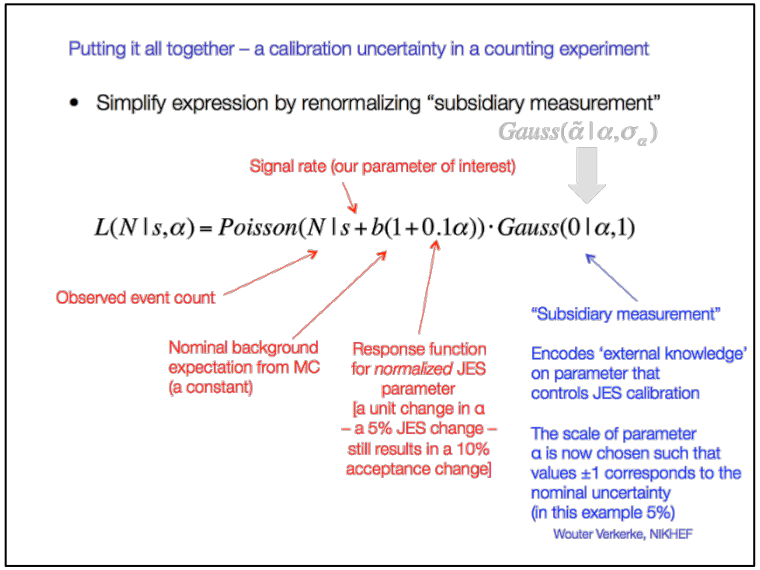
Example 1: counting expt



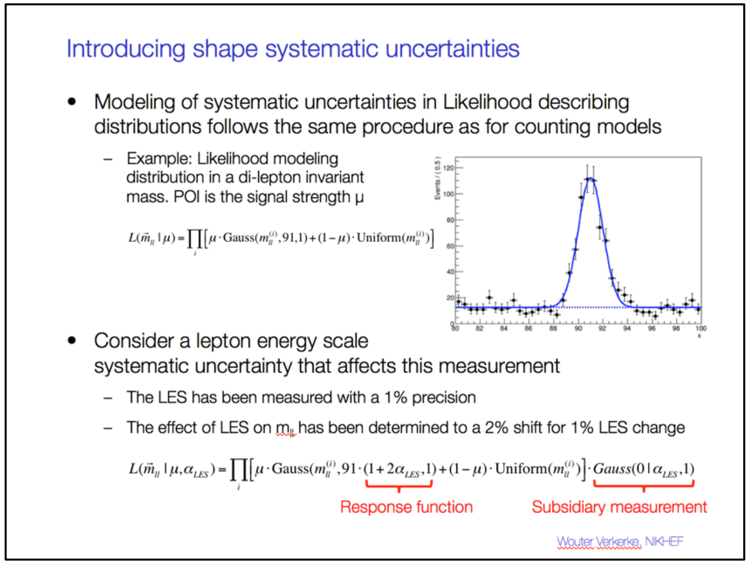
- Will now demonstrate how to construct a model for a counting experiment with a systematic uncertainty

$$L(N \mid s,\alpha) = Poisson(N \mid s + b(1+0.1\alpha)) \cdot Gauss(0 \mid \alpha, 1)$$

```
// Subsidiary measurement of alpha
w.faxtory("Gaussian::subs(0,alpha[-5,5],1)") ;

// Response function mu(alpha)
w.factory("expr::mu('s+b(1+0.1*alpha)',s[20],b[20],alpha)") ;

// Main measurement
w.factory("Poisson::p(N[0,10000],mu)");

// Complete model Physics*Subsidiary
w.factory("PROD::model(p,subs)") ;
```

Example 2: unbinned L with syst.

- Will now demonstrate how to code complete example of the unbinned profile likelihood of Section 5:

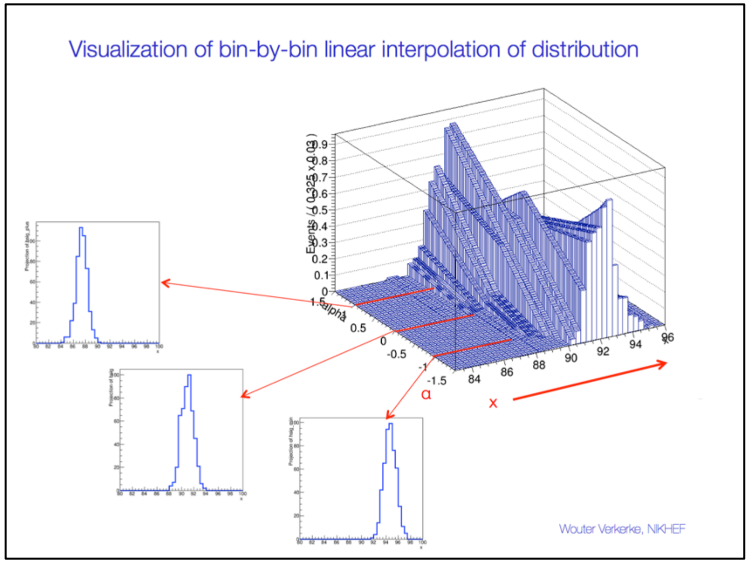


$$L(\vec{m}_{ll} \mid \mu, \alpha_{LES}) = \prod_i \left[\mu \cdot \mathrm{Gauss}(m_{ll}^{(i)}, \underbrace{91 \cdot (1+2\alpha_{LES}), 1)} + (1-\mu) \cdot \mathrm{Uniform}(m_{ll}^{(i)}) \right] \cdot \underbrace{Gauss(0 \mid \alpha_{LES}, 1)}$$

```
// Subsidiary measurement of alpha
w.factory("Gaussian::subs(0,alpha[-5,5],1)");

// Response function m(alpha)
w.factory("expr::m_a("m*(1+2alpha)",m[91,80,100],alpha)") ;

// Signal model
w.factory("Gaussian::sig(x[80,100],m_a,s[1])")

// Complete model Physics(signal plus background)*Subsidiary
w.factory("PROD::model(SUM(mu[0,1]*sig,Uniform::bkg(x)),subs)") ;
```

Example 3 : binned L with syst

- Example of template morphing systematic in a binned likelihood



$$s_i(\alpha,\ldots) = \begin{cases} s_i^0 + \alpha \cdot (s_i^+ - s_i^0) & \forall \alpha > 0 \\ s_i^0 + \alpha \cdot (s_i^0 - s_i^-) & \forall \alpha < 0 \end{cases}$$

$$L(\vec{N} \mid \alpha, \vec{s}^-, \vec{s}^0, \vec{s}^+) = \prod_{bins} P(N_i \mid \underbrace{s_i(\alpha, s_i^-, s_i^0, s_i^+)}) \cdot \underbrace{G(0 \mid \alpha, 1)}$$

```
// Import template histograms in workspace
 w.import(hs_0,hs_p,hs_m) ;

 // Construct template models from histograms
 w.factory("HistFunc::s_0(x[80,100],hs_0)") ;
 w.factory("HistFunc::s_p(x,hs_p)") ;
 w.factory("HistFunc::s_m(x,hs_m)") ;

 // Construct morphing model
 w.factory("PiecewiseInterpolation::sig(s_0,s_,m,s_p,alpha[-5,5])") ;

 // Construct full model
 w.factory("PROD::model(ASUM(sig,bkg,f[0,1]),Gaussian(0,alpha,1))") ;
```

Example 4 – Beeston-Barlow light

- Beeston-Barlow-(lite) modeling of MC statistical uncertainties

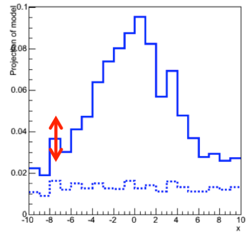
$$L(\vec{N}\,|\,\vec{\gamma}) = \underbrace{\prod_{bins} P(N_i\,|\,\gamma_i(\tilde{s}_i+\tilde{b}_i))}\underbrace{\prod_{bins} P(\tilde{s}_i+\tilde{b}_i\,|\,\gamma_i(\tilde{s}_i+\tilde{b}_i))}$$

Reducing the number NPs – Beeston-Barlow 'lite'

- Another approach that is being used is called 'BB' – lite
- Premise: effect of statistical fluctuations on sum of templates is dominant → Use one NP per bin instead of one NP per component per bin

'Beeston-Barlow'

$$L(\vec{N}\,|\,\vec{s},\vec{b}) = \prod_{bins} P(N_i\,|\,s_i+b_i)\prod_{bins} P(\tilde{s}_i\,|\,s_i)\prod_{bins} P(\tilde{b}_i\,|\,b_i)$$



'Beeston-Barlow lite '

$$L(\vec{N}\,|\,\vec{n}) = \prod_{bins} P(N_i\,|\,n_i)\prod_{bins} P(\tilde{s}_i+\tilde{b}_i\,|\,n_i)$$

Response function w.r.t. n as parameters

Subsidiary measurements of n from s~+b~

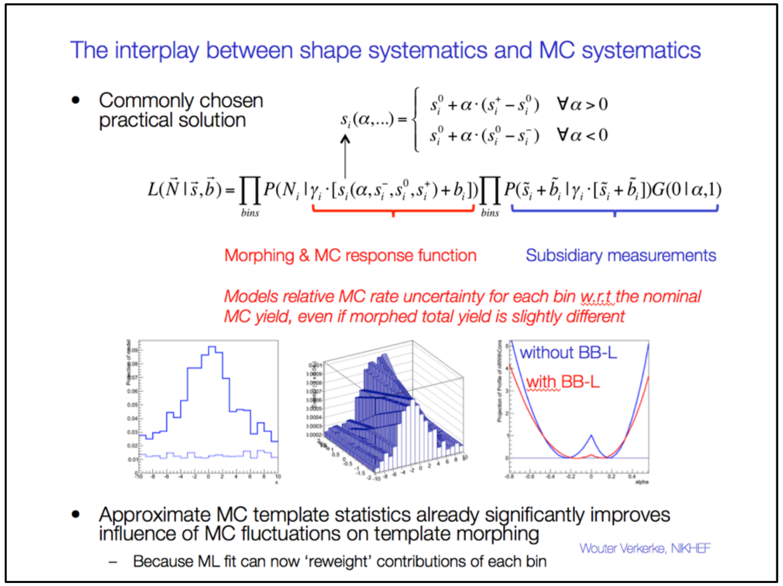
$$L(\vec{N}\,|\,\vec{\gamma}) = \prod_{bins} P(N_i\,|\,\gamma_i(\tilde{s}_i+\tilde{b}_i))\prod_{bins} P(\tilde{s}_i+\tilde{b}_i\,|\,\gamma_i(\tilde{s}_i+\tilde{b}_i))$$

Normalized NP lite model (nominal value of all γ is 1)

```
// Import template histogram in workspace
 w.import(hs) ;

// Construct parametric template models from histograms
// implicitly creates vector of gamma parameters
 w.factory("ParamHistFunc::s(hs)") ;

 // Product of subsidiary measurement
 w.factory("HistConstraint::subs(s)") ;

 // Construct full model
 w.factory("PROD::model(s,subs)") ;
```

Example 5 – BB-lite + morphing

- Template morphing model with Beeston-Barlow-lite MC statistical uncertainties



$$
s_i(\alpha,\ldots) = \begin{cases} s_i^0 + \alpha \cdot (s_i^+ - s_i^0) & \forall \alpha > 0 \\ s_i^0 + \alpha \cdot (s_i^0 - s_i^-) & \forall \alpha < 0 \end{cases}
$$

$$
L(\vec{N} \mid \vec{s}, \vec{b}) = \prod_{bins} P(N_i \mid \gamma_i \cdot [\underbrace{s_i(\alpha, s_i^-, s_i^0, s_i^+)}_{} + b_i]) \underbrace{\prod_{bins} P(\tilde{s}_i + \tilde{b}_i \mid \gamma_i \cdot [\tilde{s}_i + \tilde{b}_i]) G(0 \mid \alpha, 1)}_{}
$$

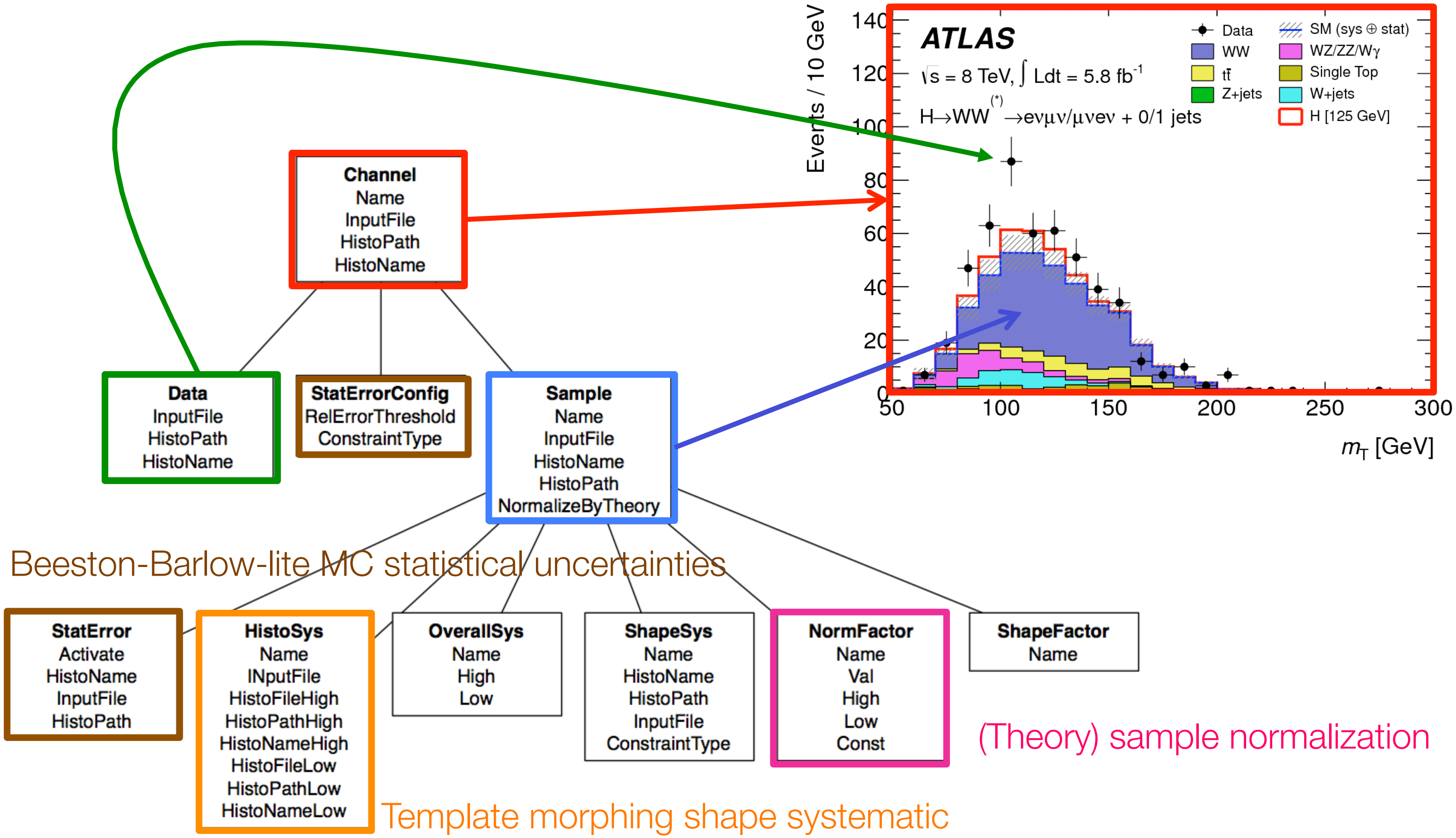
```
// Import template histograms in workspace
 w.import(hs_0,hs_p,hs_m,hb) ;

 // Construct parametric template morphing signal model
 w.factory(“ParamHistFunc::s_p(hs_p)”) ;
 w.factory(“HistFunc::s_m(x,hs_m)”) ;
 w.factory(“HistFunc::s_0(x[80,100],hs_0)”) ;
 w.factory(“PiecewiseInterpolation::sig(s_0,s_,m,s_p,alpha[-5,5])”) ;

 // Construct parametric background model (sharing gamma’s with s_p)
 w.factory(“ParamHistFunc::bkg(hb,s_p)”) ;

 // Construct full model with BB-lite MC stats modeling
 w.factory(“PROD::model(ASUM(sig,bkg,f[0,1]),
            HistConstraint({s_0,bkg}),Gaussian(0,alpha,1))”) ;
```

HistFactory – structured building of binned template models

- RooFit modeling building blocks allow to easily construct likelihood models that model shape and rate systematics with one or more nuisance parameter
  - Only few lines of code per construction
- Typical LHC analysis required modeling of 10-50 systematic uncertainties in O(10) samples in anywhere between 2 and 100 channels → Need structured formalism to piece together model from specifications. This is the purpose of HistFactory
- HistFactory conceptually similar to workspace factory, but has much higher level semantics
  - Elements represent physics concepts (channels, samples, uncertainties and their relation) rather than mathematical concepts
  - Descriptive elements are represented by C++ objects (like roofit), and can be configured in C++, or alternively from an XML file
  - Builds a RooFit (mathematical) model from a HistFactory physics model.

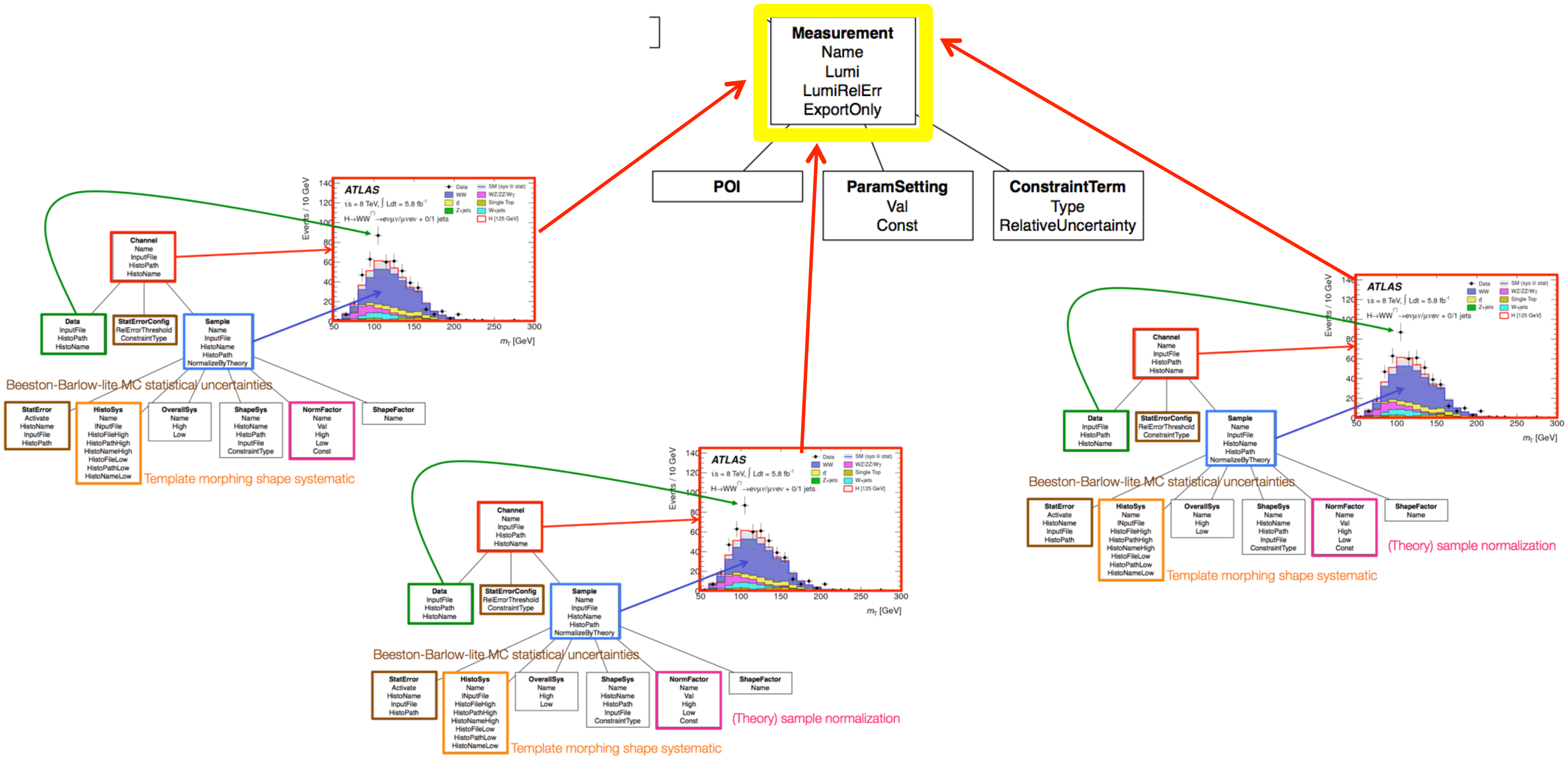
HistFactory elements of a channel

- Hierarchy of concepts for description of one measurement channel



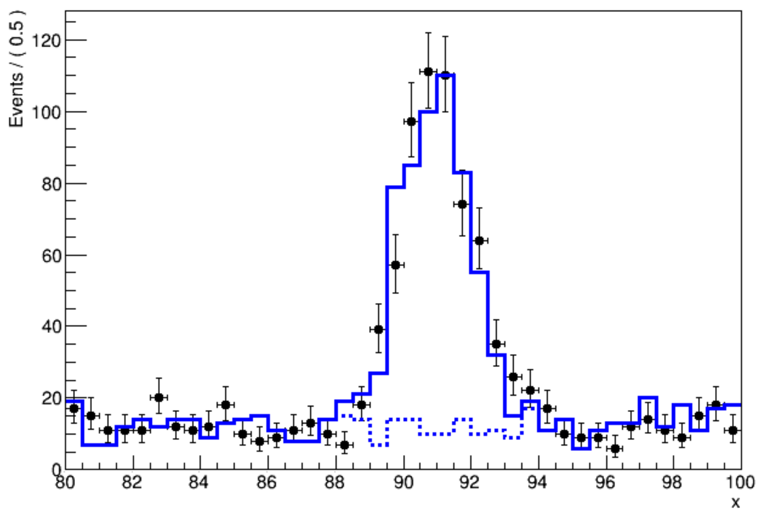
HistFactory elements of measurement

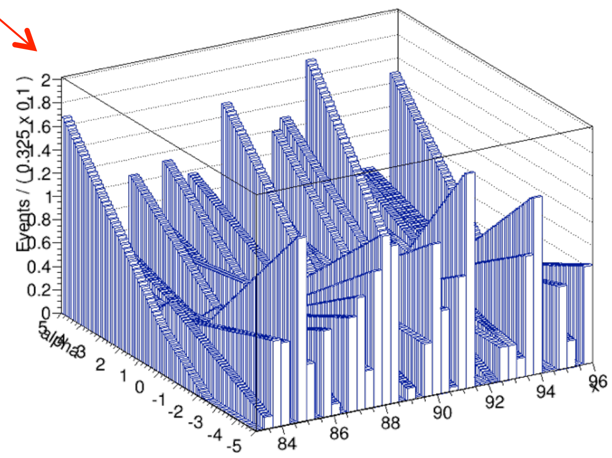
- One or more channels are combined to form a measurement
  - Along with some extra information (declaration of the POI, the luminosity of the data sample and its uncertainty)



