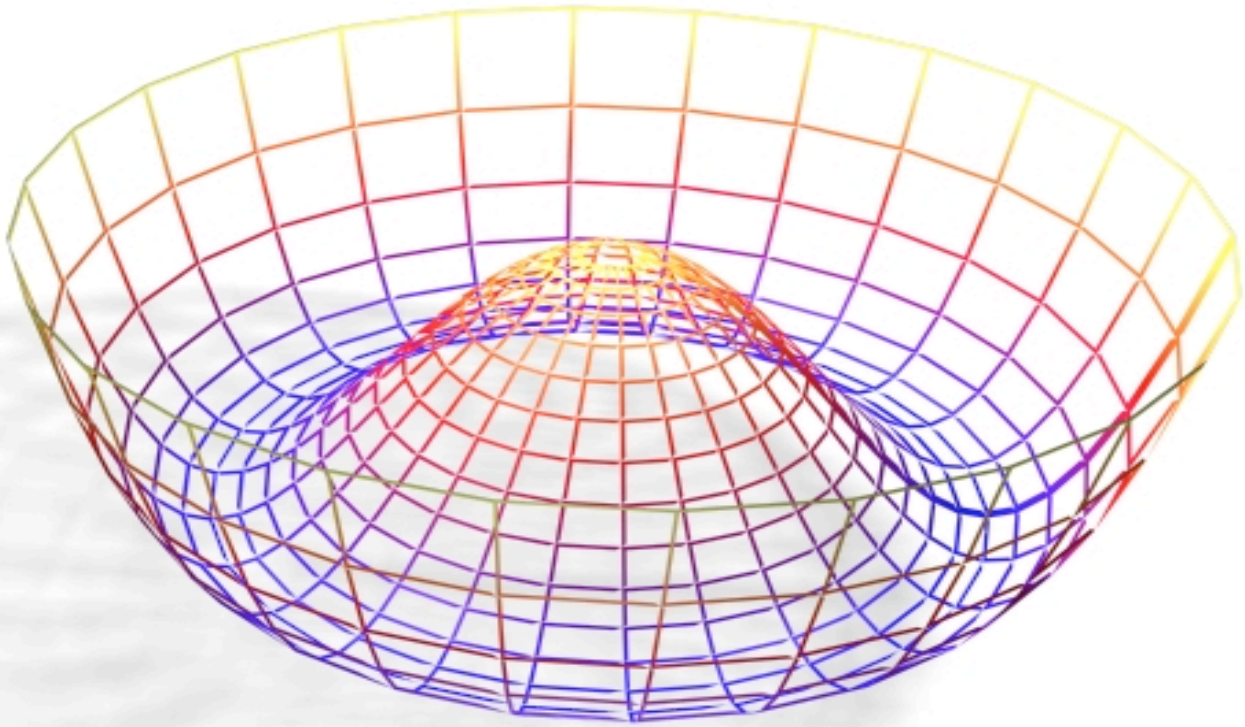
RooStats Developments



Kyle Cranmer & Wouter Verkerke

New email list

The ROOT talk forum did not support email subscriptions, and that is a problem.

- Decided not to use HyperNews based on our last meeting
- Set up a SIMBA2 list:
  - roostats-development@cern.ch
  - Sign up here: http://listboxservices.web.cern.ch/listboxservices/

I hope this will resolve some of our communication problems.

