n \rangle)^2 \rangle = \lambda$

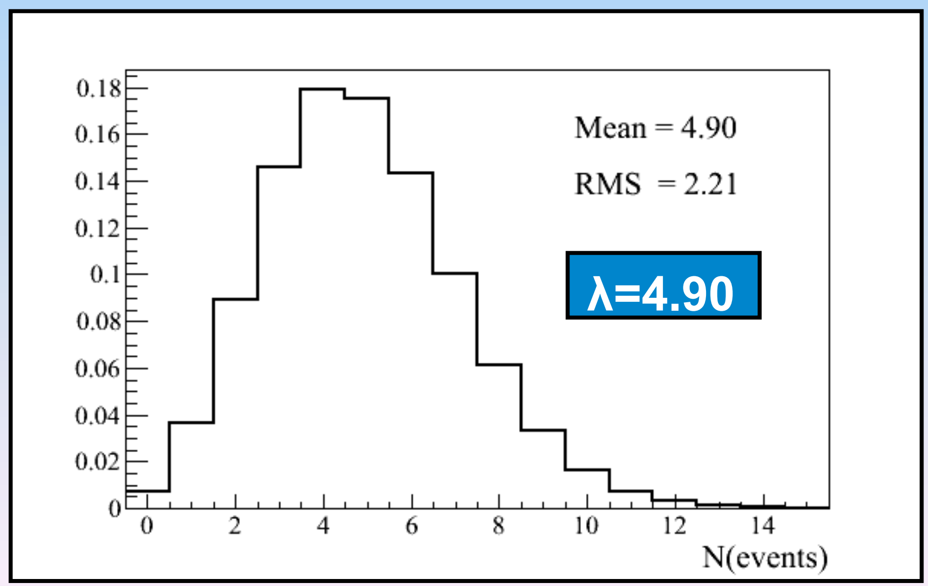
(3) Most likely: first integer ≤ λ

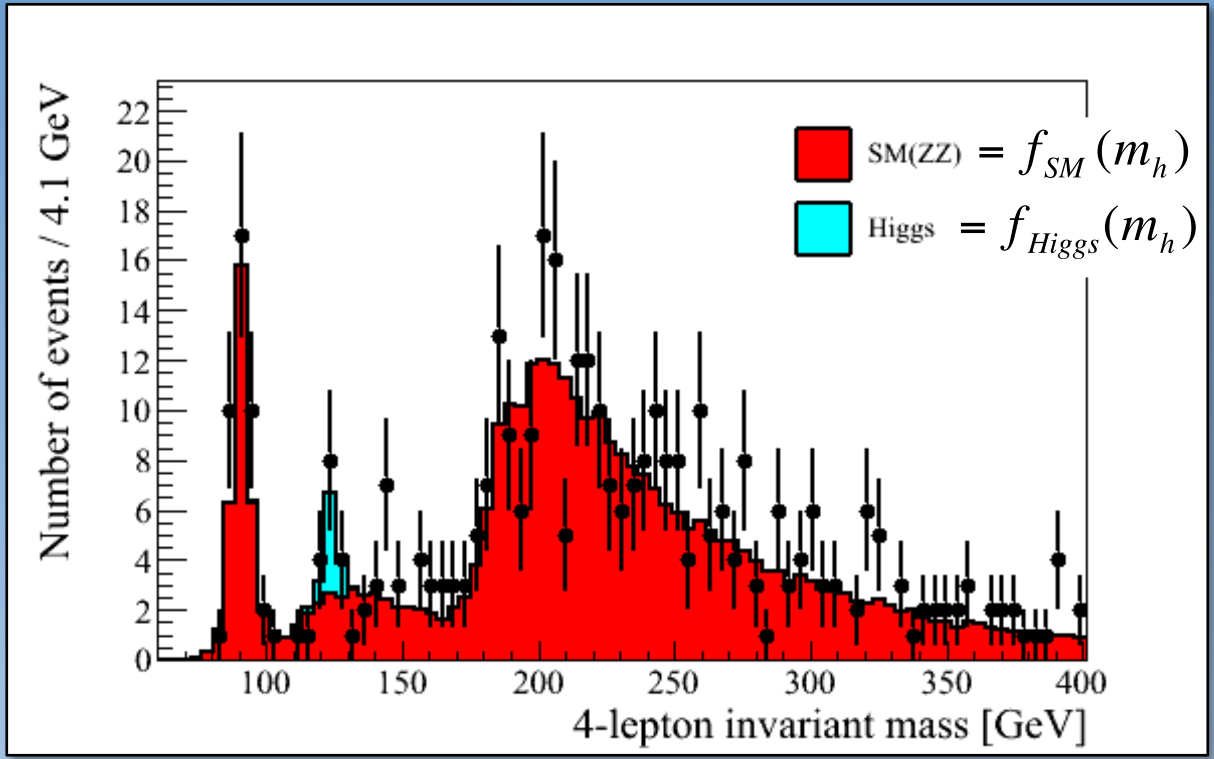
Usual way to represent the error on a data-point