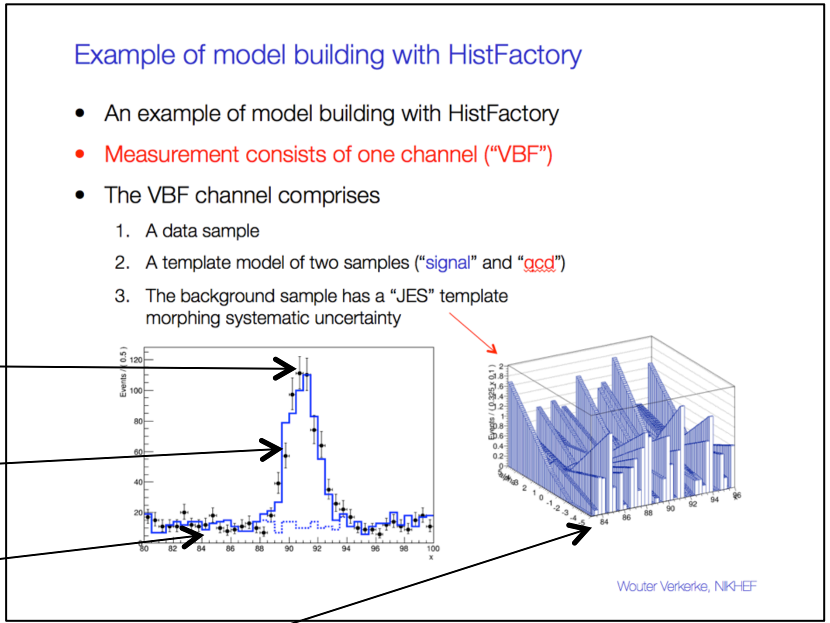
Example of model building with HistFactory

- An example of model building with HistFactory
- Measurement consists of one channel ("VBF")
- The VBF channel comprises
  1. A data sample
  2. A template model of two samples ("signal" and "qcd")
  3. The background sample has a "JES" template morphing systematic uncertainty





Model building with HistFactory




```
// external input in form of TH1 shown in green

// Declare ingredients of measurement
HistFactory::Data data ;
data.SetHisto(data_hist) ;

HistFactory::Sample signal("signal") ;
signal.SetHisto(sample_hist) ;

HistFactory::Sample qcd("QCD") ;
qcd.SetHisto(sample_hist) ;

HistFactory::HistoSys hsys("QCD_JetEnergyScale") ;
hsys.SetHistoLow(sample_hist_sysdn) ;
hsys.SetHistoHigh(sample_hist_sysup) ;
qcd.AddHistoSys(hsys) ;

HistFactory::Channel channel("VBF") ;
channel.SetData(data) ;
channel.AddSample(sample) ;

HistFactory::Measurement meas("MyAnalysis") ;
meas.AddChannel(channel) ;

// Now build RooFit model according to specs
HistFactory::HistoToWorkspaceFactoryFast h2w(meas) ;
RooWorkspace* w = h2w.MakeCombinedModel(meas) ;
w->Print("t") ;
w->writeToFile("test.root") ;
```

HistFactory model output

- Contents of RooFit workspace produced by HistFactory

RooFit probability model as specified

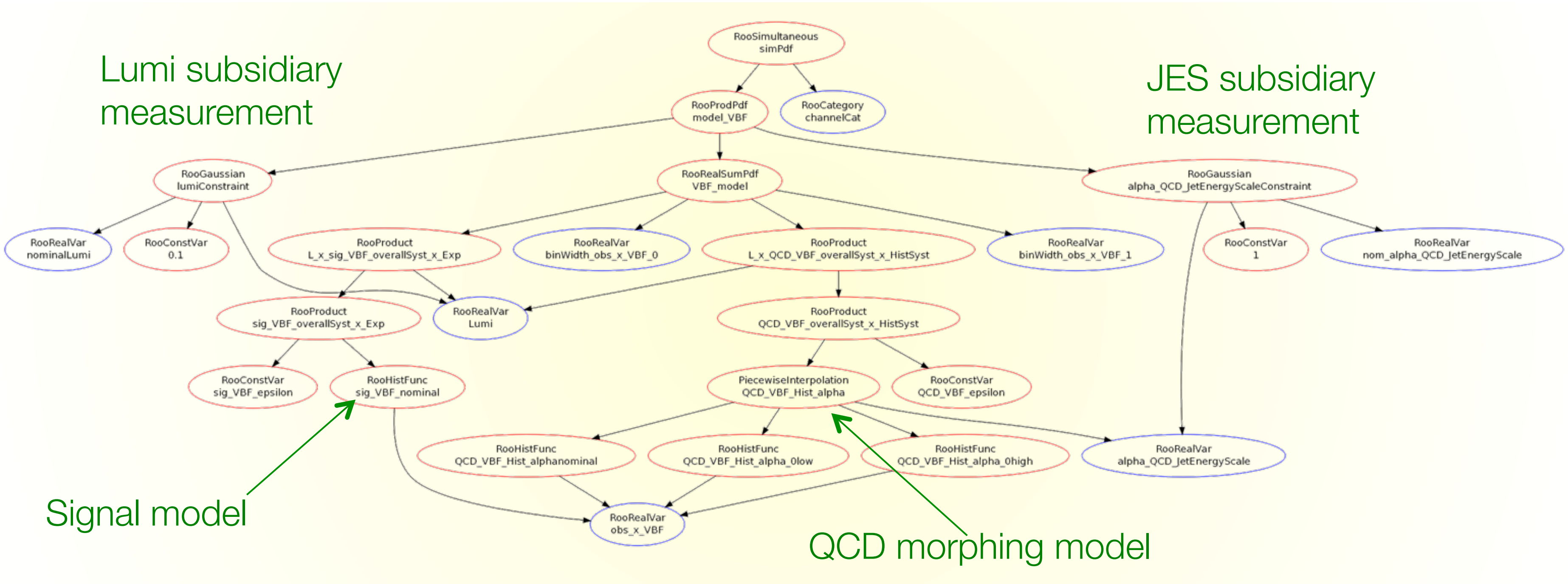
Definition of POI, NPs, Observables Global observables

Universal Model Configuration

```
RooWorkspace(combined) combined contents

variables
---------
(Lumi,alpha_QCD_JetEnergyScale,binWidth_obs_x_VBF_0,binWidth_obs_x_VBF_1,channelCat,
 nom_alpha_QCD_JetEnergyScale,nominalLumi,obs_x_VBF,weightVar)

p.d.f.s
-------
RooSimultaneous::simPdf[ indexCat=channelCat VBF=model_VBF ] = 0
  RooProdPdf::model_VBF[ lumiConstraint * alpha_QCD_JetEnergyScaleConstraint * VBF_model(obs_x_VBF) ] = 0
    RooGaussian::lumiConstraint[ x=Lumi mean=nominalLumi sigma=0.1 ] = 1
    RooGaussian::alpha_QCD_JESConstraint[ x=alpha_QCD_JetEnergyScale mean=nom_alpha_QCD_JetEnergyScale sigma=1 ] = 1
    RooRealSumPdf::VBF_model[ binW_obs_x_VBF_0 * L_x_sig_VBF_overallSyst_x_Exp + binW_obs_x_VBF_1 * L_x_QCD_VBF_overallSyst_x_HistSyst ] = 0
      RooProduct::L_x_sig_VBF_overallSyst_x_Exp[ Lumi * sig_VBF_overallSyst_x_Exp ] = 0
        RooProduct::sig_VBF_overallSyst_x_Exp[ sig_VBF_nominal * sig_VBF_epsilon ] = 0
          RooHistFunc::sig_VBF_nominal[ depList=(obs_x_VBF) ] = 0
      RooProduct::L_x_QCD_VBF_overallSyst_x_HistSyst[ Lumi * QCD_VBF_overallSyst_x_HistSyst ] = 0
        RooProduct::QCD_VBF_overallSyst_x_HistSyst[ QCD_VBF_Hist_alpha * QCD_VBF_epsilon ] = 0
          PiecewiseInterpolation::QCD_VBF_Hist_alpha[ ] = 0
            RooHistFunc::QCD_VBF_Hist_alphanominal[ depList=(obs_x_VBF) ] = 0
            RooHistFunc::QCD_VBF_Hist_alpha_0low[ depList=(obs_x_VBF) ] = 0
            RooHistFunc::QCD_VBF_Hist_alpha_0high[ depList=(obs_x_VBF) ] = 0

datasets
--------
RooDataSet::asimovData(obs_x_VBF,weightVar,channelCat)
RooDataSet::obsData(channelCat,obs_x_VBF)

embedded datasets (in pdfs and functions)
-----------------------------------------
RooDataHist::sig_VBFnominalDHist(obs_x_VBF)
RooDataHist::QCD_VBF_Hist_alphanominalDHist(obs_x_VBF)
RooDataHist::QCD_VBF_Hist_alpha_0lowDHist(obs_x_VBF)
RooDataHist::QCD_VBF_Hist_alpha_0highDHist(obs_x_VBF)

parameter snapshots
-------------------
NominalParamValues = (nominalLumi=1[C],nom_alpha_QCD_JetEnergyScale=0[C],weightVar=0,obs_x_VBF=-4.5,Lumi=1,alpha_QCD_JetEnergyScale=0,
                     binWidth_obs_x_VBF_0=1[C],binWidth_obs_x_VBF_1=1[C])

named sets
----------
ModelConfig_GlobalObservables:(nominalLumi,nom_alpha_QCD_JetEnergyScale)
ModelConfig_Observables:(obs_x_VBF,weightVar,channelCat)
ModelConfig_POI:()
globalObservables:(nominalLumi,nom_alpha_QCD_JetEnergyScale)
observables:(obs_x_VBF,weightVar,channelCat)

generic objects
---------------
RooStats::ModelConfig::ModelConfig
```

HistFactory model structure

- RooFit object structure
  - As visalized with `simPdf::graphVizTree("model.dot")` followed by `dot -Tpng -omodel.png model.dot'`



- This RooFit probability model can be evaluated without knowledge of HistFactory
  - Additional (documentary) information stored in workspace specifies a uniquely specified statistical model (definition of POI, NP etc)

Working with the Likelihood – fitting & estimation

- Given an arbitrary RooFit probability model and observed data stored in a ROOT file in a RooWorkspace
  - For example using naming hints stored in ModelConfig object

```
RooWorkspace* w = gDirectory->Get("combined") ;

// Take pdf defined by RooStats model configuration
RooStats::ModelConfig* mc = w->genobj("ModelConfig") ;
RooAbsPdf* pdf = mc->GetPdf() ;

RooAbsData* obsData = w->data("obsData") ;
```

  - From here code same for every model, from Poisson counting model to full ATLAS Higgs Combination likelihood

- Construct likelihood function from probability model and data

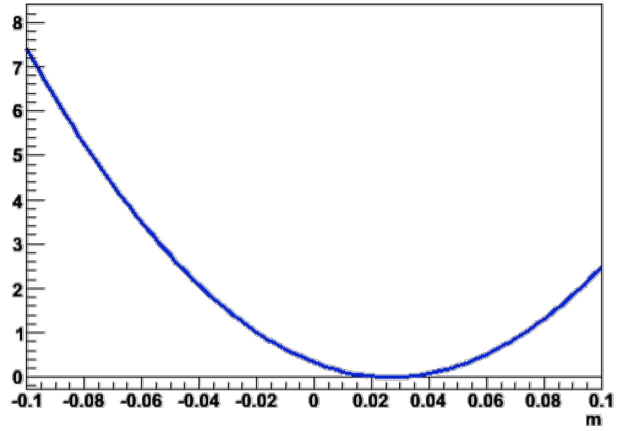
```
RooAbsReal* nll = pdf->createNLL(data) ;
```

  - Likelihood is a RooFit function with all its usual functionality

Working with the likelihood function

- Plot the likelihood function versus a parameter

```
RooAbsReal* nll = w::model.createNLL(data) ;

RooPlot* frame = w::param.frame() ;
nll->plotOn(frame,ShiftToZero()) ;
```

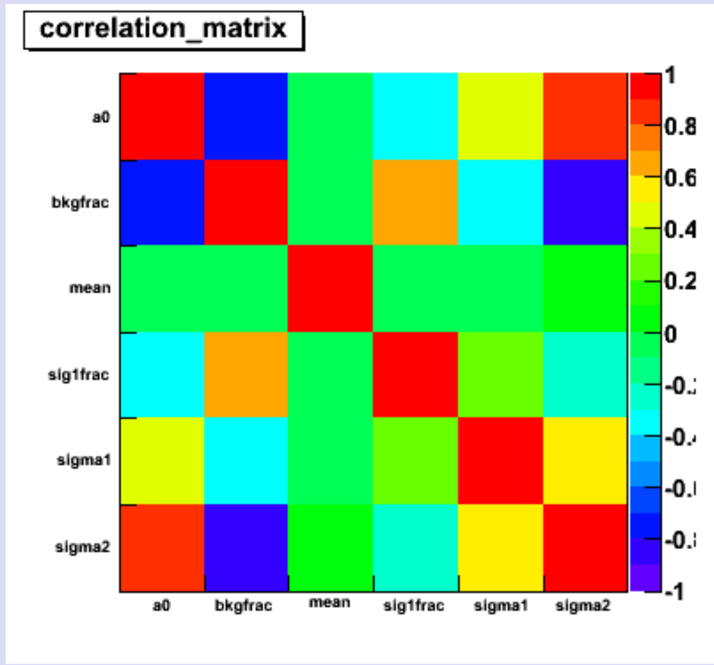


- Maximum Likelihood estimation of parameters and variance

```
RooMinimizer m(*nll) ;

// ML Parameter estimation
m.minimize("Minuit2","migrad") ;

// Variance estimation
m.hesse() ;

// Alternatively - all this in one line
pdf->fitTo(*data) ;
```

Working with covariance and correlation matrices

- Detailed information on parameter and covariance estimates can be saved for detailed information



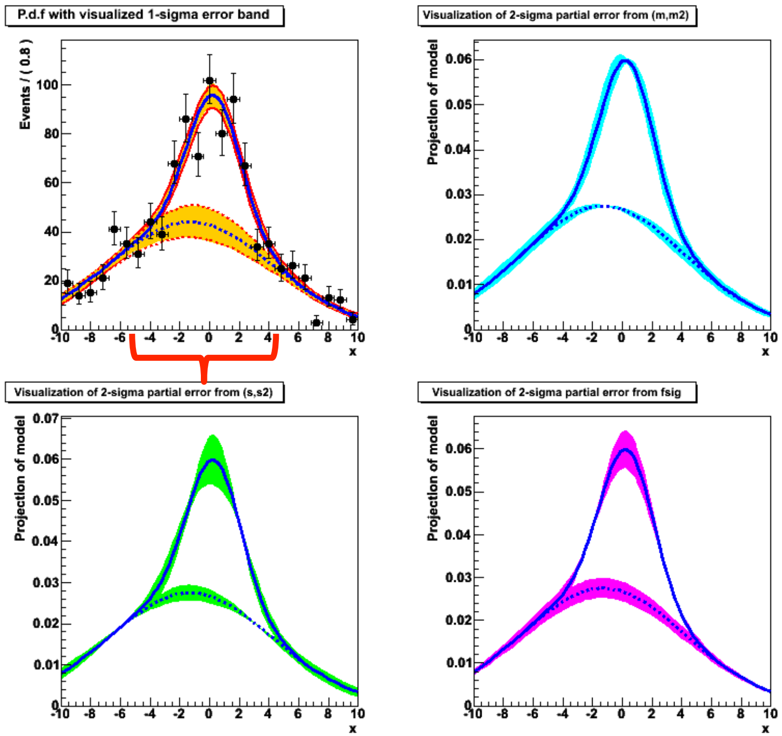

```
RooMinimizer m(*nll) ;
m.minimize("Minuit2","migrad") ;
m.hesse() ;
RooFitResult* r = m.save() ;

// Visualize correlation matrix
r->correlationHist->Draw("colz") ;

// Extract correlation,covariance matrix
TMatrixDSym cov = fr->covarianceMatrix() ;
TMatrixDSym cov = fr->covarianceMatrix(a,b) ;
```

Use covariance matrices for correlated error propagation

- Can (as visual aid) propagate errors in covariance matrix of a fit result to a pdf projection

```
w::model.plotOn(frame,VisualizeError(*fitresult)) ;
w::model.plotOn(frame,VisualizeError(*fitresult,fsig)) ;
```

  - Linear propagation on pdf projection $\Delta = \vec{E} V^{-1} \vec{E}$



- Propagated error can be calculated on arbitrary function
  - E.g fraction of events in signal range

```
RooAbsReal* fracSigRange =
     w::model.createIntegral(x,x,"sig") ;

Double_t err =
    fracSigRange.getPropagatedError(*fr);
```

Working with profile likelihood

- A profile likelihood ratio

$$\lambda(p) = \frac{L(p,\hat{\hat{q}})}{L(\hat{p},\hat{q})}$$

Best L for given p

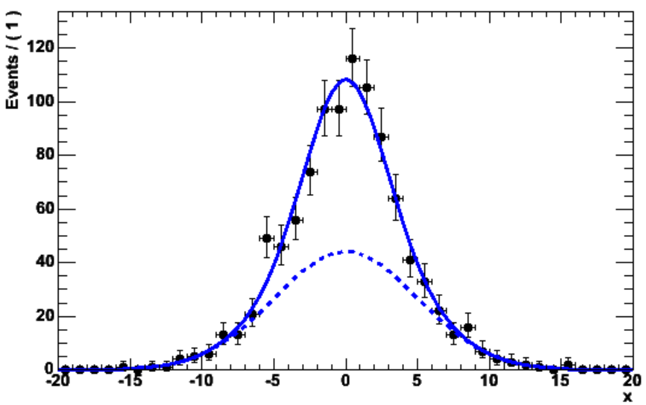
Best L

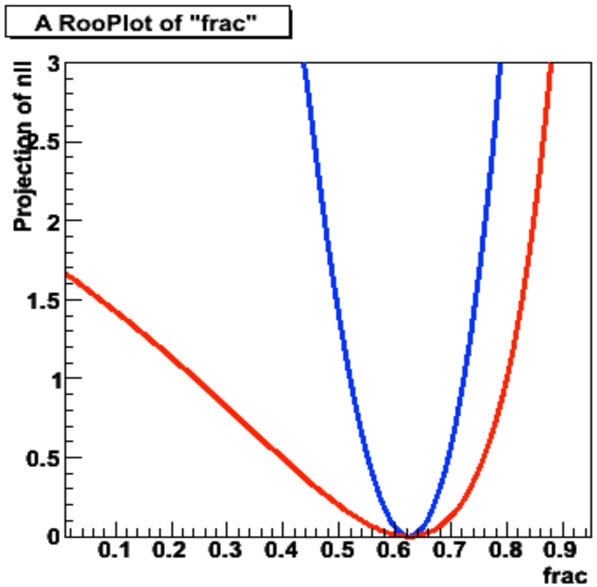
can be represent by a regular RooFit function (albeit an expensive one to evaluate)

