Upper limit for High mass X→WW search

Piergiulio Lenzi

May 25, 2018

Abstract

1 Introduction

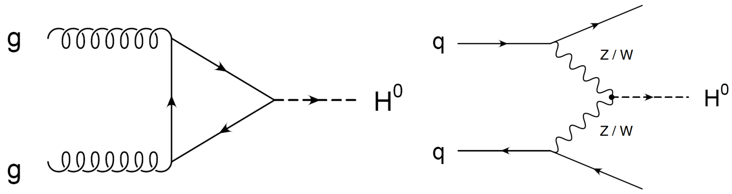
In this report we will study the statistical approach to the upper limit computation for a high energy physics search for a new particle. We will use data from the 2015 run of the CMS experiment at the LHC and the corresponding Monte Carlo (MC) simulations.

We will search for a new signal in the $WW \rightarrow 2l2\nu$ final state. This channel is also a decay channel for the Standard Model Higgs boson (H). We will however search for an additional particle, named X in the following, that is supposed to be heavier than H. We will scan a wide range of masses for X and we will set an upper limit on the cross section of X, as well as compute the significance of any excess we may observe.

2 The physics case

In this exercise we will search for the existence of a new hypothetical high mass particle, named X. **We expect this particle to be a heavy variant of the Standard Model Higgs boson**. For this reason we hypothesize that X shares with H the production mechanism. This is a reasonable assumption that is verified in several new physics models. In particular we assume that X can be produced via two main production mechanisms, called gluon fusion (ggF) and vector boson fusion (VBF), represented by the diagrams in Fig. 1.

The details and precise meaning of the Feynman diagrams of Fig. 1 are not relevant for this exercise, the relevant piece of information is that two



(a) ggF (b) VBF

Figure 1: Feynmann diagrams of the two main production mechanisms for X.

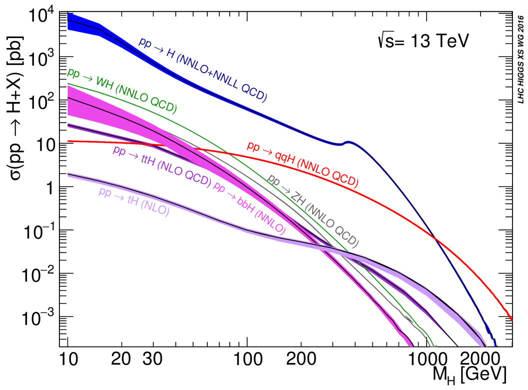


Figure 2: Standard model Higgs boson cross section as a function of the mass of the Higgs boson. The ggF mechanisms is labeled $pp \to H$, the VBF mechanism is labeled $pp \to qqH$. Also other, sub-leading mechanisms are reported, which we will neglect.

mechanisms are available and that they are marked by a substantial difference: **the ggF shows no other particles in the final state beyond X, the VBF has two quarks in addition to the X in the final state**. Although it should be noted that a precise calculation shows that additional particles, in the form of hadronic jets, can also arise in ggF, it remains true that events arising from the two mechanisms are different when it comes to the number of jets produced in addition to the X particle: most of ggF events have no high pT jets, while most of VBF events have two well separated high pT jets in the final state.

The production cross section for the Higgs boson as a function of its mass is reported in Fig. 2. Although we now know the mass of the Higgs boson to be 125 GeV, this plot is useful because it can be used as a model for the expected cross section for X.

We will assume that the cross section σ for each of the production channels of X scales with a common factor μ of the corresponding Higgs boson cross

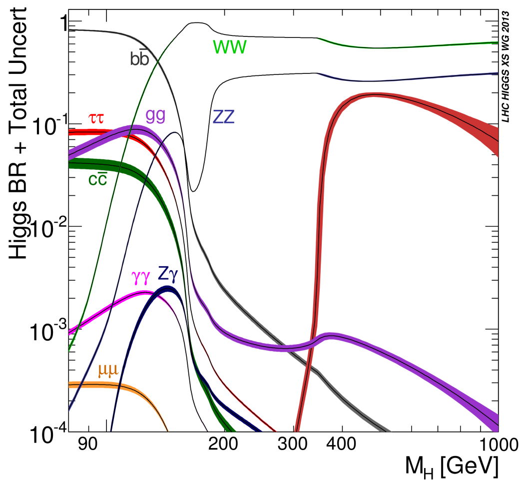


Figure 3: Standard model Higgs boson cross section as a function of the mass of the Higgs boson. The ggF mechanisms is labeled $pp \to H$, the VBF mechanism is labeled $pp \to qqH$. Also other, sub-leading mechanisms are reported, but we will neglect them.