RooFit progress

Documentation

- Extensive set of 70 tutorial macros demonstrating about 95% of RooFit functionality now in ROOT 5.21/02
  - Tested on Linux,Mac,Windows in compiled and interpreted mode
- Good progress on update of Users Manual.
  - Aiming at document of ~170 pages (15 chapters + 4 appendices)
  - Will release first 10 chapters + 1 appendix (120 pages) immediately after next week [1 week delay due to travel schedule]
  - Remainder will follow in next 1-2 months

Code

- Miscellaneous small fixes
- Ability to run latest RooFit (v2.91) with ROOT 5.18.00d
  - LCG will bundle latest RooFit with their next distribution (which contains ROOT 5.18.00d) so that is available in LHC experimental releases (on request from LHCb)

Snapshot of Users Manual Table of Contents

Table of Contents

Table of Contents 2

What is RooFit? 5

1. Installation and setup of RooFit 7

2. Getting started 8
   Building a model 8
   Visualizing a model 8
   Importing data 10
   Fitting a model to data 11
   Generating data from a model 14
   Parameters and observables 14
   Tutorial macros 15

3. Signal and Background – Composite models 16
   Introduction 16
   Building composite models with fractions 16
   Plotting composite models 18
   Using composite models 19
   Building extended composite models 20
   Using extended composite models 22
   Note on the interpretation of fraction coefficients and ranges 22
   Navigation tools for dealing with composite objects 24
   Tutorial macros 27

4. Choosing, adjusting and creating basic shapes 28
   What p.d.f.s are provided? 28
   Reparameterizing existing basic p.d.f.s 29
   Binding TFx, external C++ functions as RooFit functions 31
   Writing a new p.d.f. class 32
   Tutorial macros 35

5. Convolving a p.d.f. or function with another p.d.f. 36
   Introduction 36
   Numeric convolution with Fourier Transforms 37
   Plain numeric convolution 41
   Analytical convolution 43
   Tutorial macros 47

6. Constructing multi-dimensional models 48
   Introduction 48
   Using multi-dimensional models 49
   Modeling building strategy 51
   Multiplication 52
   Composition 53
   Conditional probability density functions 55
   Products with conditional p.d.f.s 58
   Extending products to more than two dimensions 60
   Dealing with per-event errors 60
   Tutorial macros 64

7. Working with projections and ranges 65
   Using a N-dimensional model as a lower dimensional model 65

2

   Visualization of multi-dimensional models 68
   Definitions and basic use of rectangular ranges 69
   Fitting and plotting with rectangular regions 71
   Defining and using parameterized ranges 74
   Working with implicitly defined sub-regions – Likelihood ratio plots 78
   Tuning data projection performance 81
   Selectively ignoring parts a model 82
   Tutorial macros 85

8. Data modeling with discrete-valued variables 86
   Discrete variables 86
   Models with discrete observables 86
   Plotting models in slices of discrete observables 88
   Plotting asymmetries expressed in discrete observables 90
   Unbinned ML fits of efficiency functions using discrete observables 91
   Tutorial macros 93

9. Dataset import and management 94
   Importing unbinned data from ROOT TTrees 94
   Importing unbinned data from ASCII files 95
   Importing binned data from ROOT THx histograms 95
   Manual construction, filling and retrieving of datasets 96
   Plotting, tabulation and calculations of dataset contents 99
   Calculation of moments and standardized moments 101
   Operations on unbinned datasets 102
   Operations on binned datasets 104
   Working with weighted events 105
   Tutorial macros 106

10. Organizational tools 107
   The workspace 107
   ASCII configuration files 107
   Controlling the message service 107
   Tools for debugging 107
   Working with lists and sets 107

11. Simultaneous fits 108
   Introduction 108
   Class RooSimultaneous 108
   The RooSimWSTool utility 108
   A real life complex example 108