Use RooFits universal support for multi-processor likelihood calculation

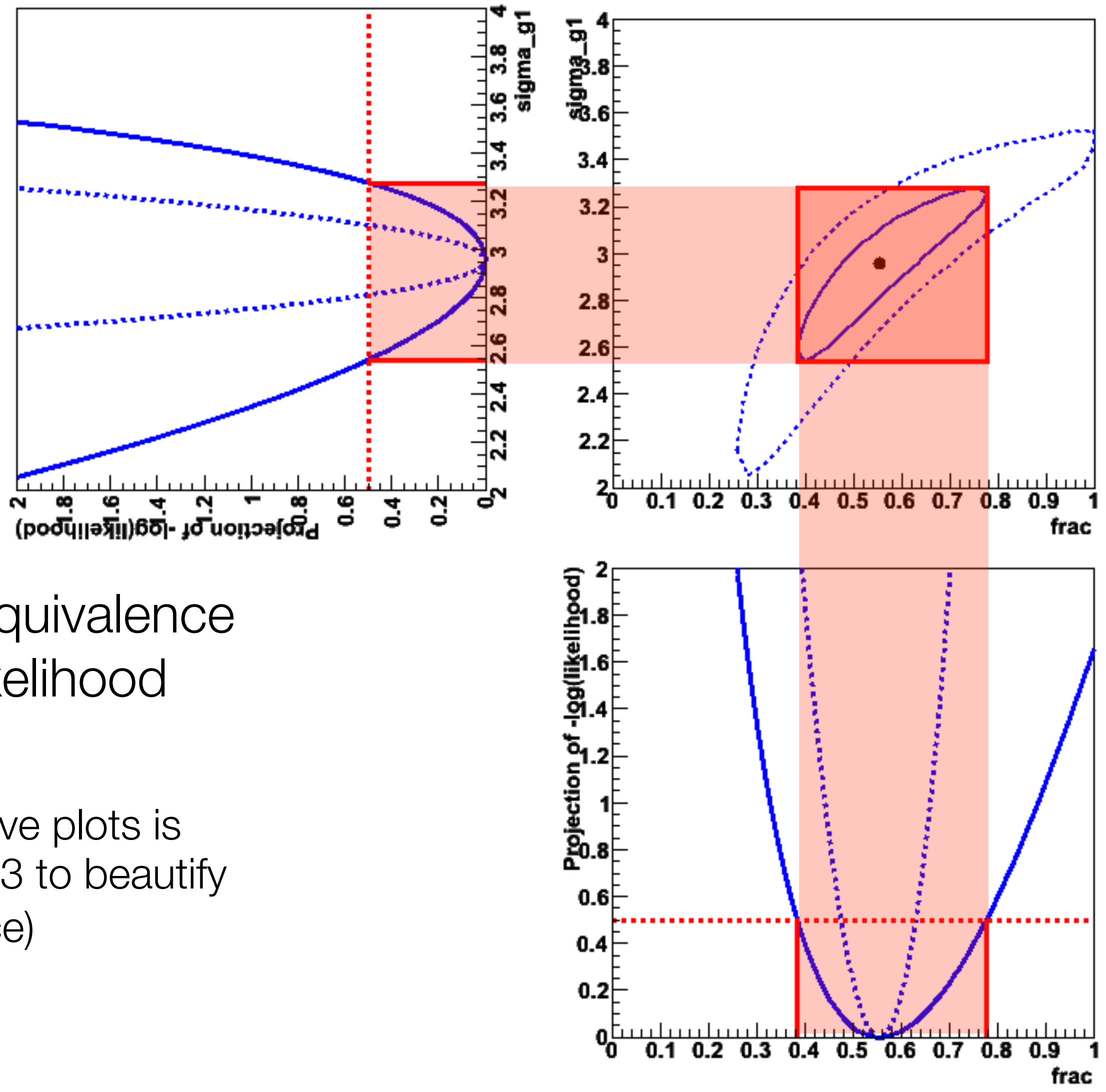
```
RooAbsReal* ll = model.createNLL(data,NumCPU(8)) ;
RooAbsReal* pll = ll->createProfile(params) ;
```

```
RooPlot* frame = w::frac.frame() ;
nll->plotOn(frame,ShiftToZero()) ;
pll->plotOn(frame,LineColor(kRed)) ;
```





On the equivalence of profile likelihood and MINOS



- Demonstration of equivalence of (RooFit) profile likelihood and MINOS errors
  - Macro to make above plots is 34 lines of code (+23 to beautify graphics appearance)

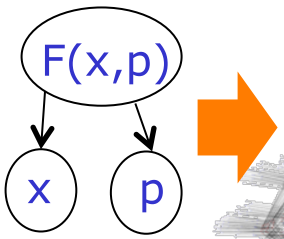
Make your own Higgs combination

- Workspace technology greatly simplifies combination of measurements
- Example: ATLAS Higgs likelihood combination
  - Individual channels build likelihood model in workspace file
  - A posteriori combine likelihood for each channel in combination group
  - Must make sure common parameter have common names, otherwise technically straightforward (in principle)
- Simplified code example

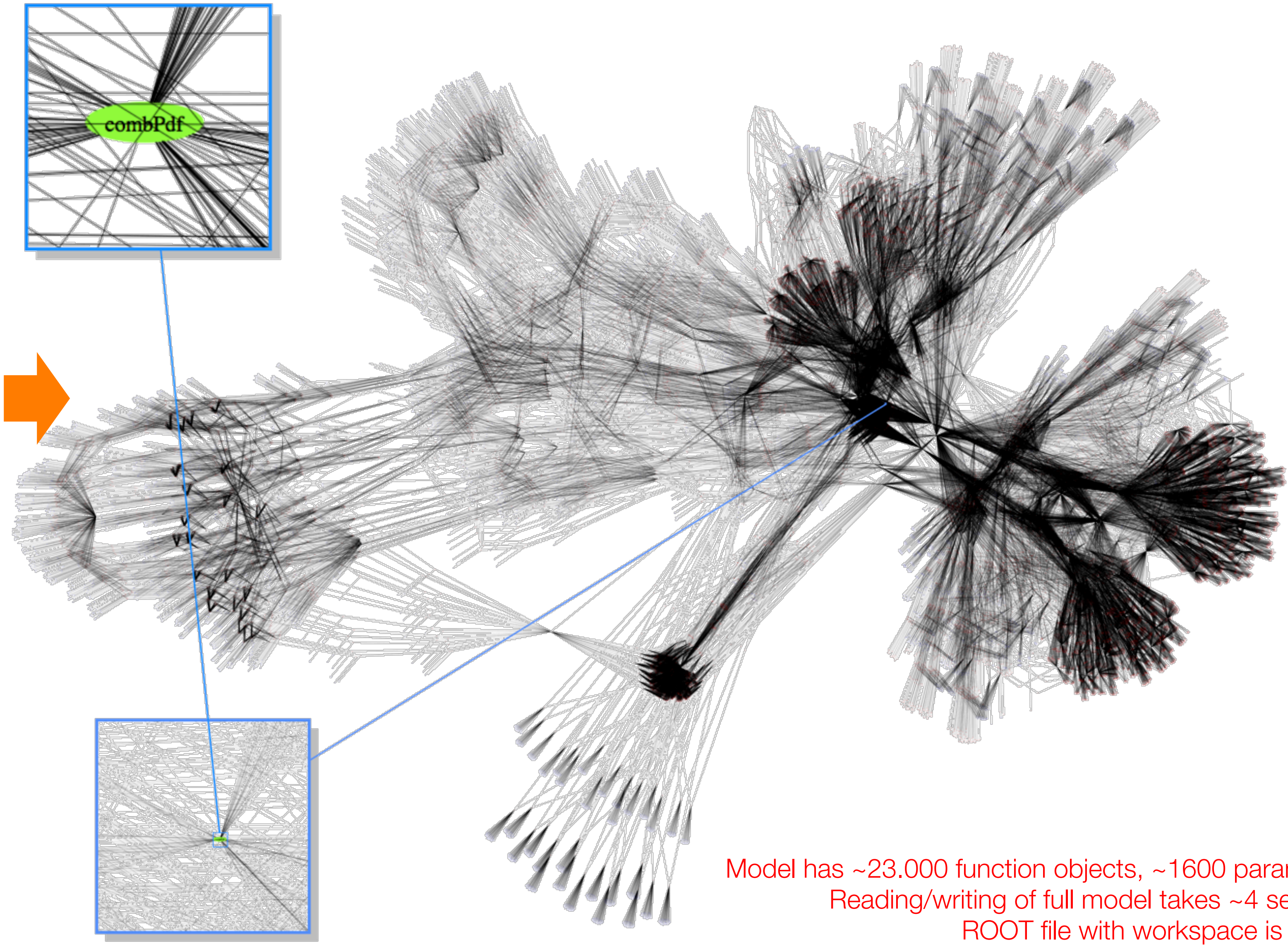
```
RooWorkspace combined("combined") ;

// Import channel models from separate workspace files
w.importFromFile("htoZZ.root:w:masterPdfZZ",…) ;
w.importFromFile("htoWW.root:w:aaronsWWPdf",…) ;

// Create joint pdf
w.factory("SIMUL::joint(index[HWW,HZZ],
                        HZZ=masterPdfZZ,HWW=aaronsWWPdf)") ;
```

  - Real life a bit more complicated, but similar to this concept.

The full ATLAS Higgs combination in a single workspace…



Atlas Higgs combination model (23.000 functions, 1600 parameters)

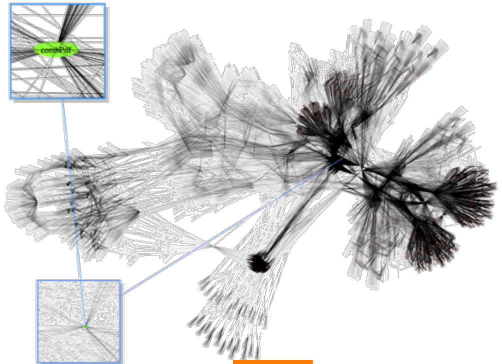


Model has ~23.000 function objects, ~1600 parameters
Reading/writing of full model takes ~4 seconds
ROOT file with workspace is ~6 Mb

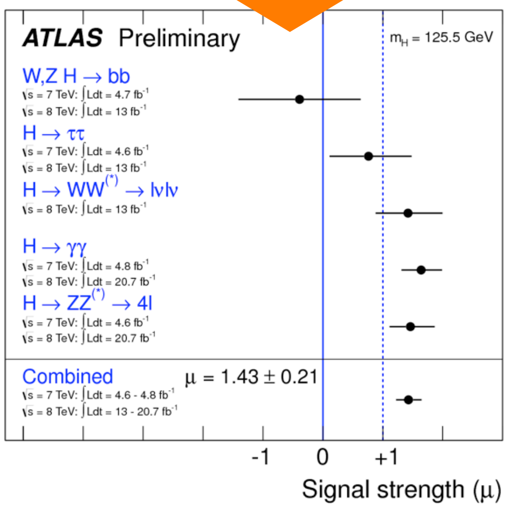
Collaborative analyses with workspaces

- Workspaces allow to share and modify very complex analyses with very little *technical* knowledge required
- Example: Higgs coupling fits

Full Higgs model



Signal strength in 5 channels



$$\sigma(gg \to H) * \mathrm{BR}(H \to \gamma\gamma) \sim \frac{\kappa_F^2 \cdot \kappa_\gamma^2(\kappa_F, \kappa_V)}{0.75 \cdot \kappa_F^2 + 0.25 \cdot \kappa_V^2}$$

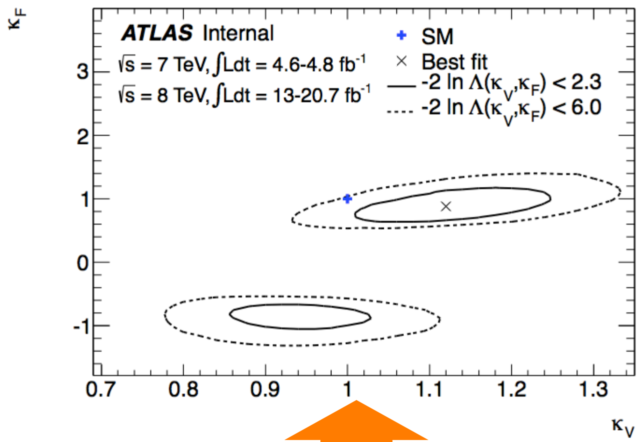
$$\sigma(qq' \to qq'H) * \mathrm{BR}(H \to \gamma\gamma) \sim \frac{\kappa_V^2 \cdot \kappa_\gamma^2(\kappa_F, \kappa_V)}{0.75 \cdot \kappa_F^2 + 0.25 \cdot \kappa_V^2}$$

$$\sigma(gg \to H) * \mathrm{BR}(H \to ZZ^{(*)}, H \to WW^{(*)}) \sim \frac{\kappa_F^2 \cdot \kappa_V^2}{0.75 \cdot \kappa_F^2 + 0.25 \cdot \kappa_V^2}$$

$$\sigma(qq' \to qq'H) * \mathrm{BR}(H \to ZZ^{(*)}, H \to WW^{(*)}) \sim \frac{\kappa_V^2 \cdot \kappa_V^2}{0.75 \cdot \kappa_F^2 + 0.25 \cdot \kappa_V^2}$$

$$\sigma(qq' \to qq'H, VH) * \mathrm{BR}(H \to \tau\tau, H \to b\bar{b}) \sim \frac{\kappa_V^2 \cdot \kappa_F^2}{0.75 \cdot \kappa_F^2 + 0.25 \cdot \kappa_V^2},$$

Reparam in terms of fermion, v-boson scale factors



Confidence intervals on Higgs fermion, v-boson couplings

Collaborative analyses with workspaces

- How can you reparametrize existing Higgs likelihoods *in practice*?
- Write functions expressions corresponding to new parameterization

$$\sigma(gg \to H) * \mathrm{BR}(H \to \gamma\gamma) \sim \frac{\kappa_F^2 \cdot \kappa_\gamma^2(\kappa_F, \kappa_V)}{0.75 \cdot \kappa_F^2 + 0.25 \cdot \kappa_V^2}$$

```
RooFormulaVar mu_gg_func("mu_gg_func",
                         "(KF2*Kg2)/(0.75*KF2+0.25*KV2)",
                         KF2,Kg2,KV2) ;
```

- Edit existing model

```
w.import(mu_gg_func) ;
w.factory("EDIT::newmodel(model,mu_gg=mu_gg_gunc)") ;
```

Top node of ***modified*** *Higgs combination pdf*

Top node of ***original*** *Higgs combination pdf*

Modification prescription
replace parameter mu_gg with function mu_gg_func everywhere

The role of the RooStats package

- Use of likelihoods so far restricted to parameter, variance estimation and MINOS-style intervals

- For p-values and Frequentist confidence intervals need to construct (profile likelihood ratio) test statistic and obtain its (asymptotic distribution)

- RooStats can do these calculations
  - Input is RooWorkspace – contains full likelihood and associated information, only ‘technical’ tool configuration is needed
  - Designed as a toolkit (with classes representing TestStatistics, NeymanConstruction, Intervals, HypothesisTestInverters)
  - Very flexible, but usually requires a bit of coding to setup to achieve the desired configuration.

An example of a custom RooStats driver script

Tool to calculate p-values for a given hypothesis

$$\int_{q_{\mu,obs}}^{\infty} f(q_\mu \mid \mu')dq_\mu$$

```
// create first HypoTest calculator (N.B null is s+b model)
FrequentistCalculator fc(*data, *bModel, *sbModel);

// configure  ToyMCSampler and set the test statistics
ToyMCSampler *toymcs = (ToyMCSampler*)fc.GetTestStatSampler();

ProfileLikelihoodTestStat profll(*sbModel->GetPdf());
// for CLs (bounded intervals) use one-sided profile likelihood
profll.SetOneSided(true);
toymcs->SetTestStatistic(&profll);

HypoTestInverter calc(*fc);
calc.UseCLs(true);

// configure and run the scan
calc.SetFixedScan(npoints,poimin,poimax);
HypoTestInverterResult * r = calc.GetInterval();

// get result and plot it
double upperLimit = r->UpperLimit();
double expectedLimit = r->GetExpectedUpperLimit(0);

HypoTestInverterPlot *plot = new HypoTestInverterPlot("hi","",r);
plot->Draw();
```

$f(q_\mu \mid \mu')$

Tool to construct test statistic distribution

$q_\mu(\mu')$

The test statistic to be used for the calculation of p-values

Tool to construct interval from hypo test results

The ‘standard’ RooStats driver script

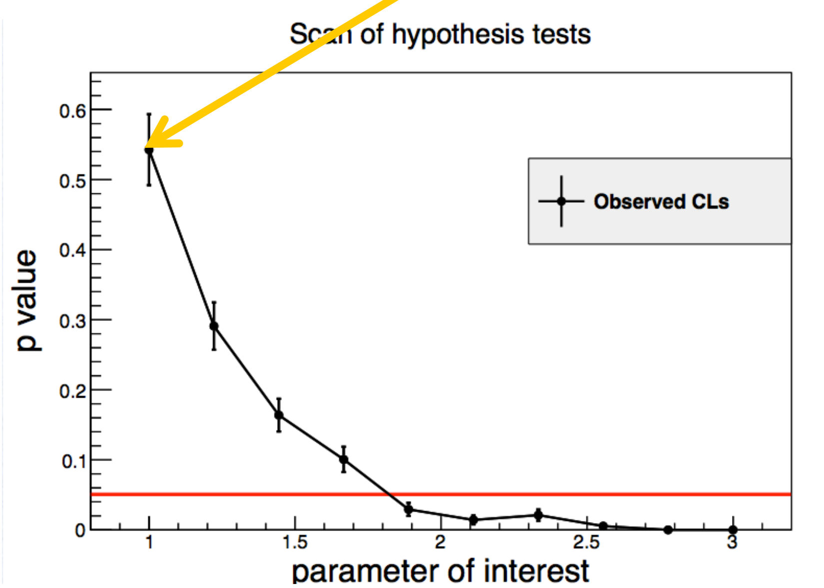
- Input information needed
  - Input workspace (file name and workspace name)
  - Name of ModelConfig object to be used in workspace
    - Specifies S+B model, B model (if not S+B with μ=0), POI, nuisance params etc
  - Name of observed dataset in workspace
- Statistics options
  - Calculator type (Frequentist, Hybrid, Asymptotic)
  - Test statistic (ProfileLR [LHC], RatioOfPLR [TeV], LR [LEP])
  - Use CL_S technique (yes/no)
- Technical options
  - Range of POI to scan
  - Fixed number of steps (for nice plots),
or -1 for adaptive sampling (for precise and fast limit calculations)

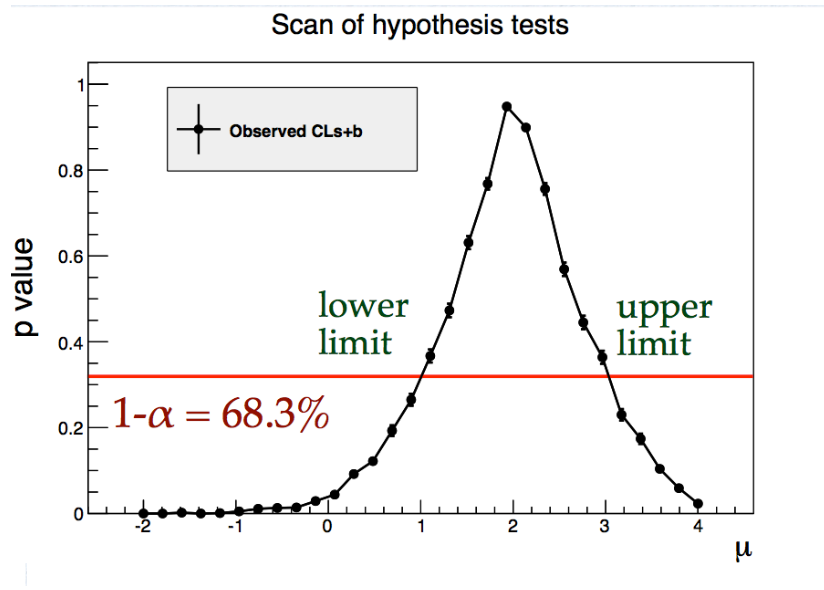
```
load the macro after having create the workspace using given macro (e.g. SPlusBExpoModel.root)
root[] .L StandardHypoTestInvDemo.C

run for CLs (with  frequentist calculator (type = 0) and one-side PL test statistics (type = 3) scan 10 points in [0,100]

root[] StandardHypoTestInvDemo("SPlusBExpoModel.root","w","ModelConfig","","data",0,3, true, 10, 0, 100)

run for Asymptotic CLs (scan 20 points in [0,100])

root[] StandardHypoTestInvDemo(SPlusBExpoModel.root","w","ModelConfig","","data",2,3, true, 20, 0, 100)

run for Feldman-Cousins  ( scan 10 points in [0,100])

root[] StandardHypoTestInvDemo(SPlusBExpoModel.root","w","ModelConfig","","data",0,2, false, 10, 0, 15)
```

Example output of hypothesis test inversion

- Hypothesis test calculator computes p-value for each value of μ

HypoTest result (p-value) at given μ (here μ=1)





One-sided in interval
(upper limit) at 95% C.L.

Two-sided in interval
at 68% C.L.