section:

$$\sigma_{X[ggF,VBF]}(M) = \mu(M)\sigma_{H[ggF,VBF]}(M) \tag{1}$$

where M is the mass of X, and obvious meaning of the other symbols. This is a reasonable assumption, verified by several new physics models. We notice that **the relative importance of the VBF mechansm grows with the X mass**, and becomes dominant above ∼1.5 TeV.

We will assume that, like H, also X has several decay channels. We assume that X has the same branching ratios of H, which are summarized in Fig. 3. It should be noted that **the WW decay channel is the one with the highest branching ratio**.

3 Data sample and analysis strategy

Owing to the large branching fraction in the WW final state we choose this channel for our search. In other words we search for the decay X→WW. The W bosons are themselves unstable. 30% of the times a W boson decays to a charged lepton (electron, muon, tauon) and a neutrino (10 % for each of the three lepton species). The remaining 70% of the times the W boson decays to a pair of hadrons. In this exercise we will concentrate on the leptonic decays of the W bosons. The reason of this choice will become clearer when we discuss backgrounds, but let us mention already that requiring leptons in the final state allows a dramatic reduction of background processes. To summarise, **we search for the X→WW→ $2l2\nu$ decay chain**.

We will use data collected by the CMS experiment in the 2015 run of the at LHC. These data correspond to an integrated luminosity of 2.3 fb^{-1}. We will base our analysis on data reconstructed with the standard CMS software and simulations for both the signal and the background processes. These data come in the form of ROOT trees.

For each event the tree stores several event variables, in particular:

- reconstructed leptons kinematic variables;
- the reconstructed missing transverse energy;
- the reconstructed jet kinematic variables.
- several weights, used both to improve the Data/Simulation agreement and to normalize the simulation to the data luminosity.

3.1 Main backgrounds

A background process in a high energy physics analysis is a physics process that yields a final state that resembles that of the signal. In the case of this analysis there are two main background processes:

- production of two W bosons without an intermediate X;
- production of a pair of top quarks, $t\bar{t}$. The top quark decays to a b quark and a W boson.

The Feynman diagrams of these two processes are reported in Fig. 4 for the interested reader.

We apply a series of cuts to reduce their contribution as much as possible. $t\bar{t}$ is reduced by requiring that jets in the event are not compatible with being originated from b quarks, using dedicated jet-tagging techniques. WW is reduced with kinematic cuts on the leptons.

Other subleading sources of background originate from processes in which at least one of the leptons is not a real lepton, but is identified as such by the reconstruction algorithms. The control of these backgrounds is a crucial part of the analysis, but is beyond the scope of this exercise. The cuts to be applied will be provided by the teachers. These cuts are implemented in the provided file `HWWYields.C`.

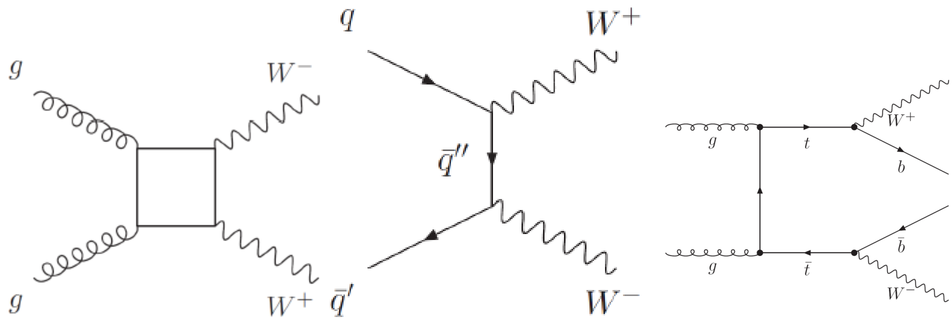


Figure 4: Gluon initiated (a) and quark initiated (b) WW. Top pair production (c).