Not default in Root







$$f(m_h) = \mu \times f_{Higgs}(m_h) + \alpha \times f_{SM}(m_h)$$

Scale factor for the Higgs

EXERCISE 2

Scale factor for the SM background

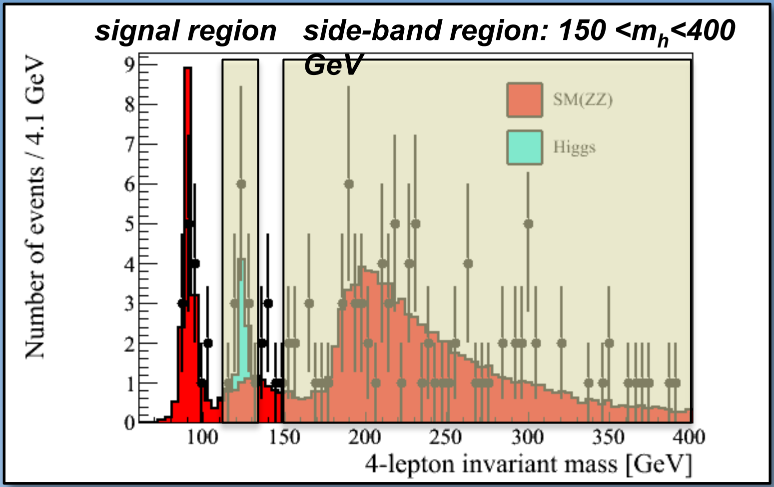
EXERCISE 1

Exercise 1

Data-driven bckground estimate in a 10 GeV mass window around 125 GeV

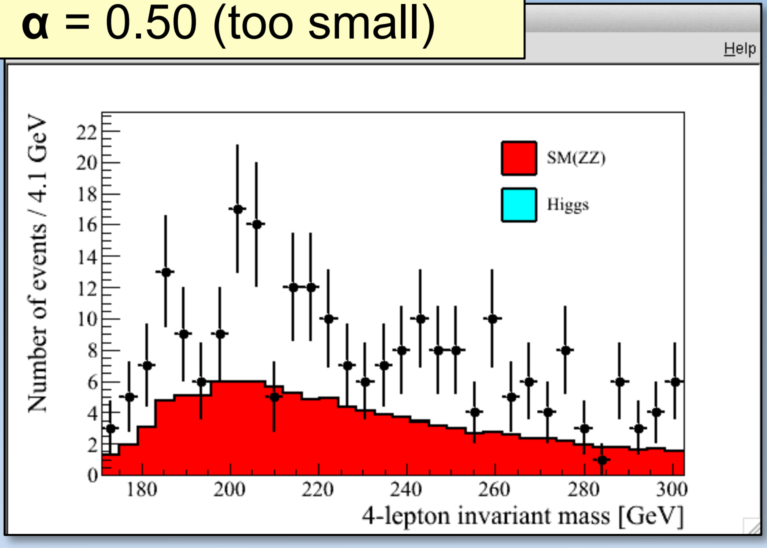
Code you could use:

```
SideBandFit()
```