Summary on RooFit/RooStats/HistFactory

- RooFit is a language to build probability models of arbitrary type, shape and complexity
  - Small set of powerful adjustable building blocks simplify building process (concepts of previous section can all be coded in O(5) lines)
  - Concept of 'workspace' allows complete separation of process of building and using likelihood models
- HistFactory is a descriptive language for measurements exclusively formulated in template likelihood models
  - Declaration of channels, samples and their properties (systematic uncertainties etc) can be turned into a RooFit probability model
- Workspace concept facilitates easy sharing, combining and editing of likelihood functions between analysis groups
- Parameter/Variance estimation and MINOS-style intervals on likelihood models calculated with RooFit/MINUIT tools
  - For 'fundamental methods' (Frequentist/Bayesian statements) RooStats toolkit can perform calculations based in RooFit models

7 Diagnostics I: MINUIT, Fit stability & convergence

MINUIT and convergence of profile likelihood fits

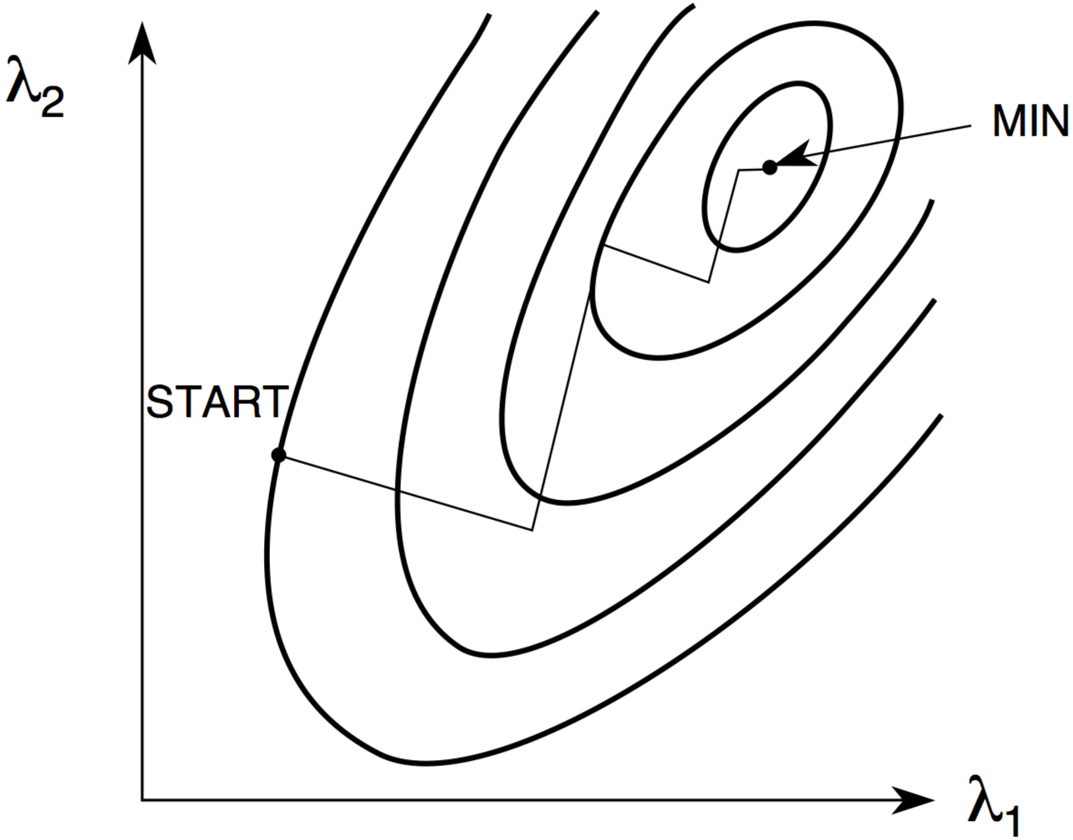
- Likelihoods with systematics modeling ('profile likelihood fits') tend to be more complex than 'normal' fits
- Sometimes these likelihood can have pathological features that frustrate the minimization process
- To help you understand I will briefly cover
  - How MINUIT works and defines 'convergence'
  - Typical problems that occur in profile likelihood models and how these affect MINUIT

MINUIT in a nutshell

- MINUIT is a function minimization and analysis packages written by Fred James
  - Original FORTRAN version more than 40 years old!
  - Currently two versions in C++ in ROOT: TMinuit and Minuit2. Former is a 'machine translated version' from FORTRAN, latter hand-ported version under the supervision of Fred James
  - I recommend to always use Minuit2 – performance has been exhaustively validated against the original minuit and you get much more useful diagnostic information out of it.
- Three analysis routines implement main functionality
  - MIGRAD: Function minimization using the *variable metric method* developed by Fletcher Davidon and Powell. (This is efffectively equivalent to the 'industry standard' method of Broyden, Fletcher, Goldfarb and Shanno 'BFGS')
  - HESSE: Error analysis: Calculates Hessian matrix of 2^{nd} derivatives and inverts this into the covariance matrix
  - MINOS: Calculates intervals based on the profile likelihood ratio

Function minimization using the variable metric method

- MINUIT does *not* implement a simple 'steepest descent' method as plain gradient often does not point well in direction of minimum



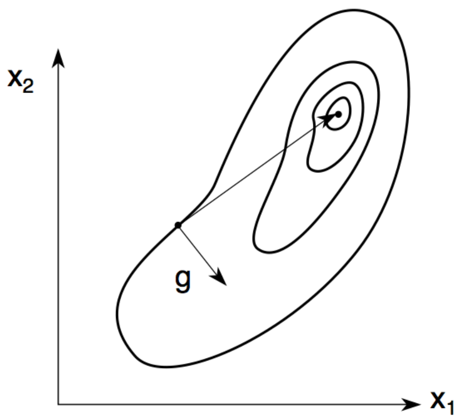
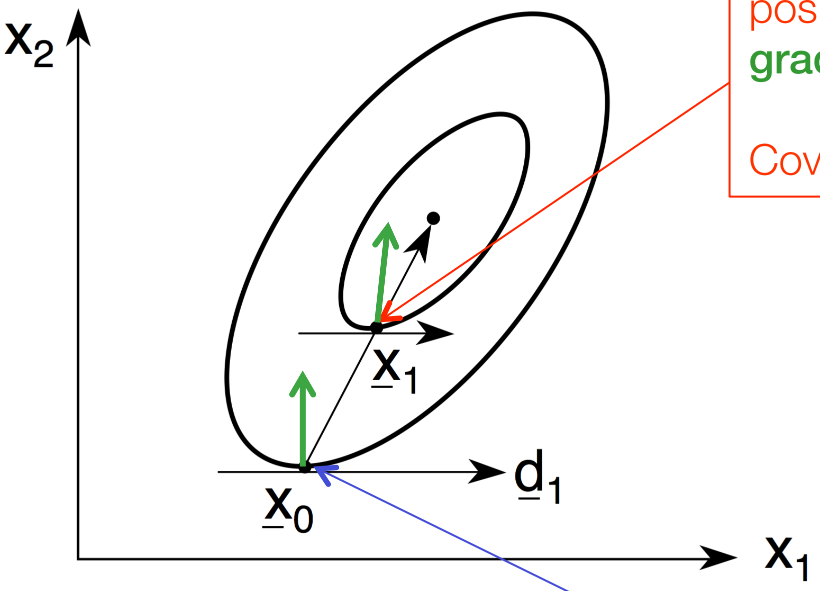


Fig. 9

Function minimization using the variable metric method

- Instead concept of ‘conjugate gradients’ that exploit knowledge of covariance information



position: $\mathbf{x_1 = x_0 - V_0 g_0}$
gradient: $\mathbf{g_1}$

Covariance: $\mathbf{V_1 = V_0 + f(V_0, x_0, x_1, g_0, g_1)}$

Davidon-Fletcher-Power rank 2 formula

$$\mathbf{V}_1 = \mathbf{V}_0 + \frac{\delta\delta^{\mathrm{T}}}{\delta^{\mathrm{T}}\gamma} - \frac{\mathbf{V}_0\gamma\gamma^{\mathrm{T}}\mathbf{V}_0}{\gamma^{\mathrm{T}}\mathbf{V}_0\gamma},$$

$$\delta = \mathbf{x}_1 - \mathbf{x} \quad \gamma = \mathbf{g}_1 - \mathbf{g}_0,$$

$$G(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1\,\partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1\,\partial x_n} \\ \frac{\partial^2 f}{\partial x_2\,\partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2\,\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n\,\partial x_1} & \frac{\partial^2 f}{\partial x_n\,\partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}.$$

position: x_0
gradient: g_0
Covariance: $V_0 = G^{-1} = I$

NB: If function is perfectly parabolic and initial V_0 is correct, convergence in one step!

Function minimization using the variable metric method

- Convergence criteria is based on 'estimated distance to minimum'
  - EDM 'estimated vertical distance to minimum' assuming parabolic function

$$\boxed{2 \cdot \mathrm{EDM} = \rho = g^T V g}$$

  - NB: Derives from general distance metric in non-Euclidian space

Covariant metric tensor

$$\Delta s^2 = \Delta x^T A \Delta x$$

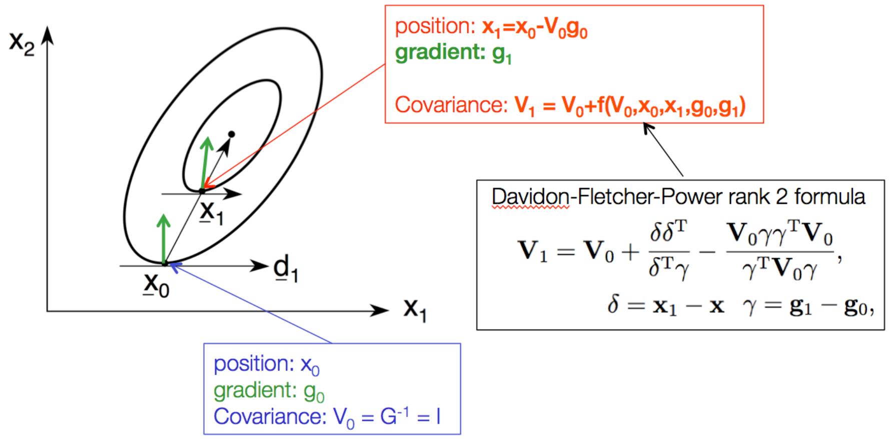
- Note that both minimization and convergence criteria depend on knowledge of covariance matrix

- There are 2 ways to calculate V
  1. From the Davidon-Fletcher-Power formula $\mathbf{V}_1 = \mathbf{V}_0 + \frac{\delta\delta^{\mathrm{T}}}{\delta^{\mathrm{T}}\gamma} - \frac{\mathbf{V}_0\gamma\gamma^{\mathrm{T}}\mathbf{V}_0}{\gamma^{\mathrm{T}}\mathbf{V}_0\gamma},$
  2. From the inversion of the Hessian matrix $\mathbf{V} = \mathbf{G}^{-1}$

Calculation of Hessian is expensive
(½N^2 likelihood evaluations)

MINUIT convergence

- After every VariableMetric step calculate EDM = ½$g^T V g$




```
VariableMetric: start iterating until Edm is < 0.001
VariableMetric: Initial state   - FCN =  -289.1204081677 Edm =      46.0713 NCalls =    1826
VariableMetric: Iteration #   1 - FCN =  -299.3073097602 Edm =      9.18415 NCalls =    2226
VariableMetric: Iteration #   2 - FCN =  -304.9468725143 Edm =      2.22698 NCalls =    2624
VariableMetric: Iteration #   3 - FCN =  -306.3323972775 Edm =      1.43793 NCalls =    3016
VariableMetric: Iteration #   4 - FCN =   -307.199970017 Edm =     0.615574 NCalls =    3410
VariableMetric: Iteration #   5 - FCN =  -307.6493784582 Edm =     0.352904 NCalls =    3804
VariableMetric: Iteration #   6 - FCN =  -307.8960954798 Edm =    0.0749124 NCalls =    4196
VariableMetric: Iteration #   7 - FCN =  -307.9549184882 Edm =    0.0498047 NCalls =    4588
VariableMetric: Iteration #   8 - FCN =  -308.0068371877 Edm =      0.03473 NCalls =    4980
VariableMetric: Iteration #   9 - FCN =  -308.0564661263 Edm =    0.0266955 NCalls =    5372
VariableMetric: Iteration #  10 - FCN =  -308.1092267909 Edm =     0.038622 NCalls =    5764
VariableMetric: Iteration #  11 - FCN =  -308.1547659161 Edm =    0.0290921 NCalls =    6156
VariableMetric: Iteration #  12 - FCN =  -308.1870210082 Edm =   0.00827767 NCalls =    6548
VariableMetric: Iteration #  13 - FCN =  -308.2008924182 Edm =    0.0034224 NCalls =    6940
VariableMetric: Iteration #  14 - FCN =  -308.2064790118 Edm =   0.00151676 NCalls =    7332
VariableMetric: Iteration #  15 - FCN =  -308.2090105175 Edm =   0.00106118 NCalls =    7724
VariableMetric: Iteration #  16 - FCN =  -308.2106535849 Edm =  0.000634155 NCalls =    8116
```

- Terminate VM procedure when EDM<0.001

MINUIT converge

```
VariableMetric: Iteration #  12 - FCN =  -308.1870210082 Edm =   0.00827767 NCalls =   6548
VariableMetric: Iteration #  13 - FCN =  -308.2008924182 Edm =    0.0034224 NCalls =   6940
VariableMetric: Iteration #  14 - FCN =  -308.2064790118 Edm =   0.00151676 NCalls =   7332
VariableMetric: Iteration #  15 - FCN =  -308.2090105175 Edm =   0.00106118 NCalls =   7724
VariableMetric: Iteration #  16 - FCN =  -308.2106535849 Edm =  0.000634155 NCalls =   8116
```

- (Terminate VM procedure when EDM<0.001)
  - Note that EDM up to here was calculated with V from DFP updater formula

$$\mathbf{V}_1 = \mathbf{V}_0 + \frac{\delta\delta^{\mathrm{T}}}{\delta^{\mathrm{T}}\gamma} - \frac{\mathbf{V}_0\gamma\gamma^{\mathrm{T}}\mathbf{V}_0}{\gamma^{\mathrm{T}}\mathbf{V}_0\gamma},$$

- From here on, procedure depends on ‘strategy code’
  - Code 0: terminate line search
  - Code 2: Recalculate **V** from $\mathbf{G}^{-1}$ (HESSE)
    if EDM(HESSE)>0.001 restart line search, else terminate
  - Code 1: If accuracy of $\mathbf{V}_n$ from DFP better than 5% terminate,
    else follow Code 2 procedure
- Strategy 1 is the default.

HESSE Convergence

- For smooth functions covariance estimates from HESSE are generally more accurate than those from Davidon-Fletcher-Powell but matrix inversion step is vulnerable to singularity issues
- Singularities detected with eigenvalue analysis of Hessian matrix G before matrix inversion
  - If 'smallest eigenvalue'/'largest eigenvalue' < 10^{-6} then matrix is declared 'not positive definite'
  - Note that happens for both negative *and* small eigenvalues
  - In that case an 'ad-hoc' term is added to the diagonal of the Hessian matrix to force it positive definite so that it can be inverted
- The 'adjusted' V from HESSE is then used to calculate the EDM
  - EDM estimate less reliable in this case, may cause MINUIT to endlessly go back to VariableMetric line search and eventually give up 'maximum number of calls exceeded'

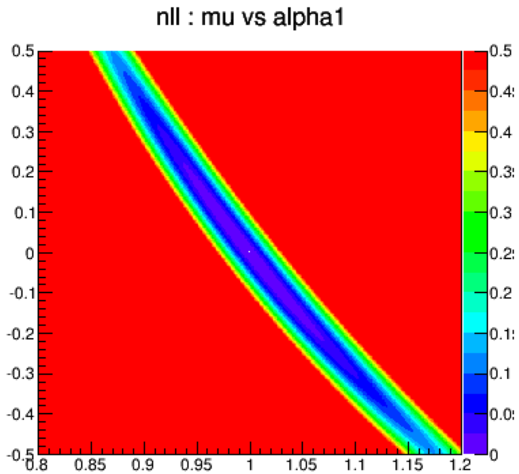
Likelihood models that cause MINUIT problems

- Example 1 – Strong correlations
  - Consider this simple likelihood model with one NP

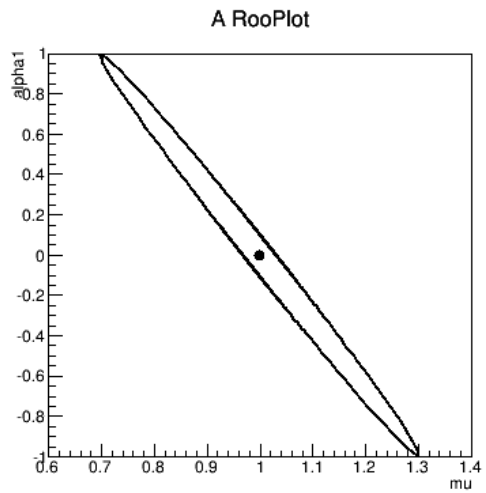
$$L_1(\mu,\alpha) = Poisson(N \mid \mu S(1+\tau\alpha))Gaussian(0 \mid \alpha,1)$$

  - What does the likelihood look like, e.g. for N=1000?

Scan of –log L(μ,α)



Error ellipse from V(μ,α) HESSE



ρ=0.9945

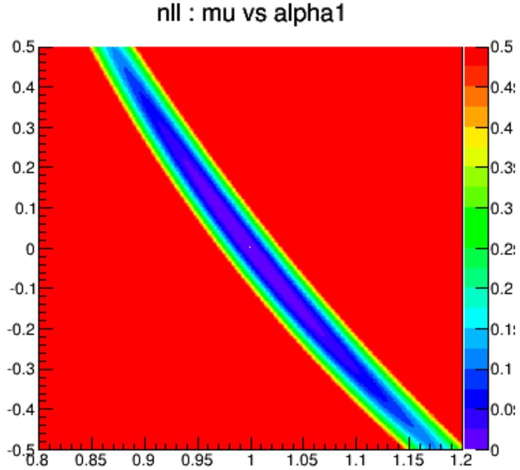
  - Strong correlations, but numerically feasible

Increasing the observed event count

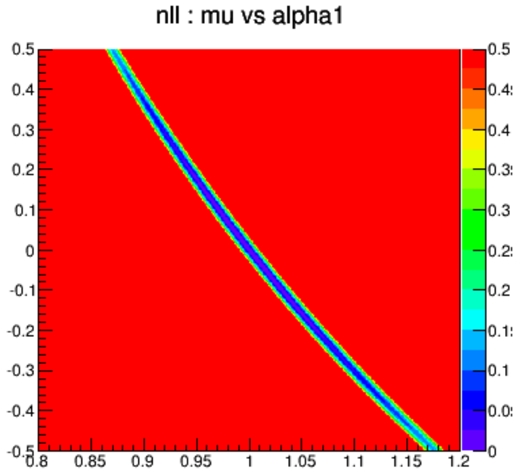
Vertical scale maximized at 0.5 units

Scan of $-\log L(\mu,\alpha)$

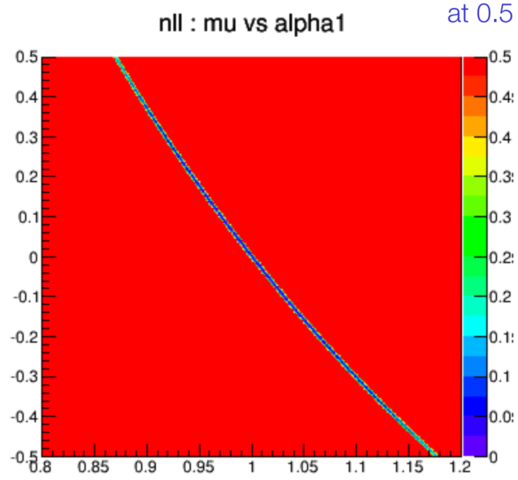
N=1000



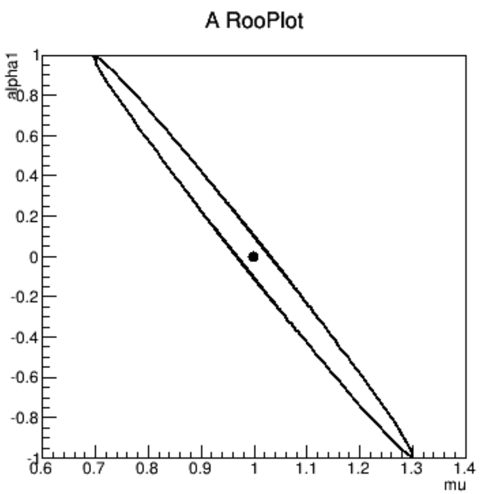
N=10.000



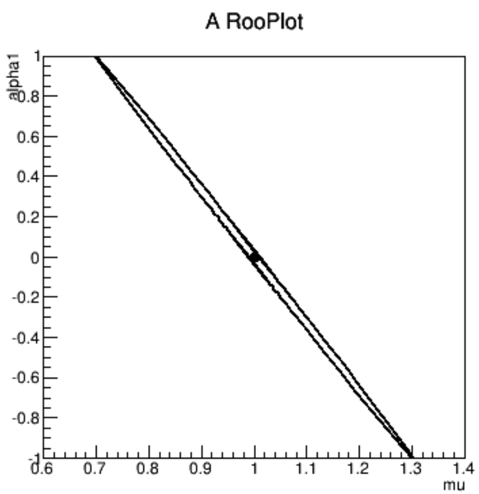
N=100.000



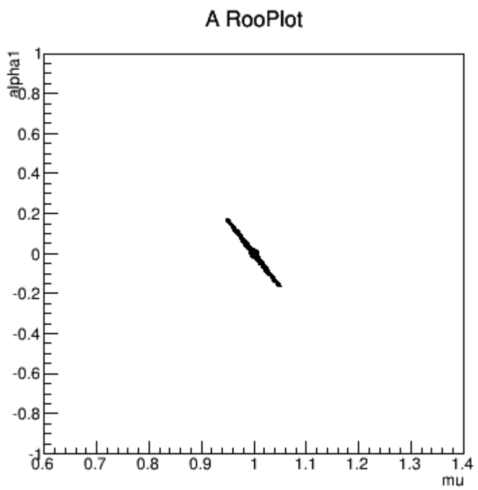
Error ellipse from $V(\mu,\alpha)$ HESSE