3.2 Main discriminating variable

After the selection cuts mentioned above we end up with a sample that is primarily composed of WW and $t\bar{t}$. In order to further discriminate a possible signal we use a kinematic variable called $m_{T,i}$. This variable is the invariant mass of the 4-momentum resulting from the sum of the two lepton 4-momenta and the MET 4-momentum. Since we are unable to reconstruct the longitudinal component of the neutrinos momenta, this variable is the closest approximation of the resonance invariant mass that we can reconstruct in a signal event, and it retains a significant discriminating power with respect to backgrounds.

The distribution of $m_{T,i}$ is shown in Fig. 5 after the selection cuts. The data point are show on top of the stack of the backgrounds. The shape of a signal for X mass of 1 TeV is also shown (multiplied by 10). The rightmost bin of each distribution is an overflow bin. The fact that the signal shape does not peak at 1 TeV is due to $m_{T,i}$ lacking the contribution from the longitudinal momenta of neutrinos.

Exercise 1:

Write a code that builds the stack of backgrounds to obtain a plot similar to Fig. 5.

A set of root files is provided, together with a root macro (`HWWYields.C`) implementing selection cuts.

Each root file contains a root tree named `latino`. This tree holds several variables, including $m_{T,i}$ in a branch called `mTi`.

Eight data files are provided, corresponding to two distinct data taking periods (called `Run2015C` and `Run2015D`) and four triggers. Data from CMS are in fact divided into different streams depending on the triggers each event fires. In this analysis we use events with at least two electrons (`DoubleEG`), at least two muons (`DoubleMu`), at least an electron and a muon (`MuonEG`) or at least a muon (`SingleMuon`). The latter is used only to recover an inefficiency in the other triggers.

Simulated events (MC) are produced for several physics processes, including $t\bar{t}$, WW, Z$\rightarrow \tau\tau$, and other subleading processes. Simulated samples for different mass hypotheses for X are also provided. For each mass hypothesis two different files are provided, one containing the simulation of the ggF production mechanism and the other containing the simulation for the VBF production mechanism.

Simulated events are weighted with several weights that are aimed at bringing data and MC in close agreement. We will not discuss these weight in detail. We will just discuss an important weight contained in the branch named `baseW`. This variable is used to equalize the luminosity of all samples to 1/fb.

Running `HWWYields.C` results in a root file (`yields.root`) containing one histogram for each background, one for each signal and one for the data. The variable plotted and the binning are controlled in the first few lines of `HWWYields.C`.

Please write a root macro that builds a stack of all the backgrounds, superimposes the data and one signal for reference.

3.3 Cut based analysis

In order to check whether the data are consistent with the signal+background hypothesis or with the background only hypothesis, we can proceed to count-

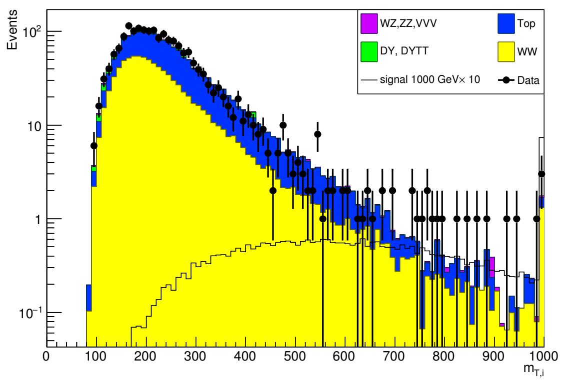


Figure 5: $m_{T,i}$ in data and simulation for 2015 data after selection cuts.

ing events passing our selection. Let N_{obs} be the data events after the selection, ν_b be the expected number of background evens and ν_s be the expected number of signal events for an X signal with mass M. We expect N_{obs} to follow the Poisson statistics with an average of $\nu_b + \nu_s$ in the signal+background hypothesis and ν_b in the background only hypothesis.