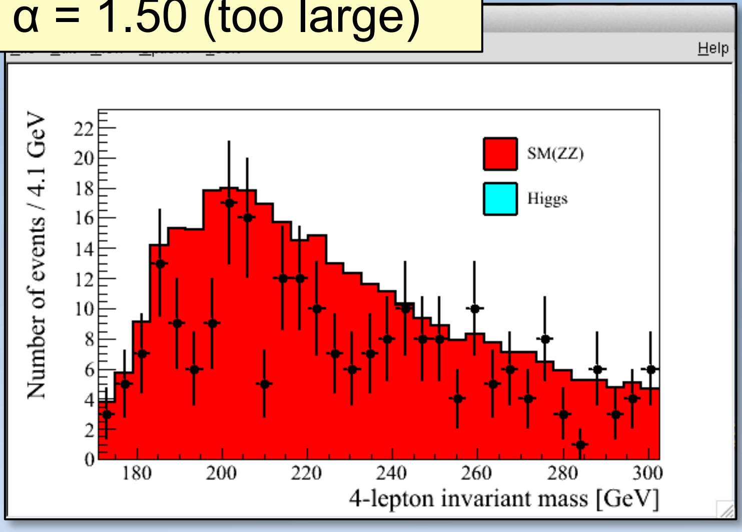


Exercise 1: determine the best estimate for the background scale factor (α) by doing a fit in the side-band region $175 \leq m_h \leq 300$ GeV (later $150 \leq m_h \leq 400$ GeV)

α = 0.50 (too small)



α = 1.50 (too large)

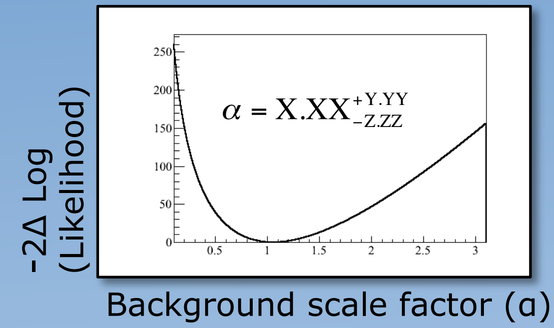


Exercise 1: determine the best estimate for the background scale factor factor (α) using a fit in a side-band region $175 \leq m_h \leq 300$ GeV

1.1 Find the best value α for using a **likelihood fit**

Computing the likelihood:
For each 'guess' of α:

$$-2\log(L) = -2 \cdot \sum_{bins} \log(Poisson(N_{bin}^{data} \mid \alpha \cdot f_{bin}^{SM}))$$



1.2 Find the best value α for using a **χ^2 fit**

$$\chi^2 = \sum_{bins} \frac{\left(N_{bin}^{data} - \lambda_{bin}^{expected}\right)^2}{N_{bin}^{data}}$$

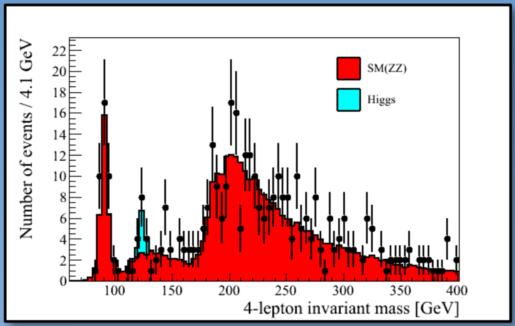
Wikipedia: Carl Friedrich Gauss is credited with developing the fundamentals of the basis for least-squares in 1795 at the age of 18. Legendre was the first to publish the method however.

1.3 Discuss the differences between the two estimates

1.4 Redo 1.1 and 1.2 with $150 \leq m_h \leq 400$ GeV. What happens ?

1.5 Use the likelihood fit, fine binning and $150 \leq m_h \leq 400$ GeV to determine the best estimate for α ($\alpha_{best} \pm \Delta\alpha$). Estimate the bckg level ($b \pm \Delta b$) in the signal region: $120 \leq m_h \leq 130$ GeV

Exercise 3



Exercise 3.5: How does the uncertainty on the background change the observed and expected significance ?

Use toy-Monte Carlo experiments (numbers or distributions)

Basic material for the Root examples:

1) Download tarball: **HascoRootStatisticsCode.tgz**

2) Unpack everything: **tar –vzxf HascoRootStatisticsCode.tgz**

Directory /RootExamples/

a) Example0*.C (* = 0,1,2,3,4,5) All *.C-files in this presentation
b) Code for Ntuple production and reading
c) rootlogon.C some standard Root settings

Directory /Exercises/

a) Histograms_fake.root
4 histograms of the 4-lepton invariant mass (H125, H200, ZZ, data)
b) Hasco_skeleton.C
skeleton code (different levels, as minimal as possible). Your code !
c) rootlogon.C
some standard Root settings