$\rho=-0.9945$



$\rho=-0.9995$

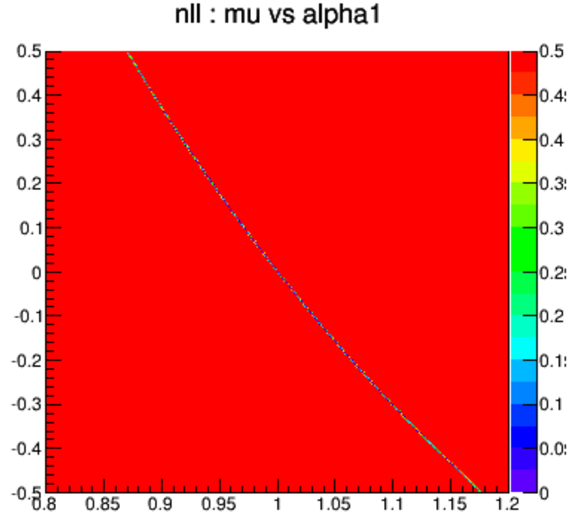


$\rho=-0.98$

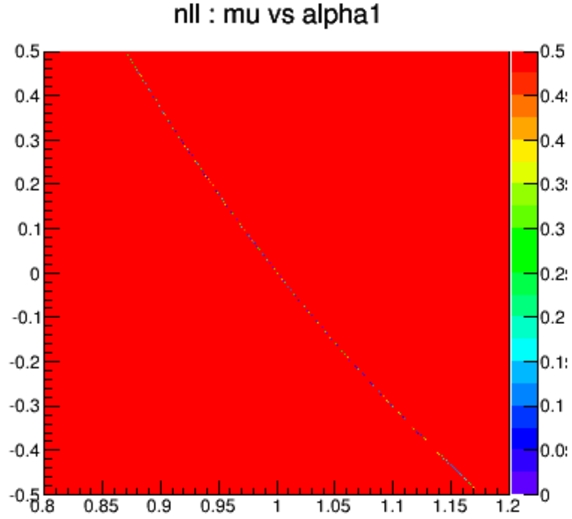
Increasing the observed event count

Scan of –log L(μ,α)

N=1.000.000

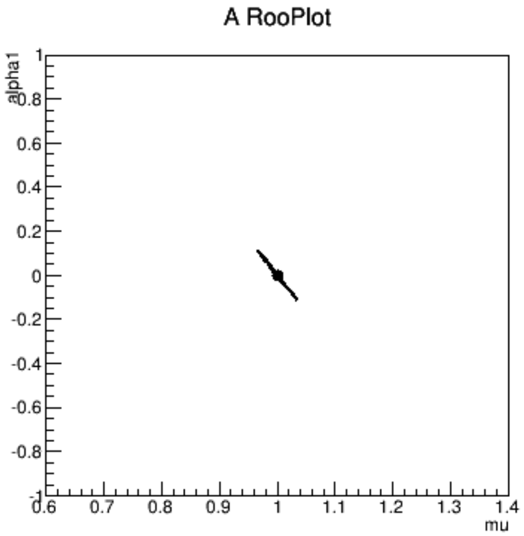


N=10.000.000

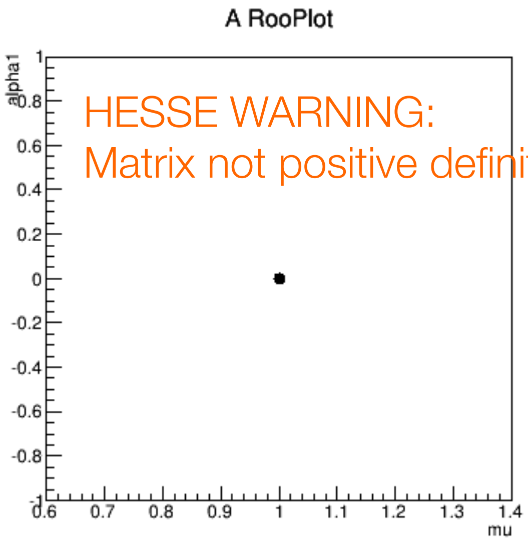


Vertical scale maximized at 0.5 units

Error ellipse from V(μ,α) HESSE



ρ=-0.9996



ρ=-0.998

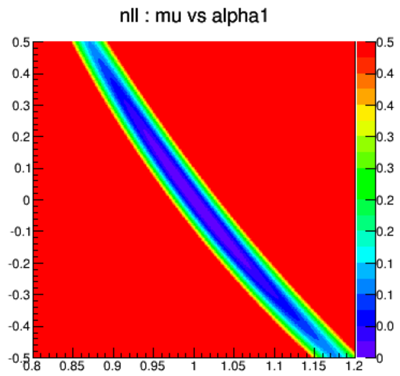
Likelihood models that cause MINUIT problems

- Example 2 – Hidden strong correlations
  - Consider this trivial extension of the previous example with 2 NPs

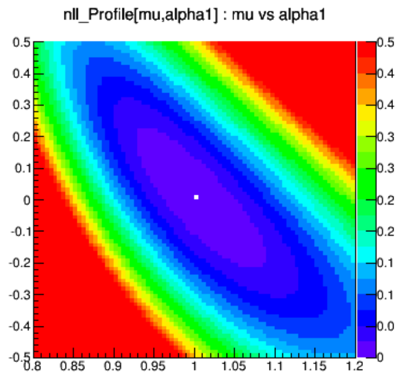
$$L_2(\mu,\alpha_1,\alpha_2) = Poisson(N \mid \mu S(1+\tau_1\alpha_1+\tau_2\alpha_2))Gauss(0 \mid \alpha_1,1)Gauss(0 \mid \alpha_2,1)$$

  - Underlying scenario: two (independent) sources of systematic uncertainty that have a similar effect on the physics measurement
  - What does (profile) likelihood look like for various S?

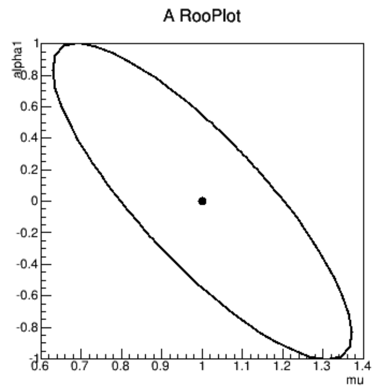
$-\log L(\mu,\alpha_1,\underline{\hat{\alpha}_2})$



$-\log L(\mu,\alpha_1,\underline{\hat{\hat{\alpha}}_2(\alpha_1,\mu)})$



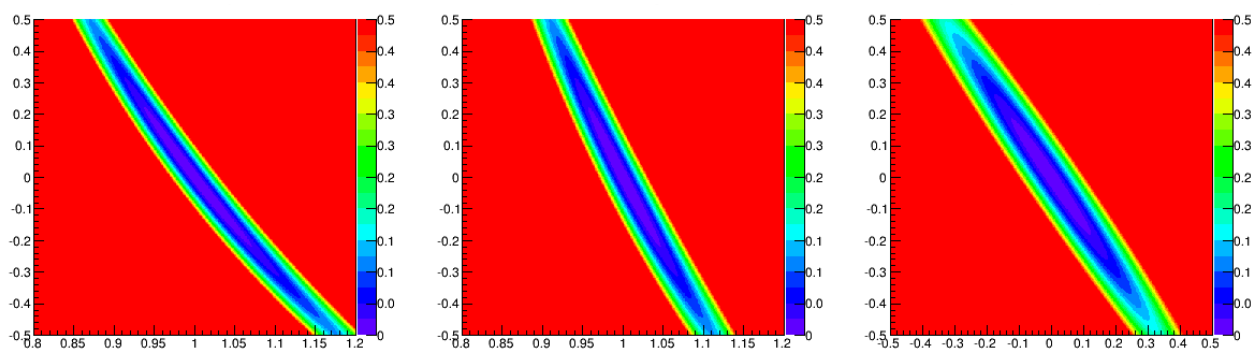
Error ellipse V(μ,α) HESSE



-logL(μ,α_1,α_2) – 1000 events

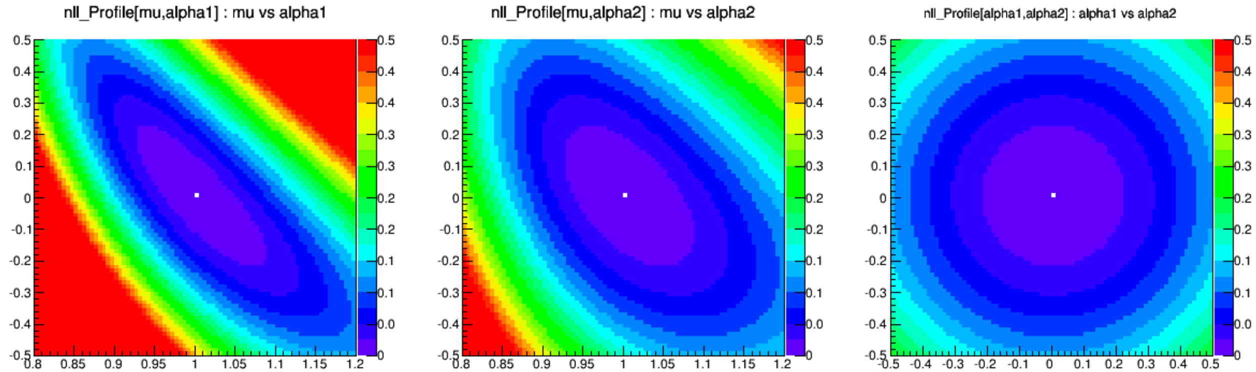
Slice in -logL

$$-\log L(a,b,\hat{c})$$

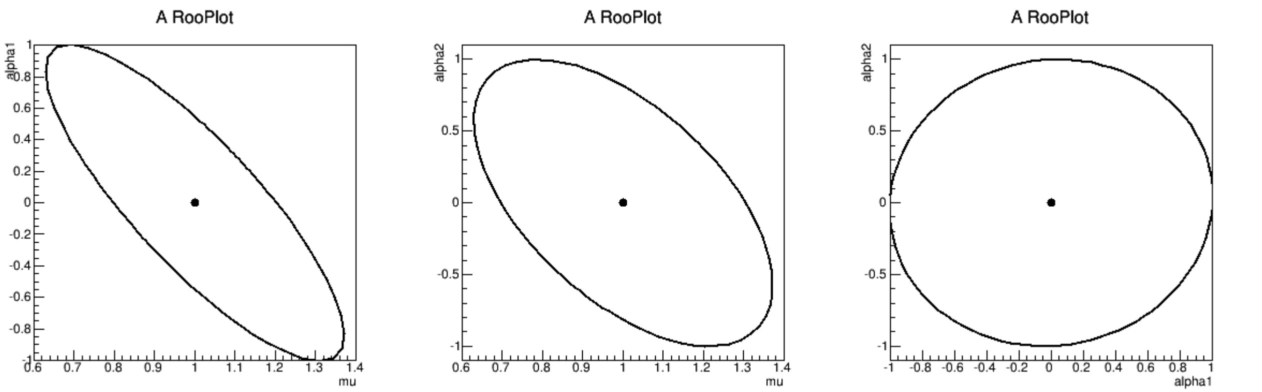


Profile likelihood

$$-\log L(a,b,\hat{\hat{c}}(a,b))$$



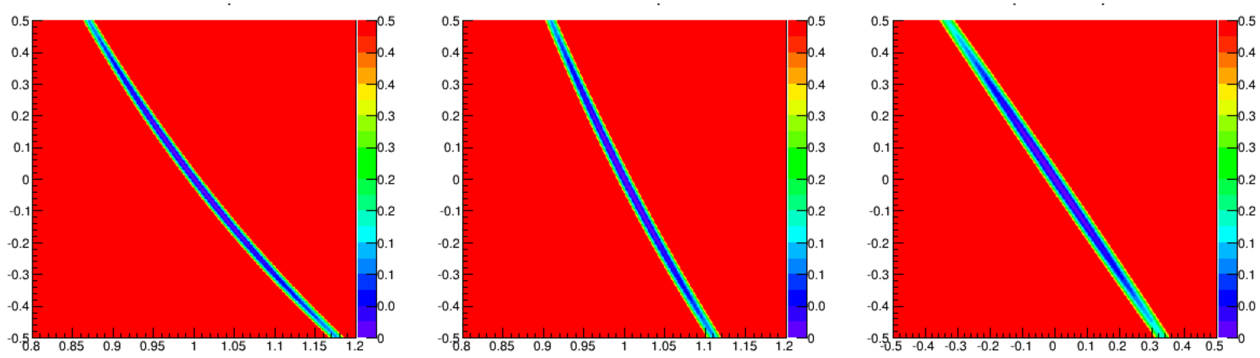
Error ellipse
from HESSE



-logL(μ,α_1,α_2) – 10.000 events

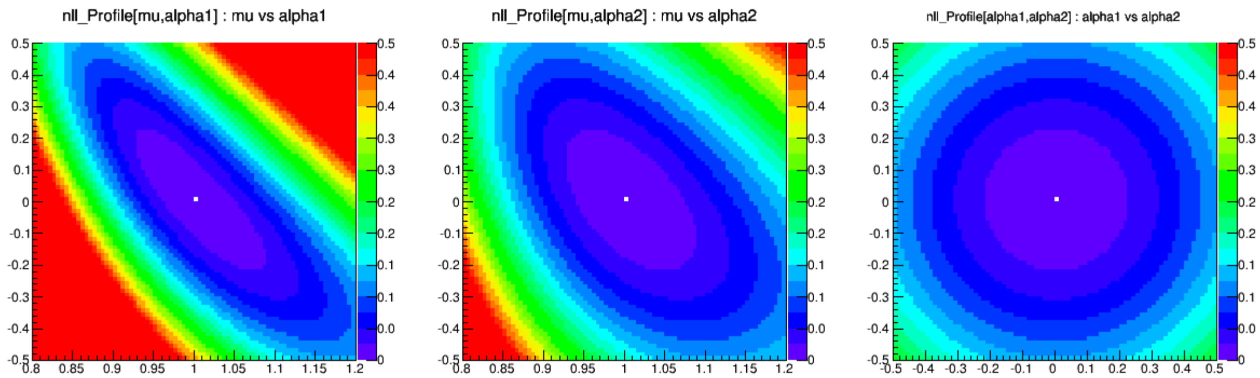
Slice in -logL

$$-\log L(a,b,\hat{c})$$



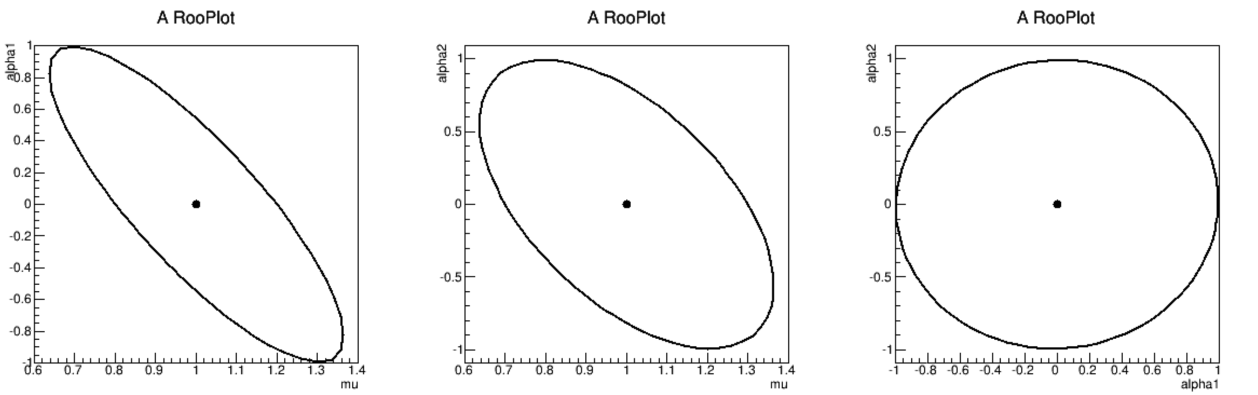
Profile likelihood

$$-\log L(a,b,\hat{\hat{c}}(a,b))$$



Note that PLL contours don't change between 1K and 10k!

Error ellipse from HESSE



-logL(μ,α_1,α_2) – 100.000 events

Slice in -logL

$$-\log L(a,b,\hat{c})$$

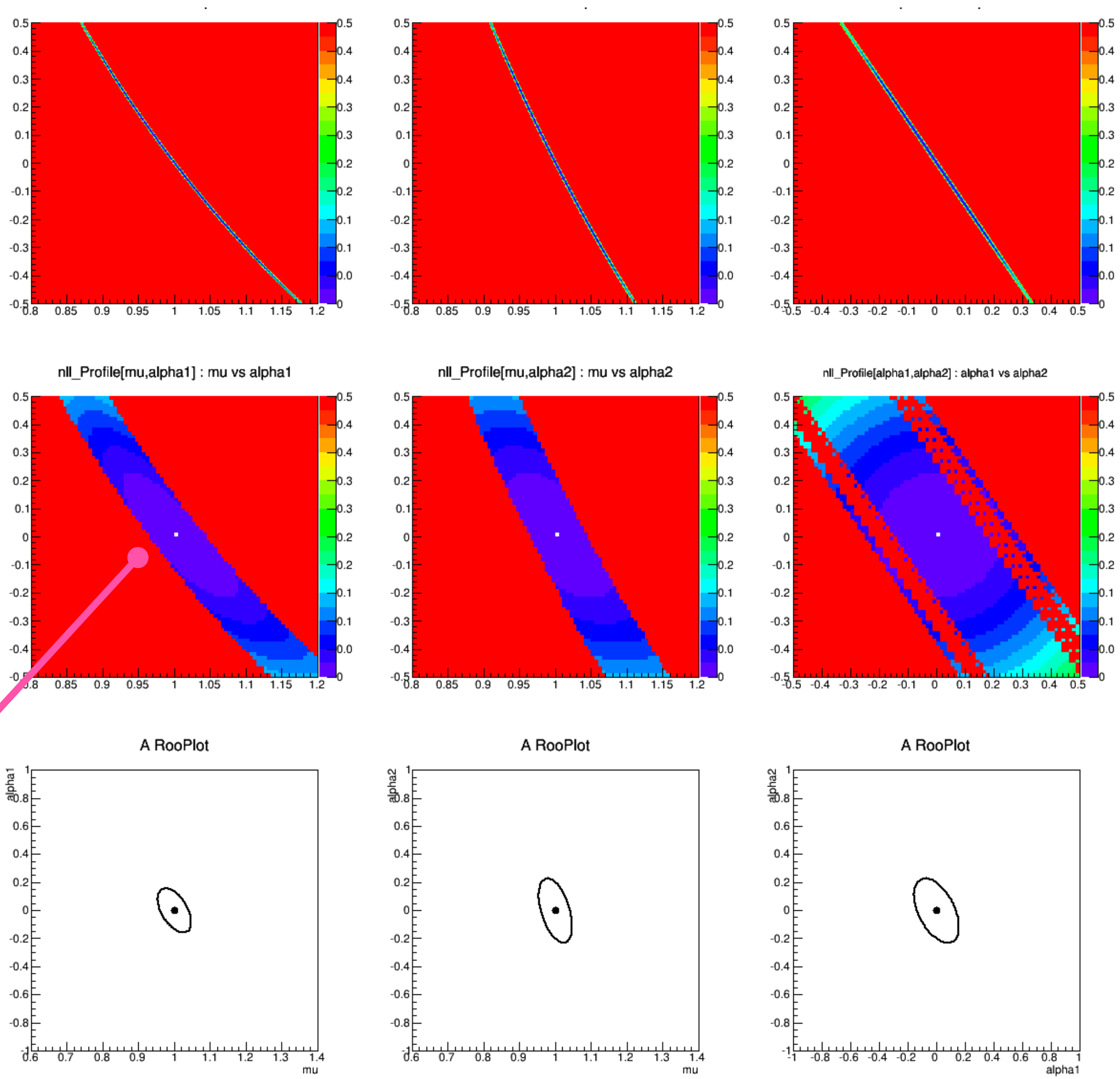
Profile likelihood

$$-\log L(a,b,\hat{\hat{c}}(a,b))$$

Note that PLL contours don't change between 10K and 100k close to min.!

(but onset of fit failures further away…)

Error ellipse from HESSE



-logL(μ,α_1,α_2) – 1.000.000 events

Slice in -logL

$$-\log L(a,b,\hat{c})$$

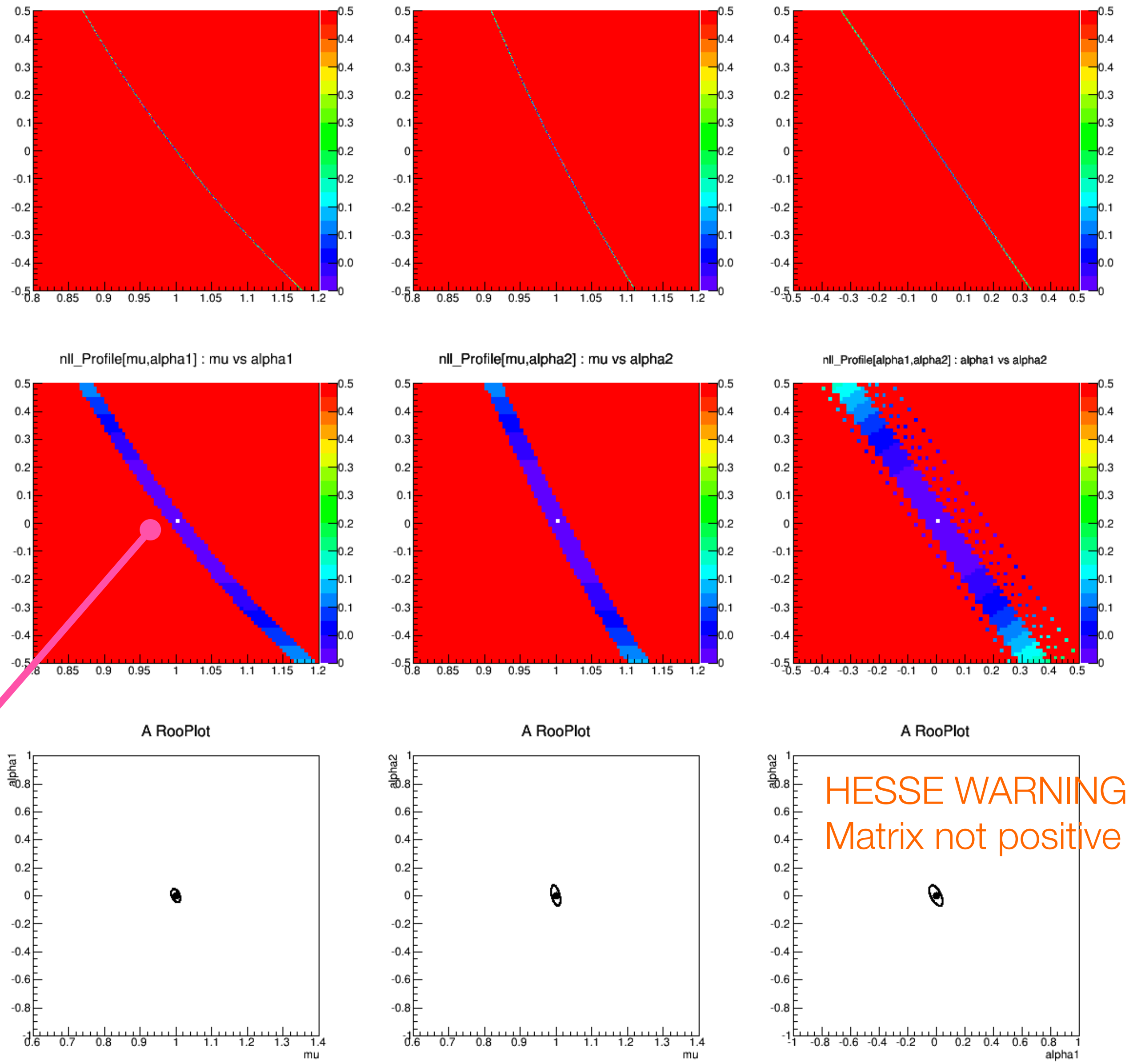
Profile likelihood

$$-\log L(a,b,\hat{\hat{c}}(a,b))$$

Note that PLL contours don't change between 100K and 1M close to min.!