The best estimate that we can give of the number of signal events, based on a single experiment, is

$$\hat{\nu}_s = N_{obs} - \nu_b \tag{2}$$

In order to understand whether the obtained value of $\hat{\nu}_s$ is significantly different from 0 we should compare it with the statistical fluctuation of the expected background. Only if $\hat{\nu}_s$ is larger than several standard deviations (5) of the expected background we can claim a discovery.

Following this discussion we are left with an open question: should we add any other selection cuts to our selection to improve the analysis sensitivity to a signal?

Exercise 2

For each X mass hypothesis find the cut on $m_{T,i}$ which maximises the sensitivity

Fig. 5 shows that $m_{T,i}$ has a good discriminating power. In particular if we only count events above a certain optimal value of $m_{T,i}$ we should be able to improve the sensitivity to a signal.

The student should find, for each X mass hypothesis, the value of $m_{T,i}$ ($m_{T,i}^{cut}$) such that by integrating events with $m_{T,i} > m_{T,i}^{cut}$ the ratio of expected number of signal events ν_s and the statistical fluctuation of background events $\sqrt{\nu_b}$ ($\nu_s/\sqrt{\nu_b}$) is maximised.

4 Maximum likelihood estimator of the signal strength

In this paragraph we will introduce the maximum likelihood fit as a way to estimate the signal strength in the presence of systematic uncertainties. We spell out since the beginning that the systematic uncertainties will be included in the fitting procedure in the form of parameters for which an external constraint is imposed in the fit. Parameters in a maximum likelihood fit which represent systematic uncertainties are called **nuisance parameters**, to distinguish them from the parameters for which we do not put external constraints, such as the signal strength (μ) in our case, which are known as **parameters of interest (POI)**.

Let us assume that the random variable that we measure in our experiment is N_{obs}. We expect it to be Poisson-like distributed around $\nu_b + \mu \cdot \nu_s$, where ν_s is the number of signal events we expect for the signal, if X is actually the SM H (remember Eq. 1). In this case the likelihood function $\mathcal{L}(\mu)$ is simply:

$$\mathcal{L}(\mu) = \frac{(\nu_b + \mu \cdot \nu_s)^{N_{obs}}}{N_{obs}!} e^{-(\nu_b + \mu \cdot \nu_s)} \tag{3}$$

Maximizing $\mathcal{L}(\mu)$ with respect to μ gives the best fit value for μ, $\hat{\mu}$. The result is the already known result of Eq 2, which we can express in terms of μ as

$$\hat{\mu} = \frac{N_{obs} - \nu_b}{\nu_s}. \tag{4}$$

We can introduce our degree of belief in the knowledge of the background distribution in the form of a nuisance parameter with an external constraint in the likelihood $\mathcal{L}(\mu)$. Let us for example assume that we know the normalization of the background with a relative uncertainty δ_{ν_b}/ν_b, and let us introduce the nuisance parameter μ_b, which is a multiplier of ν_b in much the same way as μ is multiplier for the number of expected signal events.

In order to add a constraint on μ_b in the likelihood we simply have to multiply the likelihood of Eq. 3 by a Gaussian constraint on μ_b with standard deviation δ_{ν_b}/ν_b. The new likelihood function is now function of both μ (the POI) and μ_b (a nuisance parameter), and reads:

$$\mathcal{L}(\mu; \mu_b) = \frac{1}{2\pi \frac{\delta_{\nu_b}}{\nu_b}} e^{-\frac{(\mu_b - 1)^2}{2(\frac{\delta_{\nu_b}}{\nu_b})^2}} \times \frac{(\mu_b \cdot \nu_b + \mu \cdot \nu_s)^{N_{obs}}}{N_{obs}!} e^{-(\mu_b \cdot \nu_b + \mu \cdot \nu_s)}, \tag{5}$$

where the initial part before the $\times$ symbol is the Gaussian constraint on μ_b.

The reader might be surprised by the fact that by maximizing Eq. 5 we are effectively fitting two parameters (the POI μ and the nuisance μ_b) with a single measurement (N_{obs}). One should remember, however, that the second constraint effectively comes from the assumed shape of the distribution of the nuisance parameter, a Gaussian in this case.

