Statistics Tools School 2011
Uni Mainz – Institut für Physik

PHYSICS AT THE TERA SCALE
Helmholtz Alliance

— Bayesian Analysis Toolkit

Frederik Beaujean[1], Allen Caldwell[1], Julia Grebenyuk[2], Fabian Kohn[3], Daniel Kollàr[4], Kevin Kröninger[3], Shabnaz Pashapour[3], Arnulf Quadt[3]

[1]Max-Planck-Institut für Physik, [2]DESY, [3]Georg-August-Universität Göttingen, [4]CERN

GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN

07 April 2011

Introduction

- Primary aims of data analysis:
  - Compare data with model
  - Assess the validity of the model
  - Find model parameters
- Bayesian data analysis
  - comprehensive statistical interpretation
  - not trivial to implement
  - need for accessible common tools
- The idea behind BAT is to
  - Provide all the common parts of Bayesian analysis in a software package
  - Create a flexible environment to phrase arbitrary problems
  - Develop a set of well-tested/tuned numerical algorithms and tools

Overview

- BAT:
  - Software package to solve statistical problems using Bayesian approach

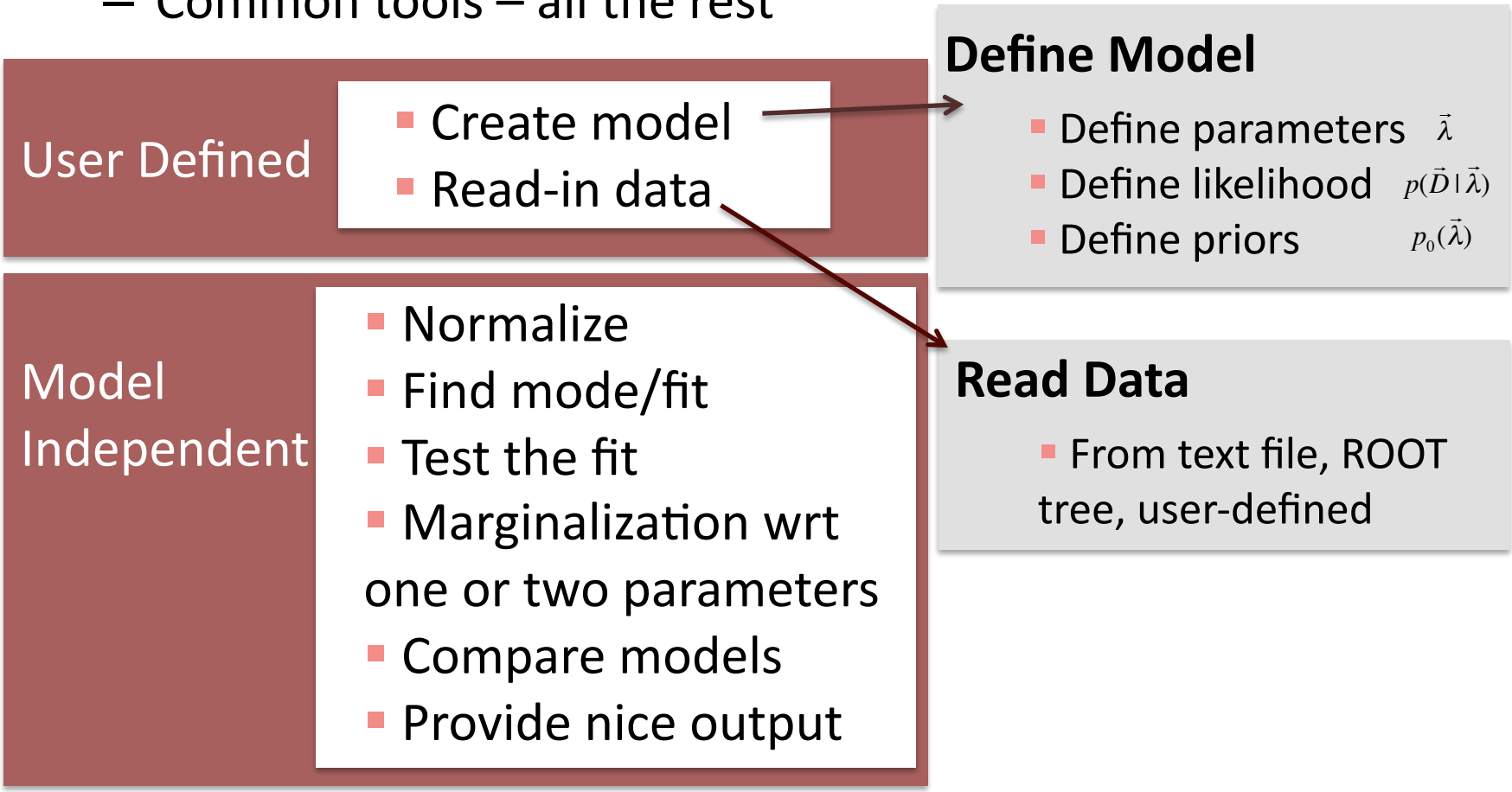
$$p(\vec{\lambda} \mid \vec{D}) = \frac{p(\vec{D} \mid \vec{\lambda}) p_0(\vec{\lambda})}{\int p(\vec{D} \mid \vec{\lambda}) p_0(\vec{\lambda}) d\vec{\lambda}}$$