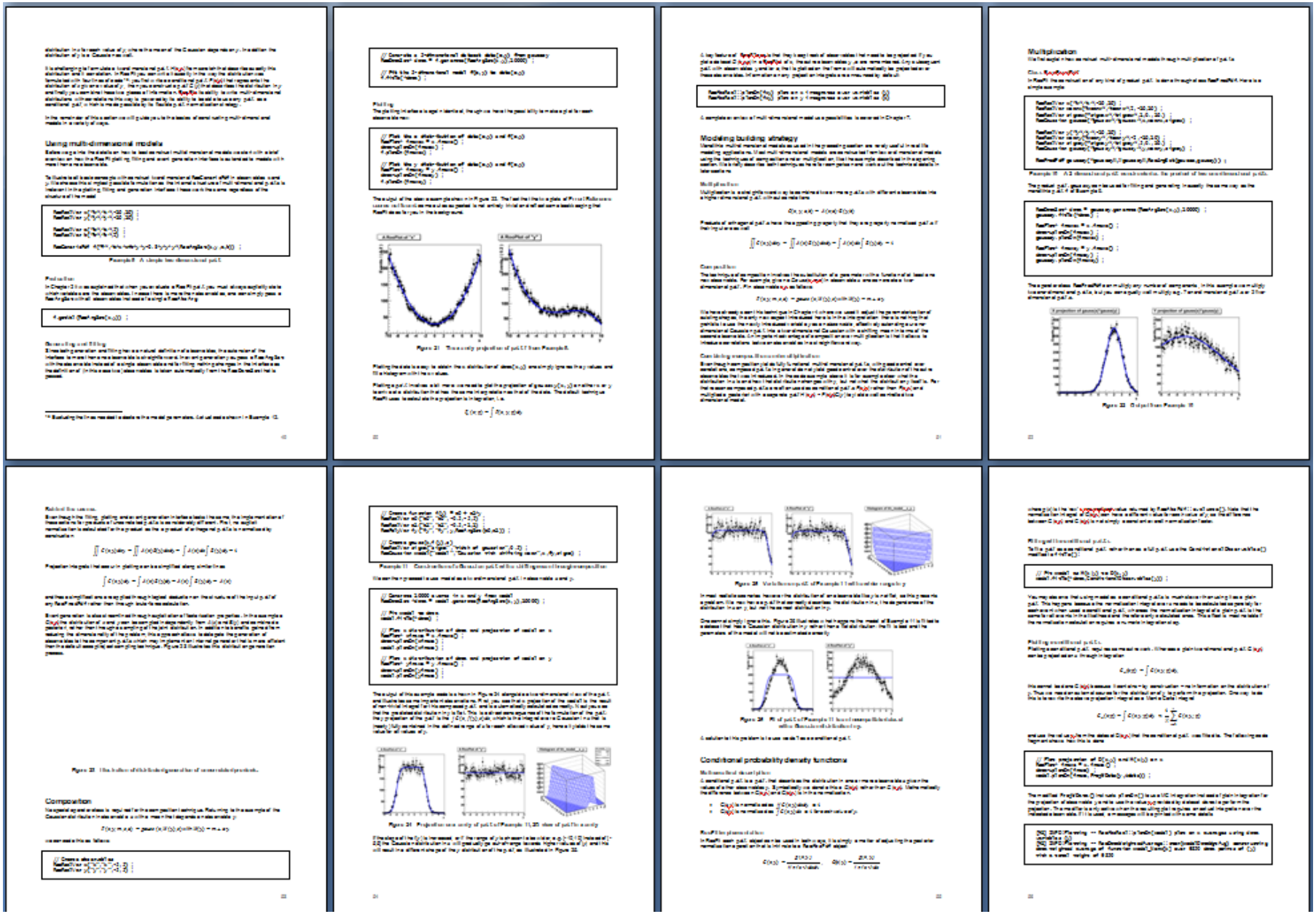
12. Likelihood calculation, minimization 109
   Math introduction 109
   Behind the scenes of fitTo() 109
   Constructing a likelihood or chi2 function 109
   Using the RooFitResult class 109
   Optimization, parallelization 109
   Some MINUIT basics 109
   Constructing the profile likelihood 110
   Working with external constraints on parameters 110
   Dealing with error conditions in the likelihood evaluation 110

13. Special models 111
   Likelihood fits for efficiencies 111
   Pdfs multiplied with efficiencies 111
   Amplitude fits 111
   Shape interpolation 112
   Non-parametric pdfs 112
   B physics decay functions 113