(but further increase of fit failures further away…)

Error ellipse from HESSE



HESSE WARNING: Matrix not positive definit

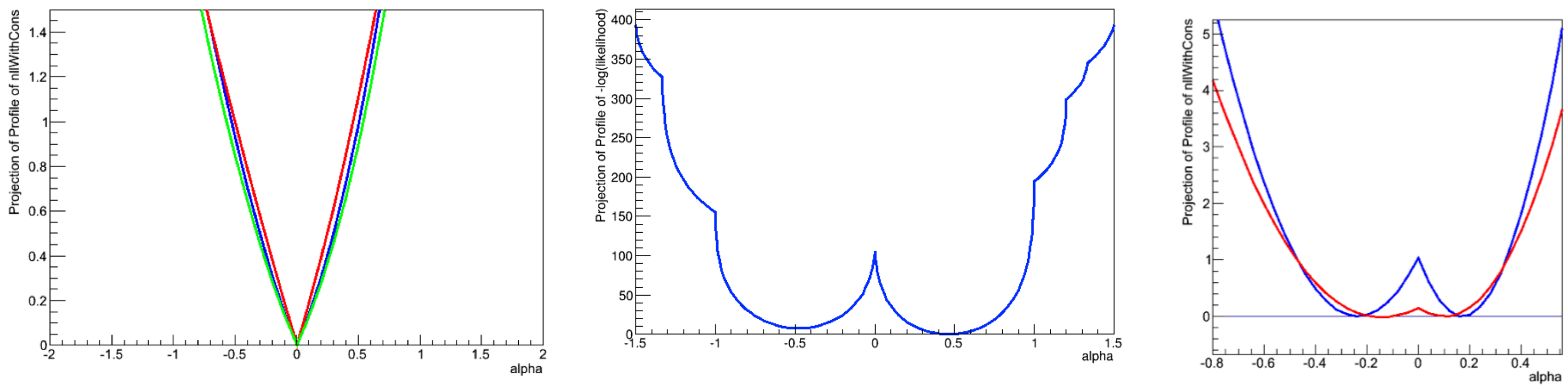
Conclusions on strong correlations

- MINUIT can handle strong correlations very well, but at some point algorithm breaks down
  - Notably HESSE will fail when ratio of weakest-to-strongest eigenvalue < 10^{-6}
- Diagnostic of the existence of strong correlations can be difficult
  - In simple models (Ex 1) this is reflected correlation coefficients
  - In more complex models (Ex 2) this may not show at all in the correlation coefficients because strong 'N-point correlations' may still project out to modest 2-point correlations (i.e. the usual Pearson correlation coefficients)
  - Better diagnostic tools is eigenvalues of Hessian matrix before inversion, but not (yet) available in Minuit2 [I am discussing this with ROOT team]
- Solution: consider to simplify model:
  - If two NPs represent conceptually distinct systematic uncertainties, but their effect on the likelihood is virtually identical, then there is effectively a redundant degree of freedom. You can eliminate one

Other likelihood pathologies

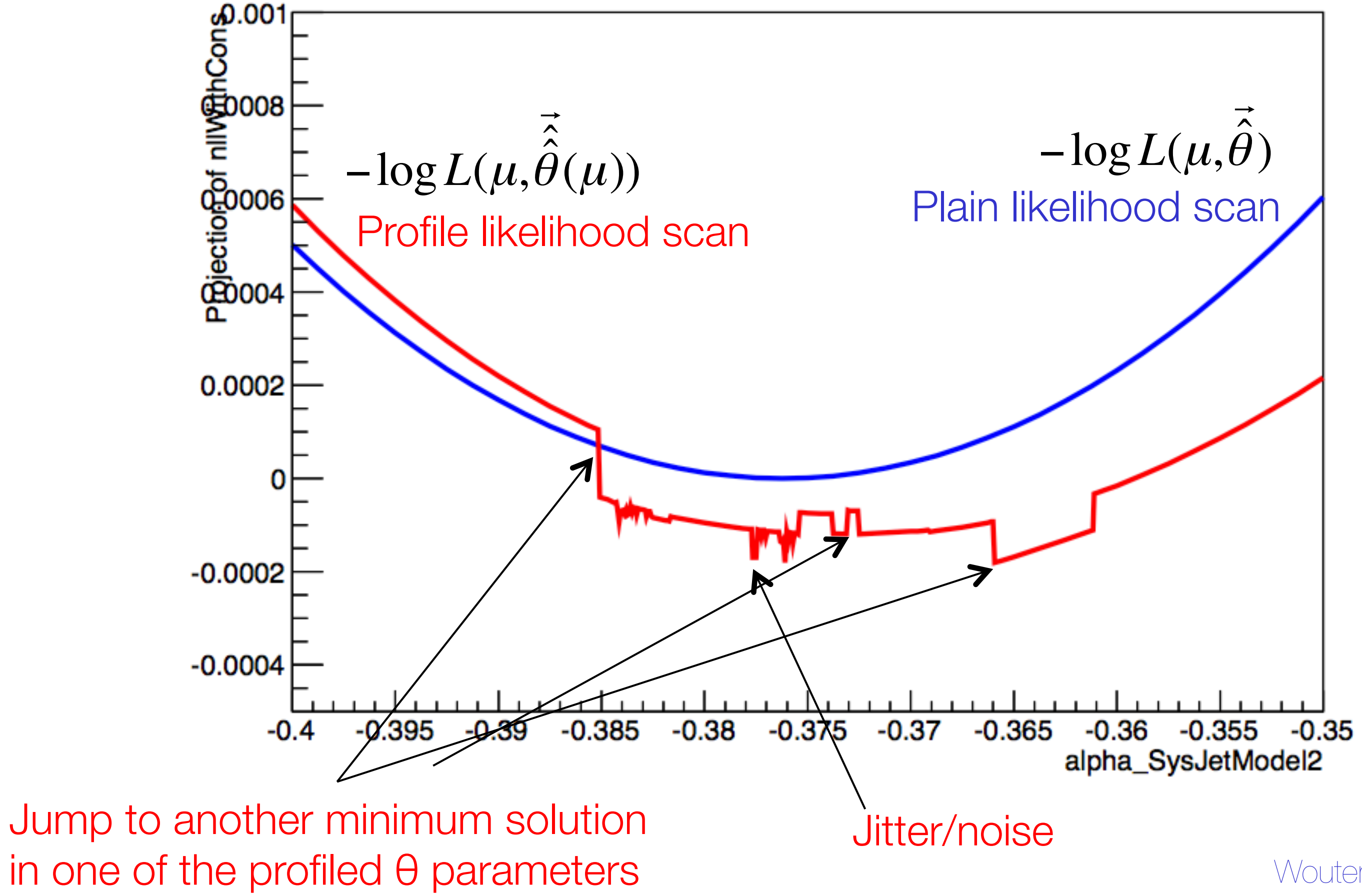
- Template morphing algorithms can introduce various other pathologies in the likelihood that cause MINUIT to fail
  - We've already seen some of them

- Kinks & Multiple minima
  - Caused by (among others) template morphing with piece-wise linear interpolation and morphing of (low-statistics) template distributions where MC statistical effects are larger than systematic effect



Other likelihood pathologies

- Effects of likelihood pathologies
  - Numerical noise and ‘jumping’ of profile likelihoods
  - Example NP (profile) likelihood scan of an ATLAS Higgs trial model



Other likelihood pathologies

- Another effect of likelihood pathologies is that calculation of derivatives and notably the Hessian from either FDP or HESSEmatrix become inaccurate
  - Slows down minimization
  - Can blow up EDM calculation → no convergence
- Red flags: EDM estimates that don't decrease ~monotonically
  - Only possible in Minuit2 (Minuit1 does not report EDM per step)

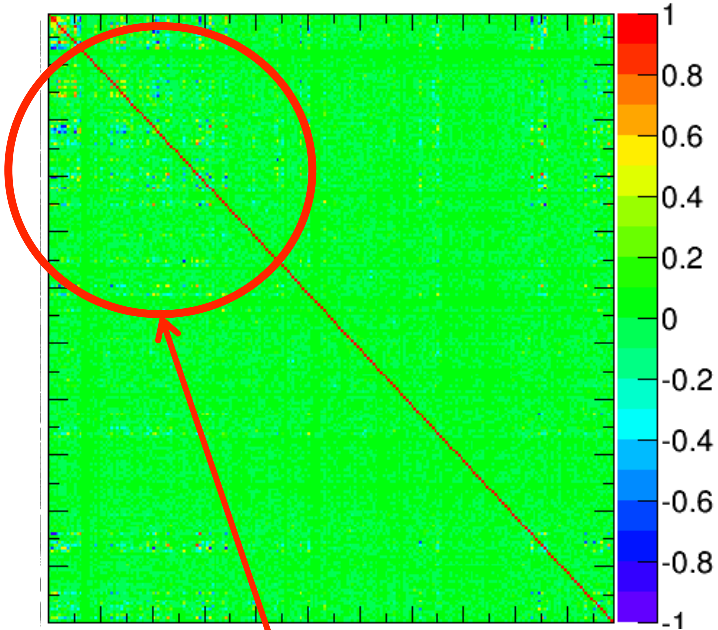
```
VariableMetric: start iterating until Edm is < 0.001
VariableMetric: Initial state   - FCN =  -289.1204081677 Edm =      46.0713 NCalls =   1826
VariableMetric: Iteration #   1 - FCN =  -299.3073097602 Edm =      9.18415 NCalls =   2226
VariableMetric: Iteration #   2 - FCN =  -304.9468725143 Edm =      2.22698 NCalls =   2624
VariableMetric: Iteration #   3 - FCN =  -306.3323972775 Edm =      1.43793 NCalls =   3016
VariableMetric: Iteration #   4 - FCN =   -307.199970017 Edm =     0.615574 NCalls =   3410
VariableMetric: Iteration #   5 - FCN =  -307.6493784582 Edm =     0.352904 NCalls =   3804
VariableMetric: Iteration #   6 - FCN =  -307.8960954798 Edm =    0.0749124 NCalls =   4196
VariableMetric: Iteration #   7 - FCN =  -307.9549184882 Edm =     0.298047 NCalls =   4588
VariableMetric: Iteration #   8 - FCN =  -308.0068371877 Edm =      3.40473 NCalls =   4980
```

- Solutions: simplify model: eliminate nuisance parameters that suffer from dominant MC statistical effects (causing multiple minima, kinks etc…)

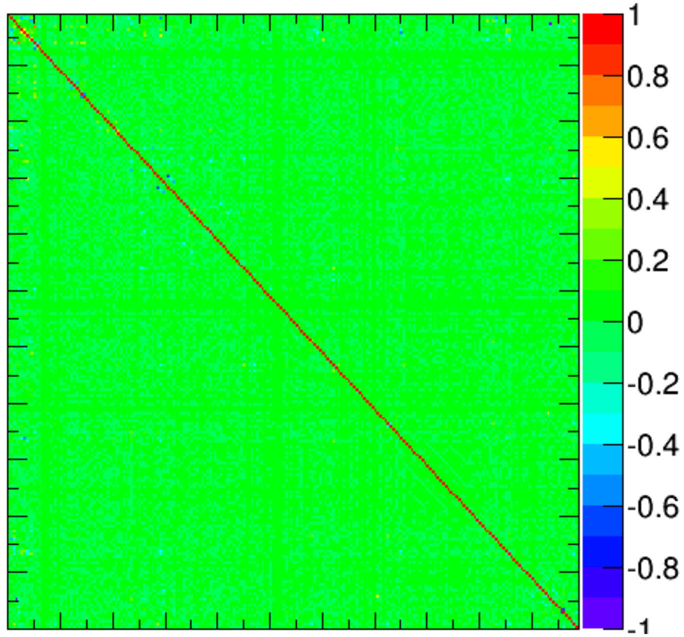
Other likelihood pathologies

- Note that pathologies can affect calculation of V via iterative DFP updating and Hessian inversion differently
- A real-life example of complex likelihood fit where DFP estimate is strongly affected by likelihood pathologies

V from Davidon-Fletcher-Powell



V from inversion of Hessian



Many spurious large correlations

- But other likelihood pathologies can affect Hessian inversion more

Summary

- A variety of pathological features in likelihood models can interfere with minimization
  - Strong correlations
  - Kinks
  - Multiple minima
  - ‘Forbidden regions’ where likelihood is not defined
- Problems affect various steps of the minimization process
  - Understanding these effects requires basic understanding of the minimization algorithms and strategies
- Solutions usually involve simplifications of models

8 Diagnostics II: Overconstraining & choices in modeling parametrization

Understanding profile likelihood fits

- The previous section discussed technical diagnostics of profile likelihood fits
  - "Why doesn't my fit converge"?

- The next level of diagnostics is on the interpretation level
  - "Do my fit results make sense"?
  - "What part of the likelihood model is measuring what?"

Role reversal of physics and subsidiary measurements

- As mention in Section 3, full (profile) likelihood treats physics and subsidiary measurement on equal footing

$$L(N,0 \mid s,\alpha) = \underbrace{Poisson(N \mid s+b(1+0.1\alpha))}_{\text{Physics measurement}} \cdot \underbrace{Gauss(0 \mid \alpha,1)}_{\text{Subsidiary measurement}}$$

- Our mental picture:

  "measures s"

  "measures α"

  "dependence on α weakens inference on s"

- Is this picture (always) correct?

Understanding your model – what constrains your NP

- The answer is no – not always! Your physics measurement may in some circumstances constrain α better than your subsidiary measurement.
- Doesn't happen in Poisson counting example
  - Physics likelihood has no information to distinguish effect of s from effect of α

$$L(N,0 \mid s,\alpha) = \underbrace{Poisson(N \mid s + b(1+0.1\alpha))}_{\text{Physics measurement}} \cdot \underbrace{Gauss(0 \mid \alpha,1)}_{\text{Subsidiary measurement}}$$

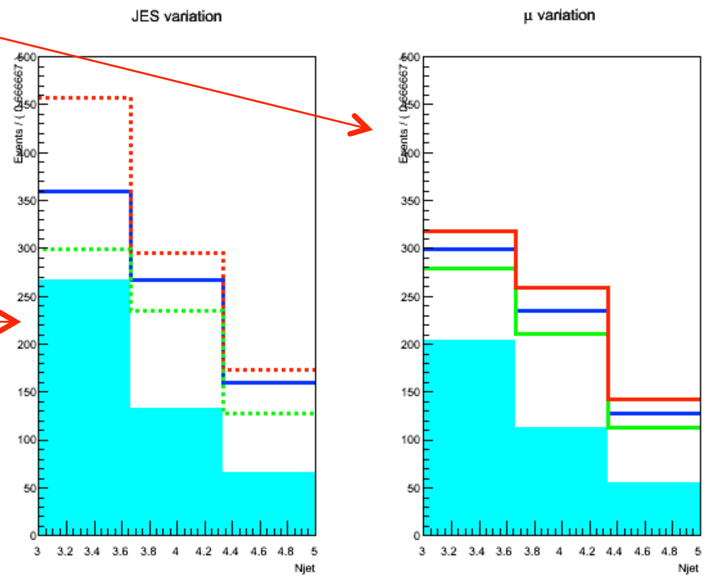
- But if physics measurement is based on a distribution or comprises multiple distributions this is well possible

Understanding your model – what constrains your NP

- A case study – measuring jet multiplicity
  - Physics observable of interest is a jet multiplicity spectrum [3j,4j,5j] after an (unspecified) p_T cut.
  - Describe data with sum of signal (mildly peaking at 4j) and a single background (exponentially falling in nj).

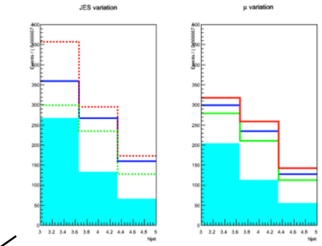
$$L(\vec{N}\,|\,\mu,\alpha_{JES}) = \prod_{i=3,4,5} Poisson(N_i\,|\,(\mu\cdot\tilde{s}_i\cdot+\tilde{b}_i)\cdot r_s(\alpha_{JES})))\cdot Gauss(0\,|\,\alpha_{JES},1)$$

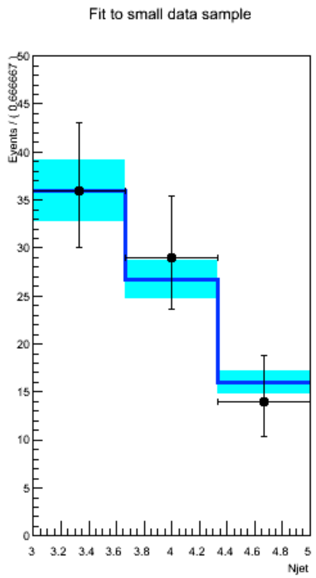
  - POI is signal strength modifier μ.
  - Jet Energy Scale is the leading systematic uncertainty
    - JES strongly affects jet multiplicity after a p_T cut,
    - Effect modeled by response function $r_s(\alpha)$
    - Magnitude of uncertainty on α constrained by subsidiary measurement



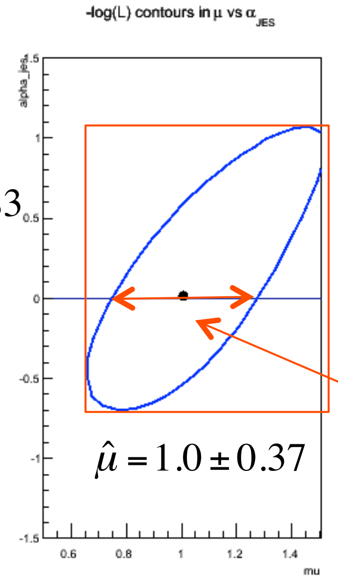
Understanding your model – what constrains your NP

- Now measure (μ,α) from data – 80 events





$\hat{\alpha} = 0.01 \pm 0.83$



$\hat{\mu} = 1.0 \pm 0.37$

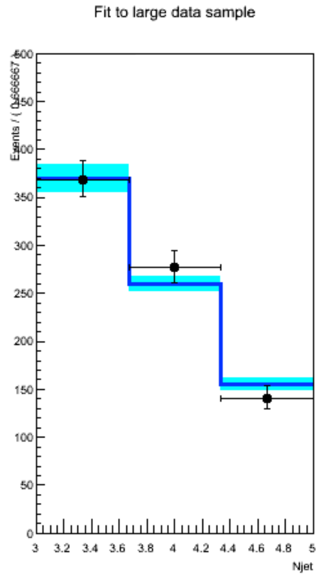
Estimators of μ, α correlated due to similar response in physics measurement

Uncertainty on μ without effect of JES

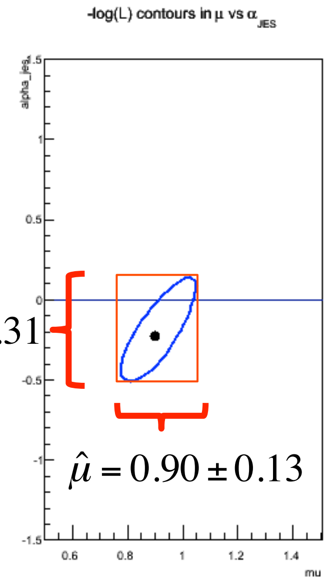
- Is this fit OK?
  - Effect of JES uncertainty propagated in to μ via response modeling in likelihood. Increases total uncertainty by about a factor of 2
  - Estimated uncertainty on α is not precisely 1, as one would expect from unit Gaussian subsidiary measurement…

Understanding your model – what constrains your NP

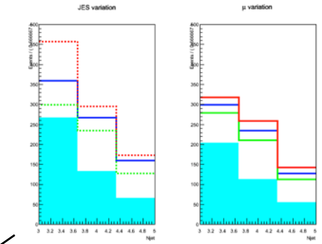
- The next year – 10x more data (800 events) repeat measurement with same model



$\hat{\alpha} = -0.23 \pm 0.31$



$\hat{\mu} = 0.90 \pm 0.13$



Estimators of μ, α correlated due to similar response in physics measurement

- Is this fit OK?
  - Uncertainty of JES NP *much reduced* w.r.t. subsidiary meas. (α = 0 ± 1)
  - Because the physics likelihood can measure it better than the subsidiary measurement (the effect of μ, α are sufficiently distinct that both can be constrained at high precision)