  - Based on C++ framework in form of a library
  - Interfaced with ROOT, Cuba, Minuit and RooStats
  - Flexible to use user-defined functions and algorithms
  - Free software: tutorials, examples, all at http://mpp.mpg.de/bat/

- The key is the use of **Markov Chain Monte Carlo**
- BAT paper: Computer Physics Communications **180** (2009) 2197-2209

The Approach

- Separate the common parts from the rest
  - Case specific – the model and the data
  - Common tools – all the rest



Common Tools

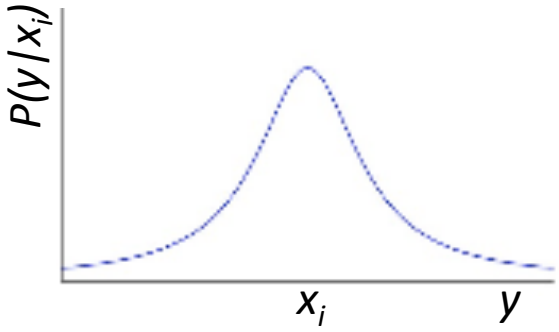
- Marginalization
  - Markov Chain Monte Carlo (MCMC) – Metropolis
  - A lot of emphasis on efficiencies, performance and validation
- Integration
  - Simple Monte Carlo algorithms
    - Sampled mean, importance sampling
  - Interface to CUBA (VEGAS)
- Optimization
  - Monte Carlo (hit & miss)
  - Interface to Minuit
  - Simulated annealing
- Error propagation
  - Calculate any function of the parameters during a run
- Goodness-of-fit
  - Ensemble testing and p-value

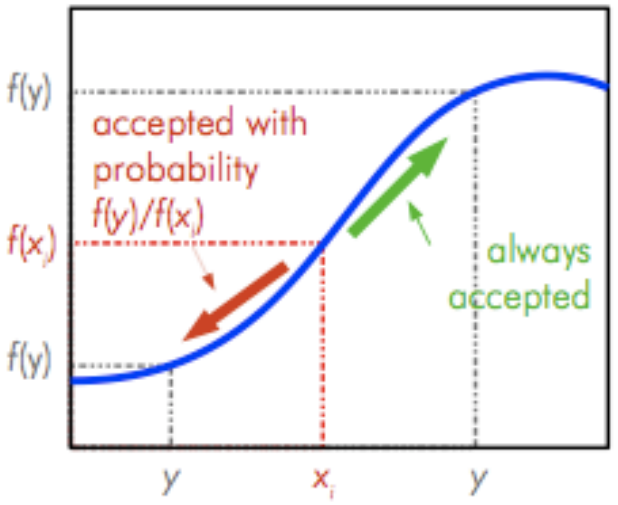
Markov Chain Monte Carlo (MCMC)

- Aim: mapping a positive function *f(x)* by taking a random walk to points with higher probabilities

- Metropolis algorithm in BAT
  - Starts at a random x_i
  - Generate a random point around x_i
    - If $f(y) \geq f(x_i)$, set $x_{i+1} = y$
    - If $f(y) < f(x_i)$, set $x_{i+1} = y$ *with probability* $r=f(y)/f(x_i)$
    - If *y* not accepted, stay where you are
    - Generate a new *y* around the new *x*
  - For an infinite number of steps
    - x_i distribution is guaranteed to converge to *f(x)*
  - For finite number of steps
    - have to check for convergence