3

Wouter Verkerke, NIKHEF

Snapshot of Users Manual Contents



Wouter Verkerke, NIKHEF

Some notes from RooStats workshop June 26 & 27
Tools for a Combination

In attendance: Kyle Cranmer, Wouter Verkerke, Amir Farbin, Gregory Schott (phone), Maurizio Pierini, Lorenzo Moneta, Walter Bonivento, Danilo Piparo (phone), George Lewis

An example shown previously

A combination example

- Combining 'ATLAS' and 'CMS' result from persisted workspaces

Read ATLAS workspace

```
TFile* f = new TFile("atlas.root") ;
RooWorkspace *atlas = f->Get("atlas") ;
```

Read CMS workspace

```
TFile* f = new TFile("cms.root") ;
RooWorkspace *cms = f->Get("cms") ;
```

Construct combined LH

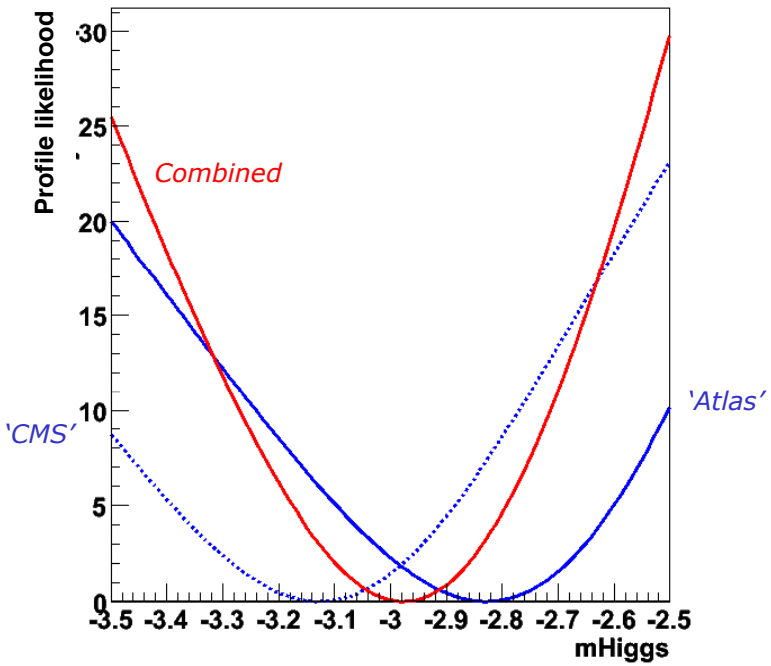
```
RooAddition nllCombi("nllCombi","nll CMS&ATLAS",
            RooArgSet(*cms->function("nll"),*atlas->function("n
```

Construct profile LH in mHiggs

```
RooProfileLL pllCombi("pllCombi","pll",nllCombi,*atlas->var("mH
```

Plot Atlas,CMS, combined profile LH

```
RooPlot* mframe = atlas->var("mHiggs")->frame(-3.5,-2.5) ;
atlas->function("nll")->plotOn(mframe)) ;
cms->function("nll")->plotOn(mframe),LineStyle(kDashed)) ;
pllCombi.plotOn(mframe,LineColor(kRed)) ;
mframe->Draw() ; // result on next slide
```



Wouter Ver

Wouter Verkerke, NIKHEF

RooFit provides two approaches, adding NLL or constructing a joint model and performing a simultaneous fit.