Exercises


Good luck
Questions/remarks:
Ivo.van.Vulpen@nikhef.nl

BACKUP

Exercise 4

compute test-statistic X

$$X = -2\ln(Q), \text{ with } Q = \frac{L(\mu_s = 1)}{L(\mu_s = 0)}$$

→ Likelihood assuming μ_s=1 (signal+background)

→ Likelihood assuming μ_s=0 (only background)

Exercise 4: create the likelihood ration test statistic – beyond simple counting

4.1 Write a routine that computes the likelihood ratio test-statistic for a given data-set

double Get_TestStatistic(TH1D *h_mass_dataset, TH1D *h_template_bgr, TH1D *h_template_sig)

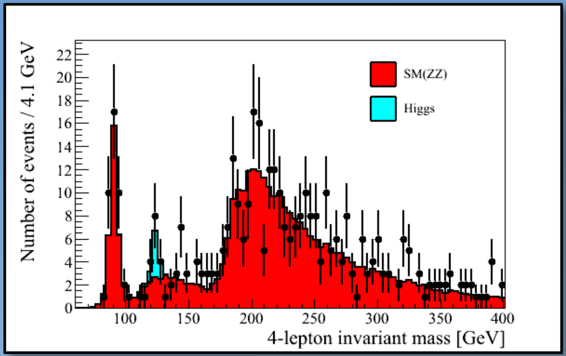
$$-2Log\left(Likelihood_{(\mu,\alpha = 1)}\right) = -2 \cdot \sum_{bins} \log\left(Poisson(N_{bin}^{data} \mid \mu \cdot f_{bin}^{Higgs} + \alpha \cdot f_{bin}^{SM})\right)$$

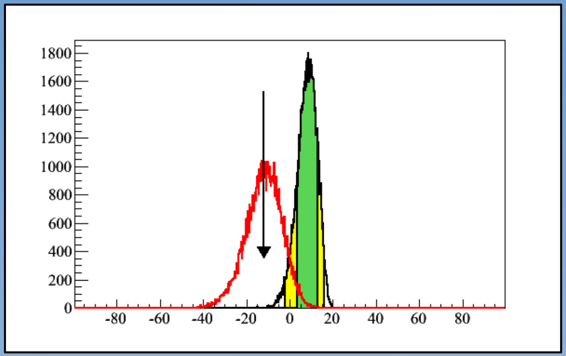
Note: log(a/b) = log(a) – log(b)

4.2 Compute the likelihood ratio test-statistic for the 'real' data

Exercise 5

Toy data-sets





Exercise 5: create toy data-sets

5.1 Write a routine that generates a toy data-set from a MC template (b or s+b)

TH1D * GenerateToyDataSet(TH1D *h_mass_template)

How: Take the histogram h_mass_template and draw a Poisson random number in each bin using the bin content in h_mass_template as the central value. Return the new fake data-set.

5.2 Generate 1000 toy data-sets for *background-only* & compute test statistic
Generate 1000 toy data-sets for *signal+background* & compute test statistic
→ plot both in one plot

5.3 Add the test-statistic from the data(exercise 4.2) to the plot

Exercise 6

Summarize separation power

Exercise 6: compute p-value

6.1 Compute the p-value or 1-Cl_b (under the background-only hypothesis):

- For the average(median) b-only experiment
- For the average(median) s+b-only experiment [expected significance]
- For the data [observed significance]

6.2 Draw conclusions:

- Can you claim a discovery ?
- Did you expect to make a discovery ?
- At what luminosity do you expect to be able to make a discovery ?

Exercise 7:

Exclude a cross-section for a given Higgs boson mass

Some shortcomings,
but we'll use it anyway

$$\sigma_h(m_h) = \zeta \cdot \sigma_h^{SM}(m_h)$$

ζ: scale factor wrt SM prediction

Exercise 7: compute CL_{s+b} and exclude Higgs masses or cross-sections

7.1 Compute the CL_{s+b}:
- For the average(median) s+b experiment
- For the average(median) b-only experiment
- For the data

7.2 Draw conclusions:
- Can you exclude m_h=200 GeV hypothesis ? What ς can you exclude ?
- Did you expect to be able to exclude the m_h=200 GeV hypothesis ? What ς did you expect to be able to exclude ?