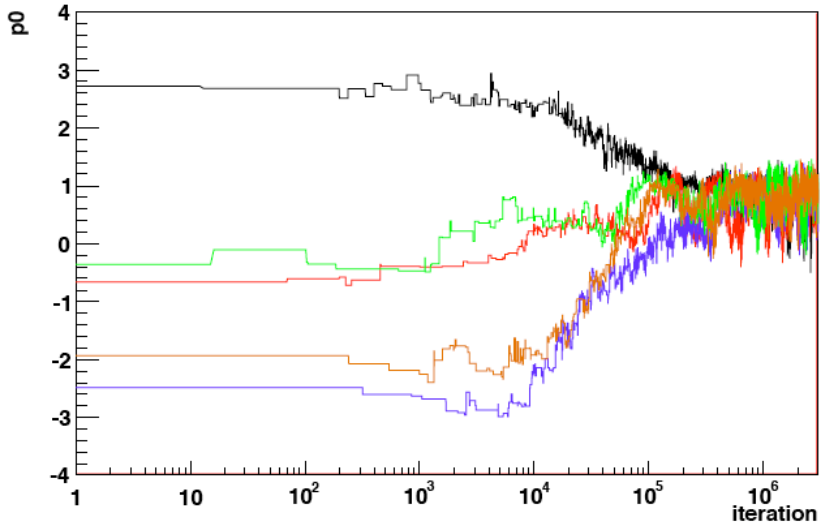
Proposal function plot





MCMC II

- Pre-run/burn-in phase
  - Use several chains/starting positions in parameter space
  - Update scales of proposal function to optimize performance
  - Monitor evolution of log-likelihood and individual parameters
- Convergence based on R-value[1]
  - A ratio of the mean of variances and the variance of the means of chains
  - Efficiency: 15%-50%
- Main run
  - All scales are fixed. Collect samples for posterior analysis
  - Get marginalized distributions
  - Save the chain as TTree.



[1] A. Gelman and D.B. Rubin, *Inference from Iterative Simulation Using Multiple Sequences, Statistical Science* ***7 (1992) 457-472***