- Simultaneous fit more general, but also more cumbersome; need some utilities

Outlining steps for a combination

Based on some experience during the ATLAS Higgs combination, we identified issues that need to be completed and outlined the steps one would take:

1. Start with a bunch workspaces with no conventions (eg. naming, parametrization, etc.)
2. Import each model into a new workspace, start enforcing conventions
    1. exists: WS::import(RooAbsArg, RenameConflictNodes("sufix"), RenameVar())
3. Make top level pdf via:
        1. RooSimultaneous pdf("","", cat), where cat is a RooAbsCategoryLValue . eg. a supercategory
        2. pdf.addPdf(pdf,state) example state is "{Atlas;Htautau}"
        3. request: can't do RooSimultaneous of RooSimultaneous now
        4. goal: provide a higher-level tool to facilitate these steps based on a scripting language
4. Import data into workspace, start enforcing conventions if necessary
    1. exists: workspace::import(RooDataSet, Rename(), RenameVar())
5. make top level DataSet via:
    1. request: workspace::joinData which will:
        1. create CompositeDataset pointing to input DataSets
        2. add a supercategory to eep track of the source of the DataSets needed for simultaneous fits
6. Feed top level PDF and DataSet to higher level statistical tools for significance, limits, etc..
    1. goal: desire to have a loose coupling between model implementation and steering

Primitives for a Factory

We discussed for some time the idea of a factory and related ideas for a scripting or configuration language

- My worry is that there are at least 4 groups working on this
  - also worry that a scripting "language" can get out of hand
  - many approaches to the factory are tightly coupled to RooFit classes (eg. hardcoded utilities), want something more generic
- We had some short presentations/discussions about existing code and what their goals are

Wouter presented a proof-of-concept for a generic factory that uses RooFit's introspection

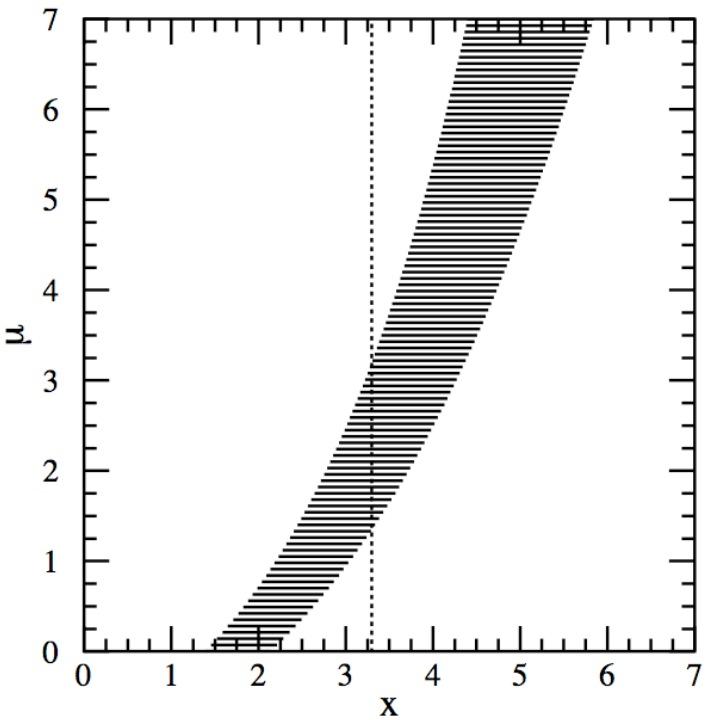
- definitely a much nicer solution
- target is ROOT 5.21-XX or 5.22