The likelihood function of Eq. 5 can be easily extended to the case in which our measurement consists of a vector of n random variables $\vec{N}_{obs} = (N^1_{obs}...N^n_{obs})$. We would like to draw the reader's attention on the fact that this is exactly the case we have when we measure number of events in a binned histograms with n bins, such as in Fig. 5. To extend Eq. 5 to handle this case, one simply needs to introduce a product of Poisson distributions, one for each of the n bin.

Similarly Eq. 5 can be easily extended to the case in which we have more nuisances. For example, our background could be (and most likely will be) composed by several contributions (e.g. one for each different background process), each with their own normalization uncertainty. In this case one simply add each nuisance as a multiplicative constraint in the likelihood. Also, the functional form of each constraint could be different for the different constraints.

Although the formalism introduced in this paragraph might sound like an overkill for a simple measurement counting experiment such as the one we are considering in this experiment, this formalism allows for handling of arbitrary number of bins and arbitrary number of nuisances. Consider for example that the H→ WW analysis in CMS has around 100 bins, a variable number of POIs, ranging from 1 to 10 depending on the particular quantity that is measured, and several tens of nuisance parameters.

5 The profile likelihood (PL) statistics

Based on $\mathcal{L}(\mu;\mu_b)$ we can construct a **test statistic** (i.e. a function of the stocastic variables we measure) $\lambda(\mu)$, called profile likelihood ratio (PL) as forllows:

$$\lambda_\mu(N_{obs}) = \frac{\mathcal{L}(\mu;\hat{\hat{\mu}}_b)}{\mathcal{L}(\hat{\mu};\hat{\mu}_b)}. \tag{6}$$

Some explanation on the symbols is required: $\hat{\mu}$ **is the best fit of the signal strength** μ, our POI. Similarly, $\hat{\mu}_b$ **is the best fit of the nuisance** μ_b. In other words, given our measurement of N_{obs}, the ($\hat{\mu}$,$\hat{\mu}_b$) pair is the maximum of the likelihood 5. Suppose now that we choose a particular value of the POI, a value of our choice and we call it μ: then $\hat{\hat{\mu}}_b$ **is the value of the nuisance** μ_b **which minimizes the NLL function ?? for our chosen value of** μ.

The reason why we go through the burden of constructing $\lambda_\mu(N_{obs})$ is two-fold:

1. by the Neyman-Pearson lemma, this test statistics gives the best power to discriminate between the hypotheses μ and $\hat{\mu}$.

2. by the Wilks theorem, the quantity $-2log(\lambda_\mu)$ is asymptotically distributed as a χ^2 with 1 degree of freedom.

The fact that $-2log(\lambda_\mu)$ is distributed as a χ^2 with 1 degree of freedom allows for the determination of the confidence interval on μ with the graphyical method, finding the crossings at +1, and also make it easy to compute the p-value. Let us now see how we can use the PL, or small modifications of it, for the task of either measuring the significance of a possible signal, or setting an upper limit.

The PL test statistic can be extended in a straightforward way to more than one measurement, and to many nuisance parameters.

5.1 PL for significance estimation

We can use a test statistic that is based on the PL to test the significance of a $\hat{\mu} > 0$ result from the best fit of our data. The test statistics that we are going to use is q_0 defined as follows:

$$q_0 = \begin{cases} -2\ln\lambda_0(N_{obs}) & \hat{\mu} \geq 0 \\ 0 & \hat{\mu} < 0 \end{cases} \tag{7}$$

What does this mean in practice? First of all, we are testing against the background only hypothesis, so we use the PL for the case $\mu = 0$, $\lambda_0(N_{obs})$ (as we were mentioning before, we can compute the PL for any choice of μ). Then, since we regard only upward fluctuations of the data as possible signals, we distinguish the cases in which the best fit value for the signal strength $\hat{\mu}$ is larger (possible signal) or smaller (background underfluctuation) than 0. As mentioned above, by the Wilks theorem, q_0 **is distributed asymptotically as a χ^2 with 1 degree of freedom in the background only hypothesis.**