Understanding your model – what constrains your NP

- Is it OK if the physics measurement constrains NP associated with a systematic uncertainty better than the designated subsidiary measurement?
  - From the statisticians point of view: no problem, simply a product of two likelihood that are treated on equal footing 'simultaneous measurement'
  - From physicists point of view? Measurement is only valid is model is valid.
- Is the probability model of the physics measurement valid?

$$L(\vec{N}\,|\,\mu,\alpha_{JES}) = \prod_{i=3,4,5} Poisson(N_i\,|\,(\mu\cdot\tilde{s}_i\cdot+\tilde{b}_i)\cdot r_s(\alpha_{JES})))\cdot Gauss(0\,|\,\alpha_{JES},1)$$

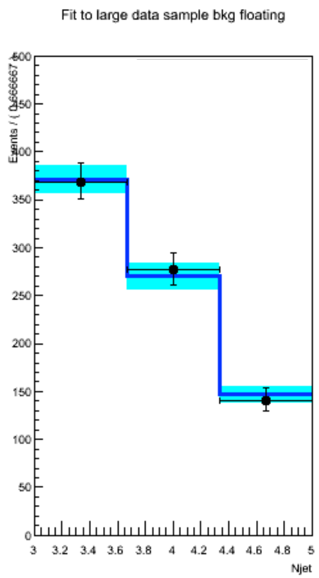
- Reasons for concern
  - Incomplete modeling of systematic uncertainties,
  - Or more generally, model insufficiently detailed

Understanding your model – what constrains your NP

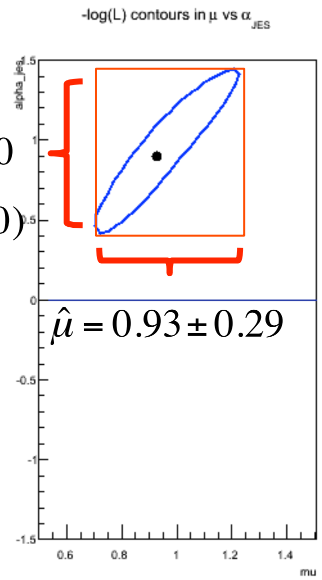
- What did we overlook in the example model?
  - The background rate has no uncertainty!
  - Insert modeling of background uncertainty

$$L(\vec{N}\,|\,\mu,\alpha_{JES},\alpha_{bkg}) = \prod_{i=3,4,5} Poisson(N_i\,|\,(\mu\cdot\tilde{s}_i\cdot+\underbrace{\tilde{b}_i\cdot r_b(\alpha_{bkg})}_{\text{Background rate response function}})\cdot r_s(\alpha_{JES}))) \cdot \underbrace{Gauss(0\,|\,\alpha_{JES},1)\cdot Gauss(0\,|\,\alpha_{bkg},1)}_{\text{Background rate subsidiary measurement}}$$

- With improved model accuracy estimated uncertainty on both α_{JES}, μ goes up again…
  - Inference weakened by new degree of freedom α_{bkg}
  - NB α_{JES} estimate still deviates a bit from normal distribution estimate…



$$\hat{\alpha}_{JES} = 0.90 \pm 0.70$$
$$(\hat{\alpha}_{bkg} = 1.36 \pm 0.20)$$



$$\hat{\mu} = 0.93 \pm 0.29$$

Understanding your model – what constrains your NP

- Lesson learned: if probability model of a physics measurement is insufficiently detailed (i.e. flexible) you can *underestimate* uncertainties
- Normalized subsidiary measurement provide an excellent diagnostic tool
  - Whenever estimates of a NP associated with unit Gaussian subsidiary measurement deviate from $\alpha = 0 \pm 1$then physics measurement is constraining or biases this NP.
  - Always inspect all NPs of your fit for such signs
- Is 'over-constraining' of systematics NPs always bad?
  - No, sometimes there are good arguments why a physics measurement can measure a systematic uncertainty better than a dedicated calibration measurement (that is represented by the subsidiary measurement)
  - Example: in sample of reconstructed hadronic top quarks $t \rightarrow bW(qq)$, the pair of light jets should always have m(jj)=mW. For this special sample of jets it will possible to calibrate the JES better than with generic calibration measurement

Commonly heard arguments in discussion on over-constraining

- Overconstraining of a certain systematic is OK "because this is what the data tell us"
  - It is what the data tells you *under the hypothesis that your model is correct*. The problem is usually in the latter condition
- "The parameter α_{JES} should not be interpreted as Jet Energy Scale uncertainty provided by the jet calibration group"
  - A systematic uncertainty is always combination of response prescription and one or more nuisance parameters uncertainties.
  - If you implement the response prescription of the systematic, then the NP in your model really is the same as the prescriptions uncertainty
- "My estimate of $\alpha_{JES} = 0 \pm 0.4$ doesn't mean that the 'real' Jet Energy Scale systematic is reduced from 5% to 2%
  - It certainly means that in your analysis a 2% JES uncertainty is propagated to the POI instead of the "official" 5%.
  - One can argue that the 5% shouldn't apply because your sample is special and can be calibrated better by a clever model, but this is a physics argument that should be documented with evidence for that (e.g. argument JES in $t \rightarrow bW(qq)$ decays)

Dealing with over-constrained systematic NPs

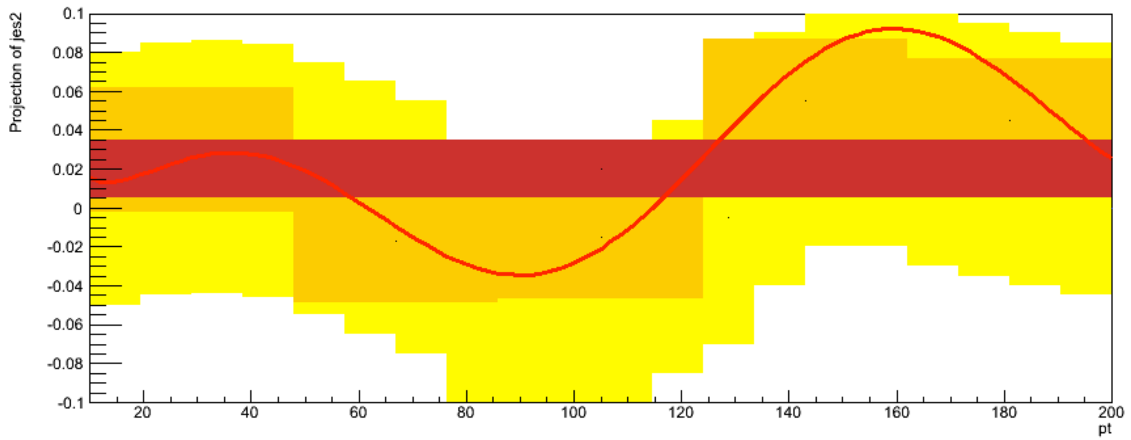
- Step 1 – Diagnostics
  - Always inspect nuisance parameters in your fit for signs of over-constraining
- Step 2 – Analyze
  - Are there systematic uncertainties overlooked in the construction of the likelihood that introduce unwarranted physics assumption in model that ML estimator exploits to constrain models?
  - Is your systematic uncertainty conceptually covered by a single nuisance parameter? do you perhaps need more NPs?
  - In case the physics likelihood comprises multiple samples, do you assume fully correlated responses functions, whereas sample composition should conceptually allow for some degree of decorrelation?
- Step 3 – Solution
  - If over-constraining is analyzed to be the result of inaccurate modeling, improve model structure, add new NPs, decompose NPs in different ways to reflect sample correlations
  - If constraint from physics is believed to be document studies as part of your physics analysis

Dealing with over-constraining – introducing more NPs

- Some systematic uncertainties are not captured well by one nuisance parameter.
- Example Jet Energy Scale
  - Statement "the JES uncertainty is 5% for all jets" does not necessarily imply that the calibration of all jets can be modeled with a single NP.
  - A single NP implies that the calibration can only be coherently off among all jets. Such an assumption allows, for example, to precisely constrain JES with a high-statistics sample of low-p_T jets, and then transport that reduced uncertainty to high-p_T jets, using the calibration scale coherency encoded in the model
  - In reality correlation between the energy scale of low-pT and high-pT jets is controlled by the detector design and calibration procedure and is likely a lot more complicated → Invalid modeling of systematic uncertainties often a result of 'own interpretation' of imprecisely formulated systematic prescription.
  - Besides this, a calibration may have multiple sources of uncertainty that were lumped together in a prescription (calibration measurements, simulation assumptions, sample-dependent effects) that would need to be individually modeled

Dealing with over-constraining – introducing more NPs

- The need for model accuracy increases with measurement precision
- Using again JES calibration example.
  - Suppose true JES data/MC miscalibration is p_T dependent (unknown)
  - How p_T bins (i.e. NPs) do you need to capture true JES uncertainty?
  - At 8% accuracy – model with 1 NP covers entire range within stated precision
  - Overconstraining 1-NP model 2% precision leads to modeling inaccuracies
  - Need to divide in p_T bins (more NPs), NP uncertainty will rise as less calibration data is available per bin

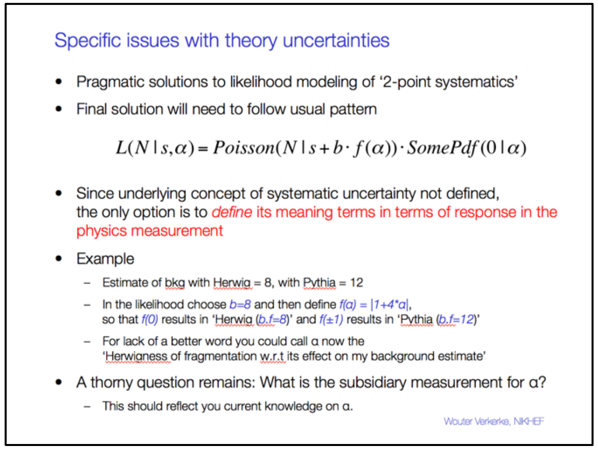


  - Optimal binning depends on experimental precision and *expected* variability of calibration (which you might infer from detector design)
- Response models that are valid at some resolution are not necessarily still valid at a better resolution
  - A good reason to be cautious with over-constraining of NPs in physics measurements

Dealing with over-constraining – Theory uncertainties

- Over-constraining of theory uncertainties in physics measurements has different set of issues than for detector uncertainties
- Different: In principle it is the goal of physics measurements to constrain theory uncertainties
  - So role of physics measurement and subsidiary measurement are not symmetric: the latter quantifies some 'degree of belief' that is not based on an experimental measurement.
  - Likelihood of physics measurement constitutes an experimental measurement and is in principle preferred over 'belief'
  - But question remains if physics likelihood was well designed to constrain *this particular theory uncertainty.*
- Same: response function and set of NPs must be able to accurately capture underlying systematic effect.
  - Sometimes easy, e.g. 'renormalization scale' has well-defined meaning in a given theoretical model and a clearly identifiable single associated parameter
  - Sometimes hard, e.g. 'Pythia vs Herwig'. Not clear what it means or how many degrees of freedom underlying model has.

Dealing with 'two-point' uncertainties



Specific issues with theory uncertainties

- Pragmatic solutions to likelihood modeling of '2-point systematics'
- Final solution will need to follow usual pattern

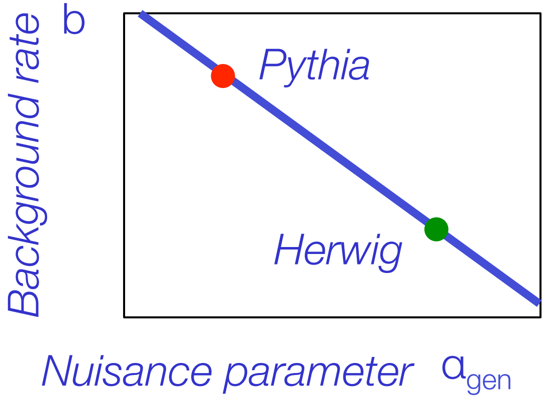
$$L(N|s,\alpha) = Poisson(N|s + b \cdot f(\alpha)) \cdot SomePdf(0|\alpha)$$

- Since underlying concept of systematic uncertainty not defined, the only option is to *define* its meaning terms in terms of response in the physics measurement
- Example
  - Estimate of bkg with Herwig = 8, with Pythia = 12
  - In the likelihood choose *b=8* and then define *f(α) = |1+4*α|*, so that *f(0)* results in 'Herwig (*b.f=8*)' and *f(±1)* results in 'Pythia (*b.f=12*)'
  - For lack of a better word you could call α now the 'Herwigness of fragmentation w.r.t its effect on my background estimate'
- A thorny question remains: What is the subsidiary measurement for α?
  - This should reflect you current knowledge on α.

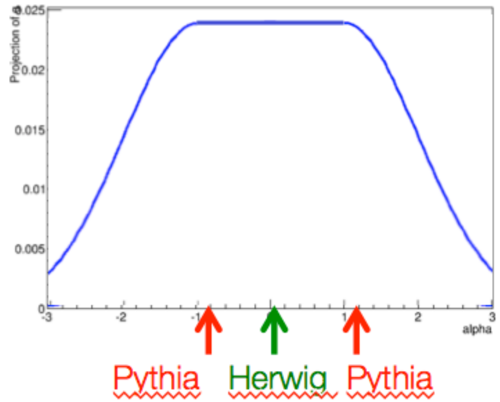
Wouter Verkerke, NIKHEF

- In discussion of rate systematics in Section 3 it was mentioned that 'two-point systematics' can always be effectively represented with an interpolation strategy

Response model



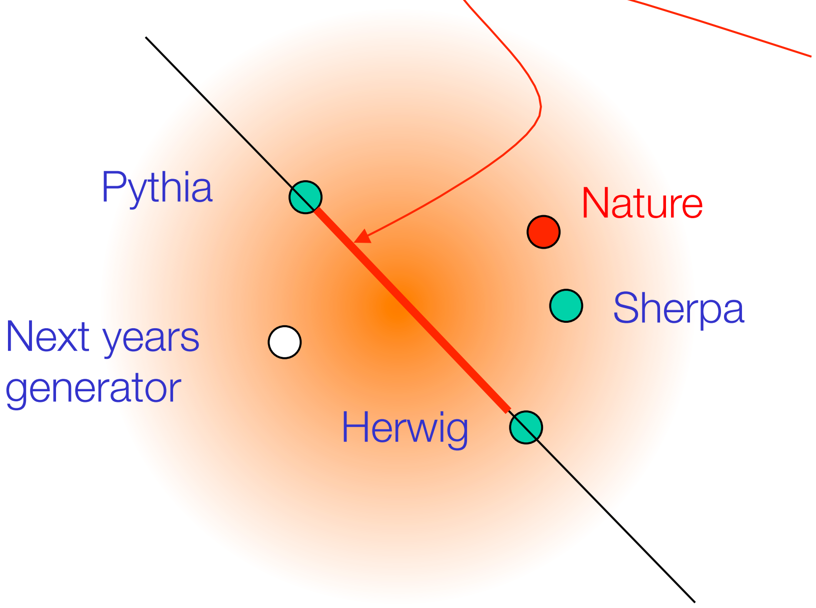
Subsidiary meas.



- But this argument relies crucially on the dimensional correspondence between the observable and the NP
  - The effect on a scalar observable can always be modeled with one NP
  - In other words the existence of a 3rd generator 'Sherpa' can always be effectively capture by the Pythia-Herwig inter/extrapolation
  - It can of course modify your subsidiary measurement (e.g. lending more credence to the Pythia outcome if its result is close, but response model is still valid)

Dealing with 'two-point' uncertainties

- If '2-point'response functions models a distribution, the response corresponding to a new 'third point' is not necessarily mapped by b(α) *for any value of α*
- This point is important in the discussion to what extent a two-point response function can be over-constrained.
  - A result $\alpha_{2p} = 0.5 \pm 1$ has 'reasonable' odds to cover the 'true generator' assuming all generators are normally scattered in an imaginary 'generator space'



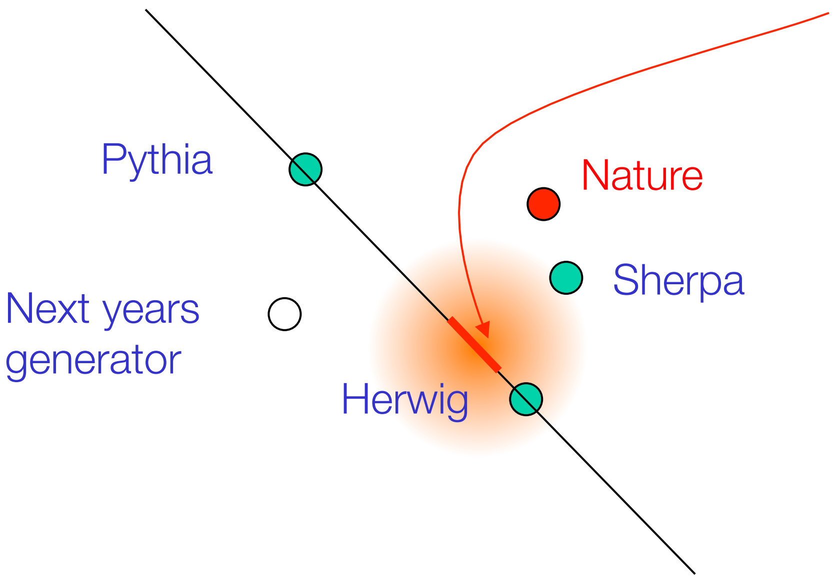
Modeled uncertainty (1 dimension) assuming 'nature is on line'

Effectively captured uncertainty

under the assumption that effect of 'position in model space' in any dimension is similar on response function

Dealing with 'two-point' uncertainties

- If '2-point'response functions models a distribution, the response corresponding to a new 'third point' is not necessarily mapped by b(α) *for any value of α*
- This point is important in the discussion to what extent a two-point response function can be over-constrained.
  - Does a hypothetical overconstrained result α_{2p} *= 0.1* ± *0.2 'reasonably'* cover the generator model space?



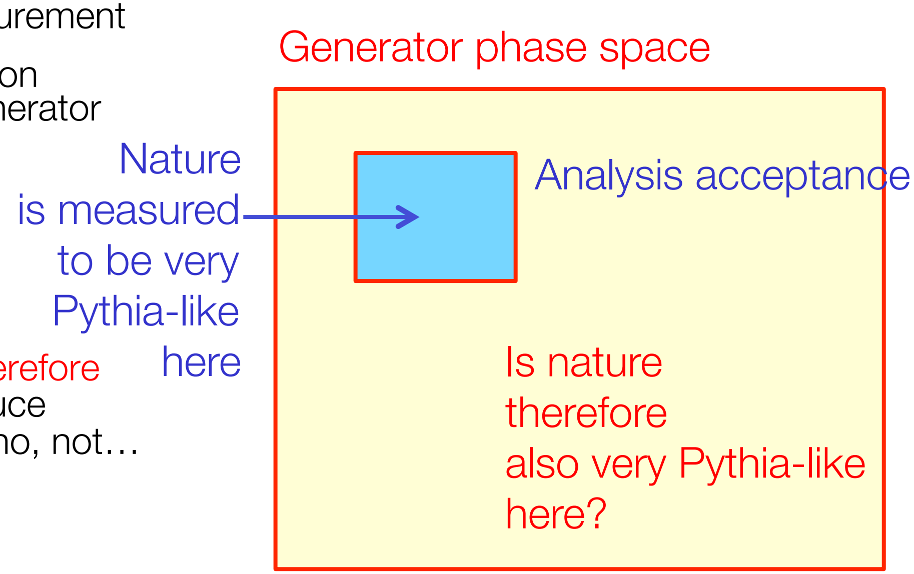
Modeled uncertainty (1 dimension) assuming 'nature' is on line

Effectively captured uncertainty

under the assumption that effect of 'position in model space' in any dimension is similar on response function

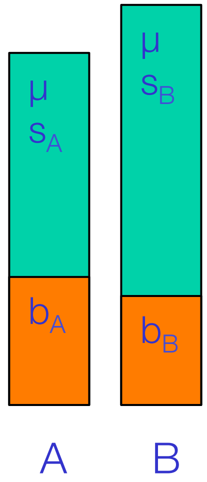
Dealing with ‘two-point’ uncertainties

- Arguments on representativeness of sampling points of ‘2 point’ models raise questions in validity of physics models that over-constrain these
- The main problem is that you become rather sensitive to things you don’t know and quantify: the ‘dimensionality’ of the generator space.
  - To understand what you are doing you’d need to know what all degrees of freedom are (and ideally what they conceptually represent)
  - Unless you know this – trying to reduce the ‘considered space of possibilities’ is rather speculative
  - The real problem is often that you don’t really know what causes the ‘Pythia/Herwig’ effect. Unless you learn more about that there is no real progress.
- The ‘unknown dimensionality’ problem often enters a model in a seemlingly standard modeling assumptions
  - Take an inclusive cross-section measurement
  - Needs to extrapolate acceptance region to full inclusive phase space using generator → Introduces generator systematic
  - Physics likelihood can ‘measure’ that nature inside acceptance is very Pythia-like insideusing 2-point response function with 1 NP
  - Is nature in the entire phase space therefore Pythia-like? If yes, we can greatly reduce inclusive cross-section uncertainty, if no, not…



Ad-hoc solutions - decorrelation

- NPs that are determined to overconstrained due to incorrect modeling assumption can be eliminated through the process of ad-hoc decorrelation
- Take two-bin simple Poisson counting model and a single systematic uncertainty that is modeled in both bins



$$L(N_A,N_B \mid \mu,\alpha) = \underbrace{P(N_A \mid (\mu\cdot\tilde{s}_A+\tilde{b}_A)\cdot r_A(\alpha))}_{A}\cdot\underbrace{P(N_B \mid (\mu\cdot\tilde{s}_B+\tilde{b}_B)\cdot r_B(\alpha))}_{B}\cdot G(0 \mid \alpha,1)$$