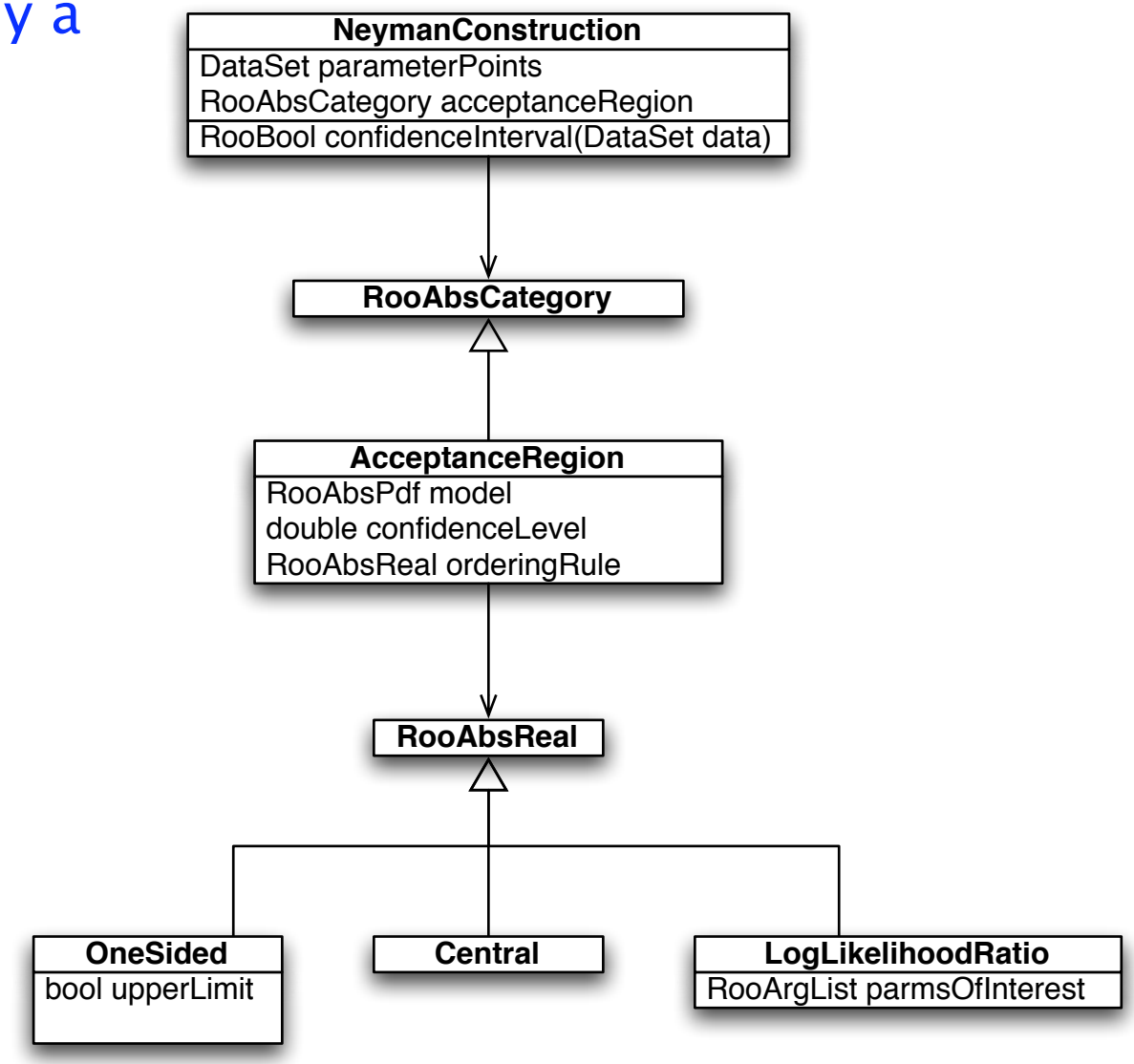
Together with existing tools like renaming and replacing of nodes in the model, we convinced ourselves that a RooFit should provide a few basic primitives and a "language" can be built on top of those

In Development: Neyman Construction

Neyman Construction is simply a set of acceptance regions corresponding to parameter points