This means that, given the value of q_0 for our dataset, one can simply compute the p-value as:

$$p = 1 - \Phi(\sqrt{q_0}), \tag{8}$$

where Φ is the cumulative distribution of a Gaussian with average 0 and standard deviation 1. Similarly the "number of sigmas" of the signal is

$$Z = \sqrt{q_0} \tag{9}$$

5.2 PL for upper limit determination

We can use a test statistics based on PL, introduced by Feldmann and Cousins, to set an upper limit on out parameter of interest μ. The test statistics that we are going to use is q_μ is defined as follows:

$$q_\mu = \begin{cases} -2\ln\frac{\mathcal{L}(\mu;\hat{\hat{\mu}}_b)}{\mathcal{L}(\hat{\mu};\hat{\mu}_b)} & \hat{\mu} \geq 0 \\ -2\ln\frac{\mathcal{L}(\mu;\hat{\hat{\mu}}_b)}{\mathcal{L}(0;\hat{\mu}_b)} & \hat{\mu} < 0 \end{cases} \tag{10}$$

This test statistics is distributed asymptotically as a χ^2 with 1 degree of freedom in the μ hypothesis. This means that if the signal exists and its signal strength is μ, q_μ is distributed as a χ^2 with 1 degree of freedom. We can then use this fact to compute the p-value of the μ hypothesis in the asymptotic approximation. As for the significance $p = 1 - \Phi(\sqrt{q_\mu})$. The μ hypothesis is excluded at $1 - \alpha$ CL if $p < \alpha$.

5.3 Brazilian plots

Exclusion limits are often shown in the form of what has become known as the Brazilian plot, due to the choice of colors. An example is shown in 6.

The **expected limit is the limit you expect to put in the background only hypothesis**. It can be derived assuming you observe data that are exactly equal to the sum of your background contributions. Such a dataset, in which one assumes to observe exactly the expected background, is also called background-only Asimov dataset. This however does not give you any sense of how the statistical fluctuations in the number of observed events may affect the limit even if the background only hypothesis is actually true.

In order to have such an idea one can make several "toy" experiment, i.e. simulate what the result of an experiment could be in a background only hypothesis. For each random extraction, one can compute the limit. The 1 and 2 sigma band can be extracted by taking the 2.5% percentile (2σ down), 34% (1σ down), 84% (1σ up), 97.5% (2σ up) of the distribution of limits from toys for each mass. The observed limit is the limit obtained from data,

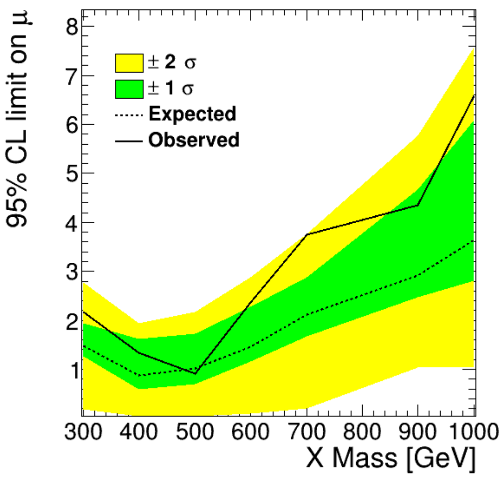


Figure 6: Expected and observed limit. The result of Exercise 3 should be something of this form.

using the real data.

5.4 Setting up a model in RooFit

Based on our previous discussion, it is clear that the first step to use the PL as a means to measure the significance of a signal is to setup a maximum likelihood fit of the POI with constraints on the nuisances.

Stated differently, we want to setup a parametric model which describes our data. The parameters of the model will be the POI and the nuisances. This can be achieved in a relatively easy way in `RooFit`. In order to make the code more generic, we will drop the simplification of the single counting analysis, and we will assume that our measurement consists of a set of event counts, $\vec{N}_{obs} = (N^1_{obs}...N^n_{obs})$, which is represented practically by a histogram, a `TH1` in `ROOT` language. Of course, the simple counting experiment can be simply modeled in this framework as a single-bin histogram.

The following is a guide describing the `RooFit` code used to implement the ML fit we are interested in. The code implementing the fit is in the `HWWWorkspace.C` file.