  - The physics part of this likelihood may over-constrain α if effect of changing μ or α has a different effect on (N_A,N_B)
  - Can eliminate overconstraint due to correlation between A and B samples by introducing separate NPs for A and B sample

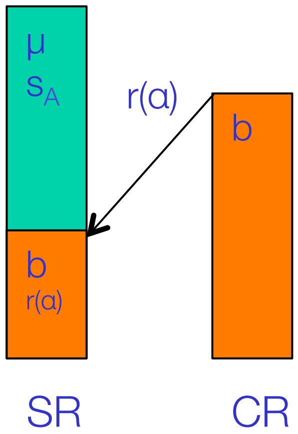
$$L(N_A,N_B \mid \mu,\alpha_A,\alpha_B) = P(N_A \mid (\mu\cdot\tilde{s}_A+\tilde{b}_A)\cdot \underbrace{r_A(\alpha_A)}) \cdot P(N_B \mid (\mu\cdot\tilde{s}_B+\tilde{b}_B)\cdot \underbrace{r_B(\alpha_B)})\cdot G(0 \mid \alpha_A,1)\cdot G(0 \mid \alpha_B,1)$$

  - Interpretation: e.g. for JES, effectively independent calibrations due to different sample composition (e.g. different p_T spectra)
  - *Note that some physics POIs are sensitive to ratios of yields, in such cases a correlated NP may be the more conservative choice*

Ad-hoc solutions – Decorrelation

- Another common type of likelihood model prone to overconstraining issues is the ‘signal/control region model’

$$L(N_{SR}, N_{CR} \mid b, \mu, \alpha) = P(N_{SR} \mid (\mu \cdot \tilde{s} \cdot r_s(\alpha) + b \cdot r_{trams}(\alpha)) \cdot P(N_{CR} \mid b) \cdot G(0 \mid \alpha, 1)$$



  - Control regions measures background rate in CR, mapped background rate is SR via transport factor r(a)
  - Signal in SR modeled from MC simulation
- Both signal acceptance and background transport factor depend on simulation and are subject to systematic uncertainties
  - Common solution → coherent modeling of response functions e.g. for JES
  - But transport factor sensitive to ratio of JES response in CR and SR, signal modeling to JES response in SR only.
  - Since measurement of ‘A/B and B’ is equivalent to measurement of A and B physics likelihood can still over-constrain single JES NP from such a model
  - Solution: decorrelate JES for signal model and transport factor

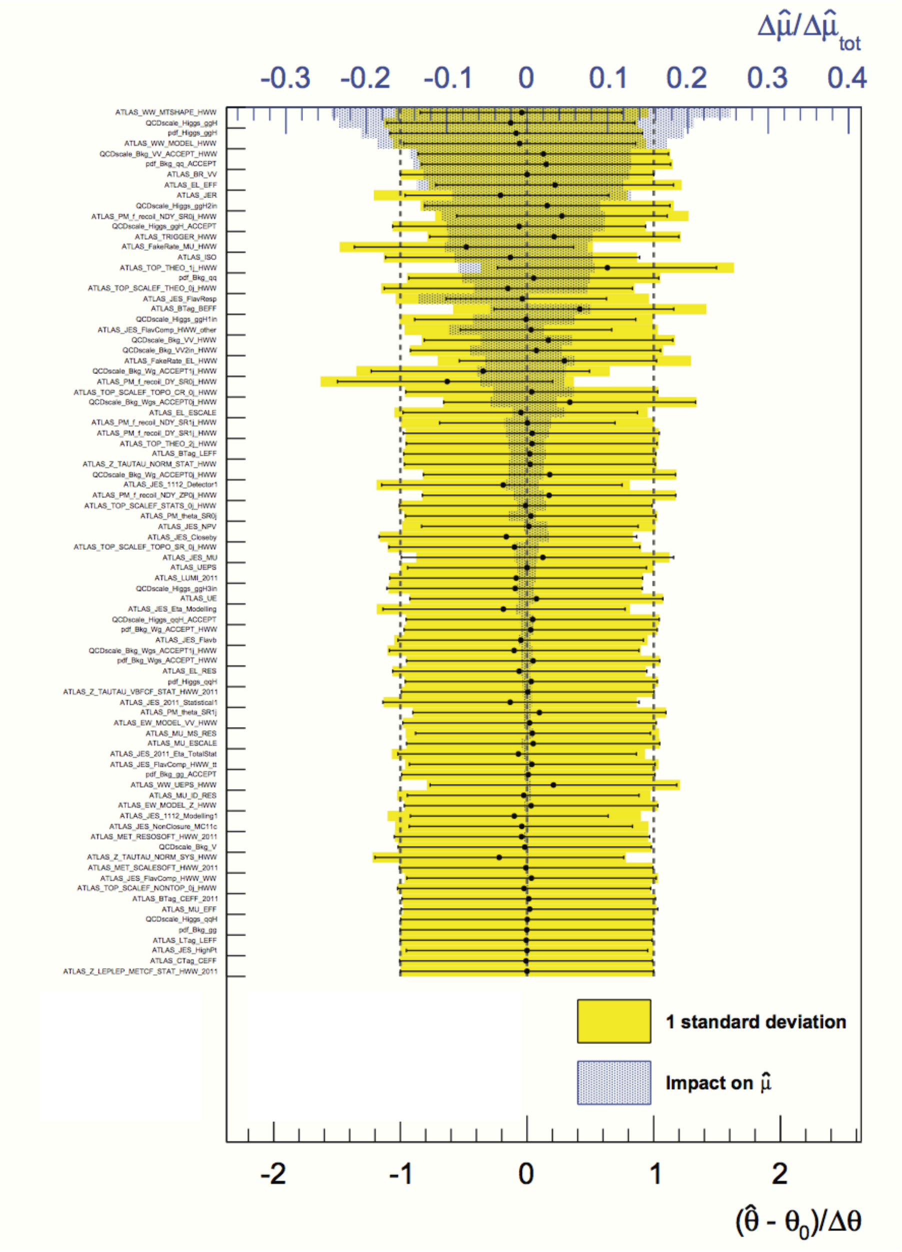
Summary

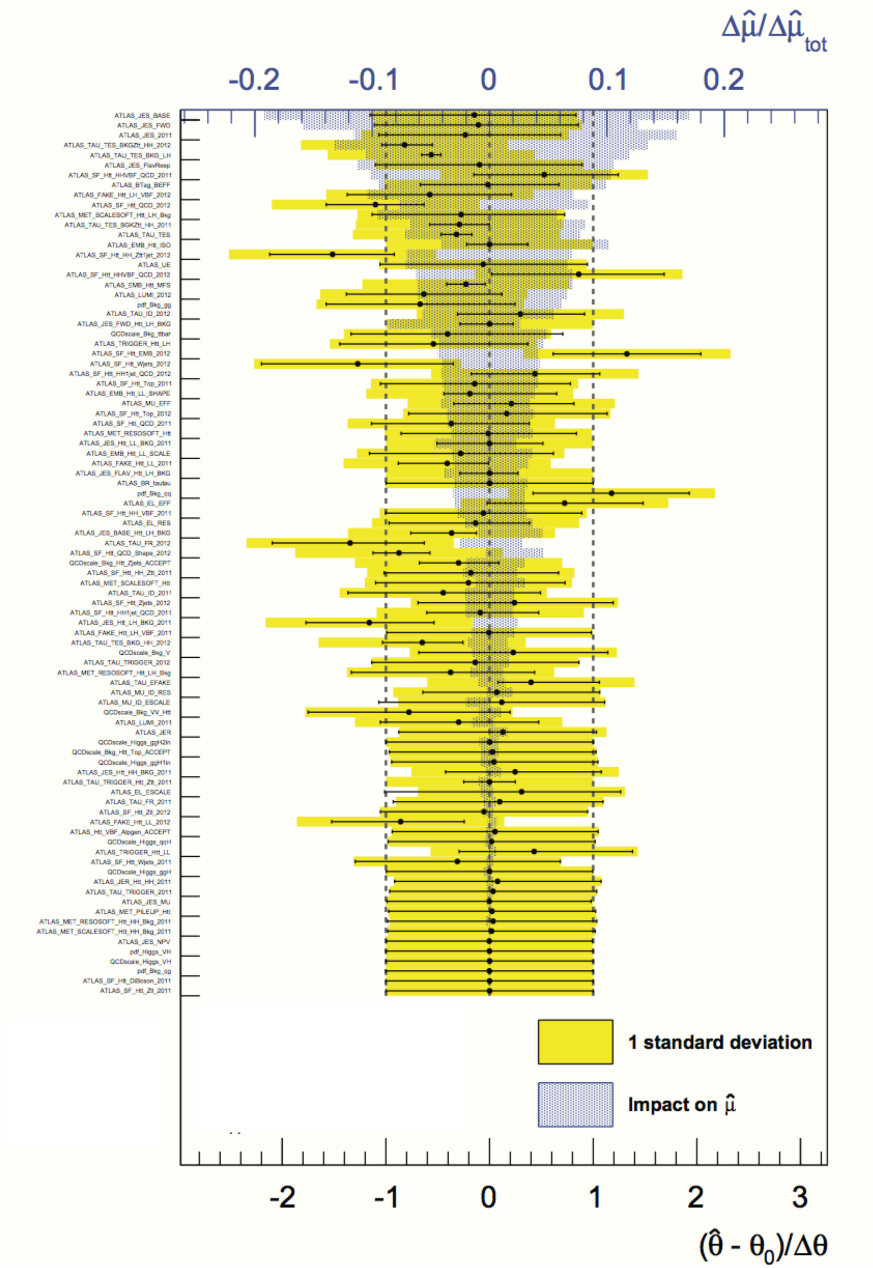
- When performing profile likelihood fits
  - Diagnose carefully if NPs associated with systematic uncertainties are (over)constrained by physics measurements
  - For overconstrained NPs, assess correctness of response model, and choice of sufficient number of NPs to describe underlying systematic uncertainty
  - If overconstraining can be justified on physics arguments, document this as part of the analysis
  - If overconstraining cannot be justified, upgrade to improved response model, or perform ad-hoc decorrelations if that is not possible
  - Use your physics judgement – focus on modeling problems that matter for your POI
    - For ‘irrelevant’ NPs (i.e. those that correlate weakly to your POI) overconstraining may be a non-issue, on the other hand, de-correlation of such NPs will not adversely affect your result and simply analysis approval discussions.

Summary

- Diagnostics over NP overconstraining provide powerful insight into your analysis model
  - An overconstrained NP indicates an externally provided systematic is inconsistent with physics measurement
  - This may point to either an incorrect response modeling of that uncertainty, to result in a genuinely better estimate of the uncertainty
  - Solution not always clear-cut, but you should be at least aware of it.
  - Note that over-constraining always points to an underlying physics issue (lack of knowledge, simplistic modeling) → Treat it as a physics analysis problem, not as a statistics problem
- Diagnostic power of profile likelihood models highlights one of the major shortcomings of the 'naïve' strategy of error propagation (as discussed in Section 1)
  - Physics measurement can entangle in non-trivial ways with systematic uncertainties

Example of likelihood modeling diagnostics





9 Tables of systematic uncertainties

The table of systematic uncertainties

- A common fixture in physics publication of measurements is the "table of systematic uncertainties"
- Implicit in interpretation of "table of systematics" is the notion that all listed effects are uncorrelated
  - For the 'naïve' method of systematics: error propagation, systematics are usually seen and treated this way (regardless of the correctness of this approach

Source of systematic uncertainty on A_C	Electron channel	Muon channel
Detector modelling		
Jet energy scale	0.012	0.006
Jet efficiency and resolution	0.001	0.007
Muon efficiency and resolution	<0.001	0.001
Electron efficiency and resolution	0.003	0.001
b-tag scale factors	0.004	0.002
Calorimeter readout	0.001	0.004
Charge mis-ID	<0.001	<0.001
b-tag charge	0.001	0.001
Signal and background modelling		
Parton shower/fragmentation	0.010	0.010
Top mass	0.007	0.007
$t\bar{t}$ modelling	0.011	0.011
ISR and FSR	0.010	0.010
PDF	<0.001	<0.001
W+jets normalization and shape	0.008	0.005
Z+jets normalization and shape	0.005	0.001
Multijet background	0.011	0.001
Single top	<0.001	<0.001
Diboson	<0.001	<0.001
MC Statistics	0.006	0.005
Unfolding convergence	0.005	0.007
Unfolding bias	0.004	<0.001
Luminosity	0.001	0.001
Total systematic uncertainty	0.028	0.024

- For systematic uncertainties modeled as NPs in likelihood models the likelihood of the physics measurement can introduce correlations
  - This is good, as it is the result of proper modeling
  - But it complicates the naïve pictures of systematic uncertainties, of which the effect can be independently reported in a table

Where do correlations originate?

- Given the full profile likelihood

$$L(\vec{x} \mid \mu, \vec{\alpha}) = L_{physics}(\vec{x} \mid \mu, \vec{\alpha}) \cdot \prod_i Gauss(0 \mid \alpha, 1)$$

- The subsidiary measurements are (usually) designed to factorize, hence covariance $V(\alpha_i, \alpha_j)$ is usually 0 for $i \neq j$

$$L_{subs}(0 \mid \mu, \vec{\alpha}) = \prod_i Gauss(0 \mid \alpha, 1)$$

- Correlation of NP estimators originates from physics measurement
  - Consider this simple counting measurement, which introduce 100% correlation between α_i, α_j

$$L(N \mid \mu, \alpha_1, \alpha_2) = P(N \mid (\mu \cdot \tilde{s}_A + \tilde{b}_A \underbrace{(1 + \alpha_1 + \alpha_2)}_{\text{Evil response function introducing 100\% NP correlation}}) \cdot G(0 \mid \alpha_1, 1) \cdot G(0 \mid \alpha_2, 1)$$

  - Correlations in response function between conceptually uncorrelated systematics are not a mistake – it can just happen for some measuement that e.g. JES and ISR happen to have a nearly identical effect on the POI. That's just physics…

The total and and statistical uncertainty

- The only uncertainty that is *unambiguously defined* is the total uncertainty!

$$V_{tot}(\mu) = V[L(\mu, \vec{\alpha})]$$

  – Where V[] is ML variance estimator (inverse of Hessian matrix)

- Despite that, it is possible to give meaningful information about partial uncertainties, but some caveats apply

- The easiest-to-define component uncertainty is the statistical uncertainty
  – Evaluated as the uncertainty on the POI with all NPs fixed at the values obtained from an unconditional minimization

$$V_{stat}(\mu) = V[L(\mu, \hat{\vec{\alpha}})]$$

  – Note that this interpretation of 'statistical uncertainty' entail the statistics of all data used in the likelihood (including all control samples)

The total systematic uncertainty

- The total systematic uncertainty is the defined by

$$V_{syst}(\mu) = V_{tot}(\mu) - V_{stat}(\mu)$$

$$\sigma_{syst}(\mu) = \sqrt{\sigma^2_{tot}(\mu) - \sigma^2_{stat}(\mu)}$$

  - Note that there is no direct estimator for the 'total systematic uncertainty' it follows from the subtraction of total and statistical variance

- For individual systematic uncertainties (as defined by a corresponding NP) two useful procedures can be defined
  - Keep 'other' NPs either fixed at unconditional estimates (incl) or not (excl)

$$V_{\theta}^{excl}(\mu) = V[L(\mu,\theta,\vec{\alpha})] - V[L(\mu,\hat{\theta},\vec{\alpha})]$$

$$V_{\theta}^{incl}(\mu) = V[L(\mu,\theta,\hat{\vec{\alpha}})] - V[L(\mu,\hat{\theta},\hat{\vec{\alpha}})]$$

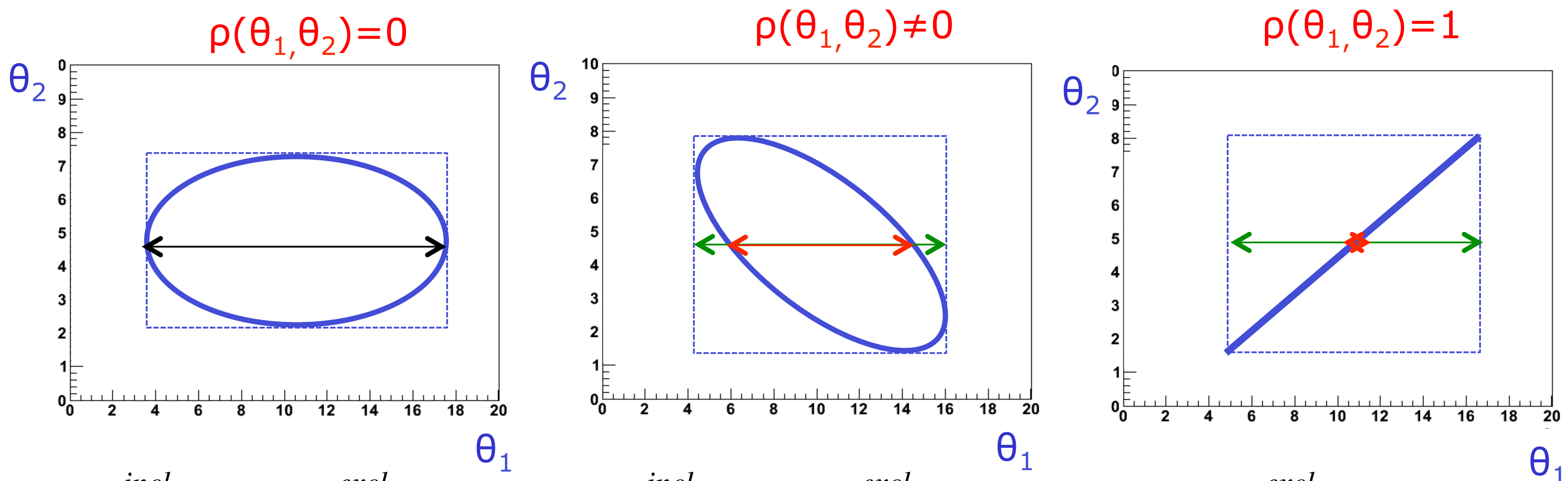
Individual systematic uncertainties

- If there is only 1 NP then both definitions are identical, how to interpret ‘inclusive’ and ‘exclusive variance’ with >1NP?

$$V_\theta^{excl}(\mu) = V[L(\mu,\theta,\vec{\alpha})] - V[L(\mu,\hat{\theta},\vec{\alpha})]$$

$$V_\theta^{incl}(\mu) = V[L(\mu,\theta,\hat{\vec{\alpha}})] - V[L(\mu,\hat{\theta},\hat{\vec{\alpha}})]$$

- Consider example with 2 NPs θ_1, θ_2 that both have $\rho(\mu,\theta_{1,2}) \neq 0$



$V_\theta^{incl}(\mu) = V_\theta^{excl}(\mu)$ $\quad$ $V_\theta^{incl}(\mu) > V_\theta^{excl}(\mu)$ $\quad$ $V_\theta^{excl}(\mu) = 0$

Reporting individual uncertainties in systematic tables

- For tables of systematic uncertainty in the presence of correlations
  - Quadratic sum of inclusive uncertainties
    will add up to more than the total uncertainty
  - Quadratic sum of exclusive uncertainties
    will add up to less than the total uncertainty
  - This is not a problem, just a feature, but you should be aware of it
- A sensible strategy can also be to report the effect of groups of uncertainties, where groups are chosen to be approximately uncorrelated with each other
  - Modify variance estimation procedure to fix/release sets of NPs θ

$$V_{\theta}^{excl}(\mu) = V[L(\mu,\vec{\theta},\vec{\alpha})] - V[L(\mu,\hat{\hat{\vec{\theta}}},\vec{\alpha})]$$

$$V_{\theta}^{incl}(\mu) = V[L(\mu,\vec{\theta},\hat{\vec{\alpha}})] - V[L(\mu,\hat{\hat{\vec{\theta}}},\hat{\vec{\alpha}})]$$

Numeric pitfalls with calculation of exclusive uncertainties

- Any calculation of an individual systematic uncertainty relies on the subtraction of two (numerically obtained) variance estimates
  – Numeric precision can affect your result
- Numerical precision effects can be particularly strong in 'exclusive' uncertainties as for both variance estimates the inversion of a (potentially large) Hessian matrix is required
  – Effect of numeric noise can be much larger than effect of systematic uncertainty (for small uncertainties)
  – You can estimate the magnitude of numeric effects from toy MC studies
- Also be careful with 'artificial asymmetric' uncertainties
  – When comparing MINOS intervals instead of Hessian Variance estimates, small shifts due to numeric noise can create highly-asymmetric intervals for individual 'exclusive uncertainties'
  – Also here study the effect with toy MC studies

Summary

- When constructing tables of systematic uncertainties from profile likelihood models be aware that correlations between NP estimators from the physics measurement will spoil naïve picture of component uncertainties that add in quadrature to the uncertainty
  - Can minimize problem by choosing uncorrelated reporting groups
  - Choose between reporting inclusive or exclusive uncertainty per component
  - Be aware of numeric pitfalls when calculating exclusive uncertainties for small components

10 Summary & conclusions

Summary

- Modelling of systematic uncertainties in the likelihood ('profiling') is the best we know to incorporate systematic uncertainties in rigorous statistical procedures
  - Profiling requires more a 'exact' specification of what a systematic uncertainty means that traditional prescritions → this is good thing, it makes you think about (otherwise hidden) assumption
  - It's important to involve the 'author' of uncertainty prescription in this process, as flawed assumptions can be exploited by statistical methods to arrive at unwarranted conclusions
  - Systematic uncertainties that have conceptual fuzziness ('pythia-vs-herwig') are difficult to capture in the likelihood, but this is a reflection of an underlying physics problem
  - Good software tools exist to simplify the process of likelihood modeling
  - It's important to carefully diagnose your profile likelihood models for both technical and interpretational problems ('over-constraining')