- Acceptance region needs ordering rule + pdf + confidence level





- A few approaches for nuisance parameters will be possible

More on nuisance parameters in Neyman Construction

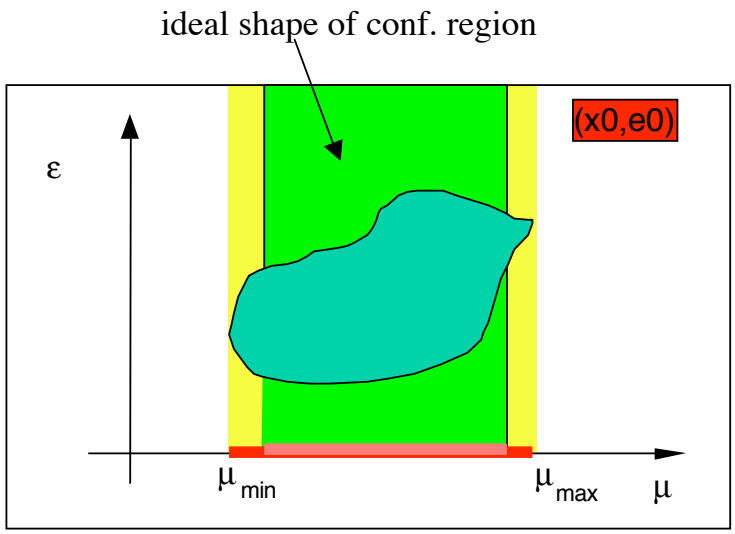
When generalizing the Neyman Construction to include nuisance parameters, there are a few options:

- Full Construction:
  - extend construction to include nuisance parameters
  - project onto space of parameters of interest
    - causes over coverage
- "Profile Construction" / Hybrid Resampling:
  - only perform construction on sub-space defined by constrained MLE of Nuisance parameters
  - proceed as normal for each point
    - approximate, can under-cover for some values of the Nuisance parameter
- Calibrated "Profile Construction"
  - only perform construction on sub-space defined by constrained MLE of Nuisance parameters
  - Adjust cutoff on the ordering rule so that it covers for EVERY value of the nuisance parameters

Overcoverage

The issue with overcoverage

- ***Projecting*** on μ enlarges the CR $\Rightarrow$ overcoverage.
- BUT: we chose to ignore information on ε $\Rightarrow$ we may be able to trade a part of it for additional info on μ, but some overcoverage may just be a natural consequence.
- No basis to assume there is an intrinsic weakness in the projection method: it may well be that ***any*** algorithm that covers can't avoid some overcoverage (as in the discretization effect)
- Real Question: how can I find the smallest μ interval that does not *undercover ?*
- => You want to make the confidence region as close as possible to a rectangle
- Difficulty: the band is build in the x-e plane, but the results are seen in the μ-ε plane. Not very intuitive, also considering that the minimum non-trivial dimensionality is 4



G. Punzi - PHYSTAT 05 - Oxford, UK

Calibrated Profile

Confidence Intervals with Nuisance Parameters

Bodhisattva Sen & Michael Woodroofe
Department of Statistics
University of Michigan

July 18, 2006

Method

Another look at construction of CI's

- *Correct interpretation* of Confidence level

$$P_{\theta,\eta}(\Lambda_\theta(X) \geq c_\alpha(\theta)) \geq 1-\alpha \text{ for all } \theta, \eta$$

- Equivalently: $\min_\eta P_{\theta,\eta}(\Lambda_\theta(X) \geq c_\alpha(\theta)) \geq 1-\alpha$ for each θ
- Thus $c_\alpha(\theta)$ should be the *largest* value of c for which

$$\min_\eta P_{\theta,\eta}(\Lambda_\theta(X) \geq c) \geq 1-\alpha$$

- Can we *analytically* find $c_\alpha(\theta)$?

Conclusion

A combination utility has been outlined at mini Workshop

- requires some new feature requests targeted for next ROOT release

Higher level tools (Neyman Construction, Markov Chain MC, etc.) in development.

- Some overlap in efforts.
  - Plan on stopping MCMC development given Karlsruhe developments
  - Want to insist that Neyman Construction can support all strategies towards dealing with Nuisance Parameters
  - Want to insist that it can use multiple ordering rules
- Now that we have email list, I hope that we can iron out:
  - How is PLScan different than profile likelihood provided by RooFit?
  - Want to converge on factor ideas