Let us call `h_data` a histogram of event counts in a certain variable ($m_{T,i}$ for example) after selection. Our task is to write a parametric model that describes that histogram as the sum of different signal and background components.

For the signal and each background we are able to derive from simulation the expected shape and normalization in the relevant variable in the form of a histogram. We can thus derive from simulation the `TH1s` represented by the following variables: `ww`, `top`, `dytt`, `vv`, `sig`. The variable names correspond to the process that is represented by each variable.

From each of those variable we derive a binned PDF (represented by instances of class `RooHistPdf`). The variable names of these PDFs in `HWWWorkspace.C` are: `pdf_ww`, `pdf_top`, `pdf_dytt`, `pdf_vv`, `pdf_s`, again with hopefully obvious meaning of the names.

We also single out variables (instances of class `RooRealVar`) representing the normalization of each process (`nu_ww`, `nu_top`, `nu_dytt`, `nu_vv`, `nu_s`). Also, and most importantly, we introduce instances of class `RooRealVar` to describe the POI (`mu`) and a nuisance representing a multiplier for each background process (`mu_ww`, `mu_top`, `mu_dytt`, `mu_vv`). We describe introduce formulas for the normalization of each signal and background, so that, for example, the normalization of the WW background is represented by variable `norm_ww` which is the product `mu_ww*nu_ww`. Similarly for the signal and the other backgrounds.

Finally, the model that we will use to fit the data is represented by the variable `model` (an instance of class `RooAddPdf`), which effectively represents the expression:

`model=norm_s*pdf_s+ norm_ww*pdf_ww+ norm_top*pdf_top+ norm_dytt*pdf_dytt+ norm_vv*pdf_vv`.

In the file `HWWWorkspace.C` we also prepare the nuisance constraints, to be used later. These have the form of log-normal distributions of the `mu_` parameters.

Everything is saved in an instance of `RooWorkspace` and returned to the calling code, for later use.

Exercise 3

Observed significance with LLR

Using the workspace obtained with `HWWWorkspace.C` it is relatively easy to compute the significance of the observed data. Given the `model` we can construct the negative log likelihood (NLL) using an expression of the form:

`RooAbsReal* nll=model->createNLL(data, ExternalConstraints(*constraints) )`

where `data` is a `RooDataHist` representing the event counts in data and `constraints` is the set of Gaussian constraints on the nuisance.

Starting from the variable `nll` we can then construct the LLR with the expression:

`RooAbsReal* profileLogLikelihood = nll->createProfile(*mu) ;` and evaluate it for a value equal to 0 for the variable `mu`, so that we can construct the test statistics of Eq. 7.

Exercise 4

Upper limit with LLR Using the workspace one can compute the upper limit with the test statistics defined in Eq. 10. Once one had the negative log likelihood NLL, as described above, one can compute the denominator of Eq. 10 with something like:

```
RooAbsReal* nll = model->createNLL(*data,
    ExternalConstraints(*constraints) );
RooMinuit m(*nll);
m.setPrintLevel(-1000); //silence minuit
m.migrad(); //minimize
if (mu->getVal() < 0){ //if best fit is negative
  //redo the fit freezing mu to 0
```

```
  mu->setVal(0);
  mu->setConstant(true);
  m.migrad();
}
double minNLL = nll->getVal(); //value of the negative log
   likelihood at global minimum
```

Then one can compute q_μ with

```
mu->setVal(k*0.01); //set mu to a given value
mu->setConstant(true); // make it constant, so the fit does not
   touch it
m.migrad(); // minimize
double minNLLmu = nll->getVal(); // net the minimum of NLL for
   that choice of mu
double q_mu = 2*(minNLLmu - minNLL); // compute q_mu
double CLsb = 1.-ROOT::Math::chisquared_cdf(q_mu,1.); // compute
   its p-value
```

Please make the Brazilian plot, testing both the simple cut and count analysis of Exercise 2, and an analysis in which you feed the machinery with the m_i^T distribution directly, for example with 10 bins. Which analysis yields the **better** expected limit? **Why do you think it would be a very bad idea to try and optimize the observed limit (or the observed significance)?**