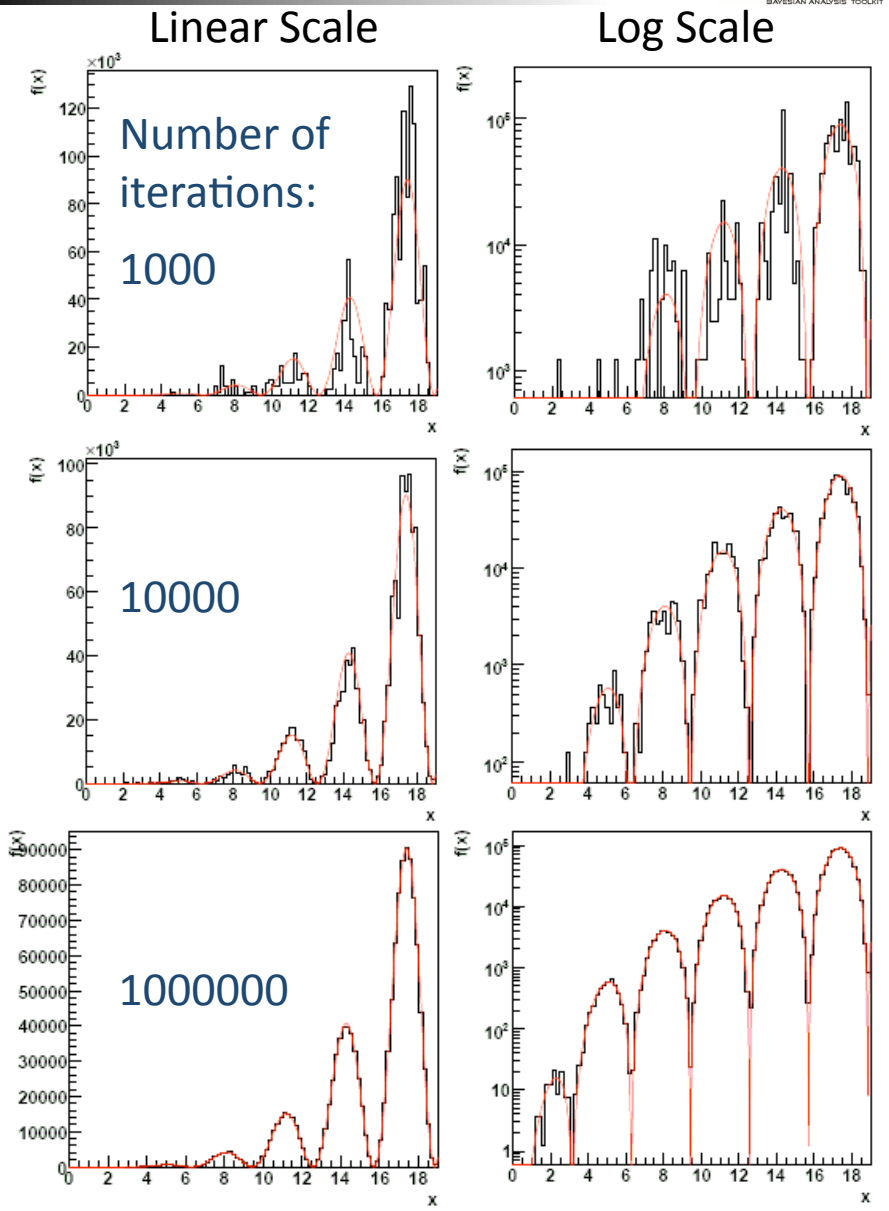
MCMC in Action

- Mapping an arbitrary function:

  $f(x) = x^4 \sin^2 x$

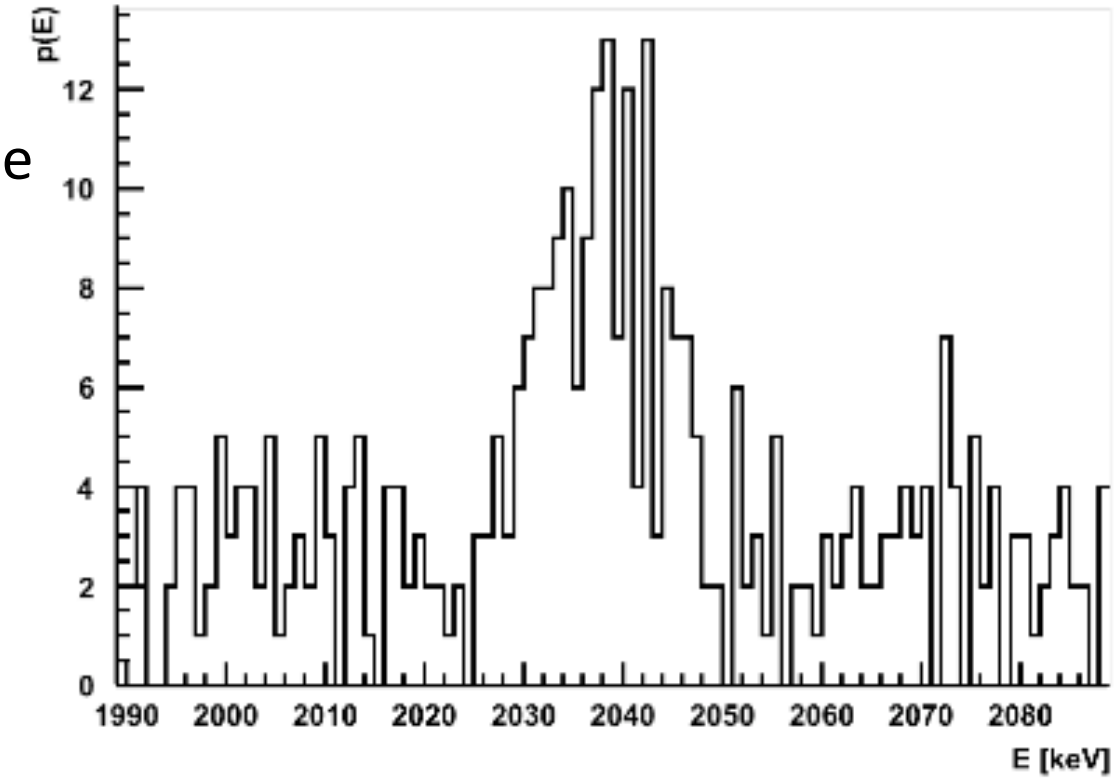
- MCMC sampled distribution quickly converges to the underlying distribution

- Complicated shapes with multiple minima and maxima



An Example: Data

- Estimate signal strength of Gaussian signal on top of flat background
- Data generated with the following settings:
  - Gaussian signal:
  - position μ = 2039 keV
  - width σ = 5 keV
  - strength <S> = 100
  - Flat background: strength <B> = 3/keV
- Number of events per bin fluctuate with Poisson distribution



An Example: Statistical Model

- Gaussian signal on top of flat background
- 4 (+2) fit parameters: Gauss (3) and flat (1) (+2 nuisance parameters for efficiency)

- Prior knowledge:
  - Background: 300 +- 173 in 100 keV (e.g., from sideband analysis)
  - Signal strength: exponentially decreasing (e.g., theoretical intuition)
  - Signal position: flat (e.g., no idea about the mass of a resonance)
  - Signal width: 5 +- 1 keV (detector resolution)
  - Signal and background efficiency fixed to 1 (in this example)

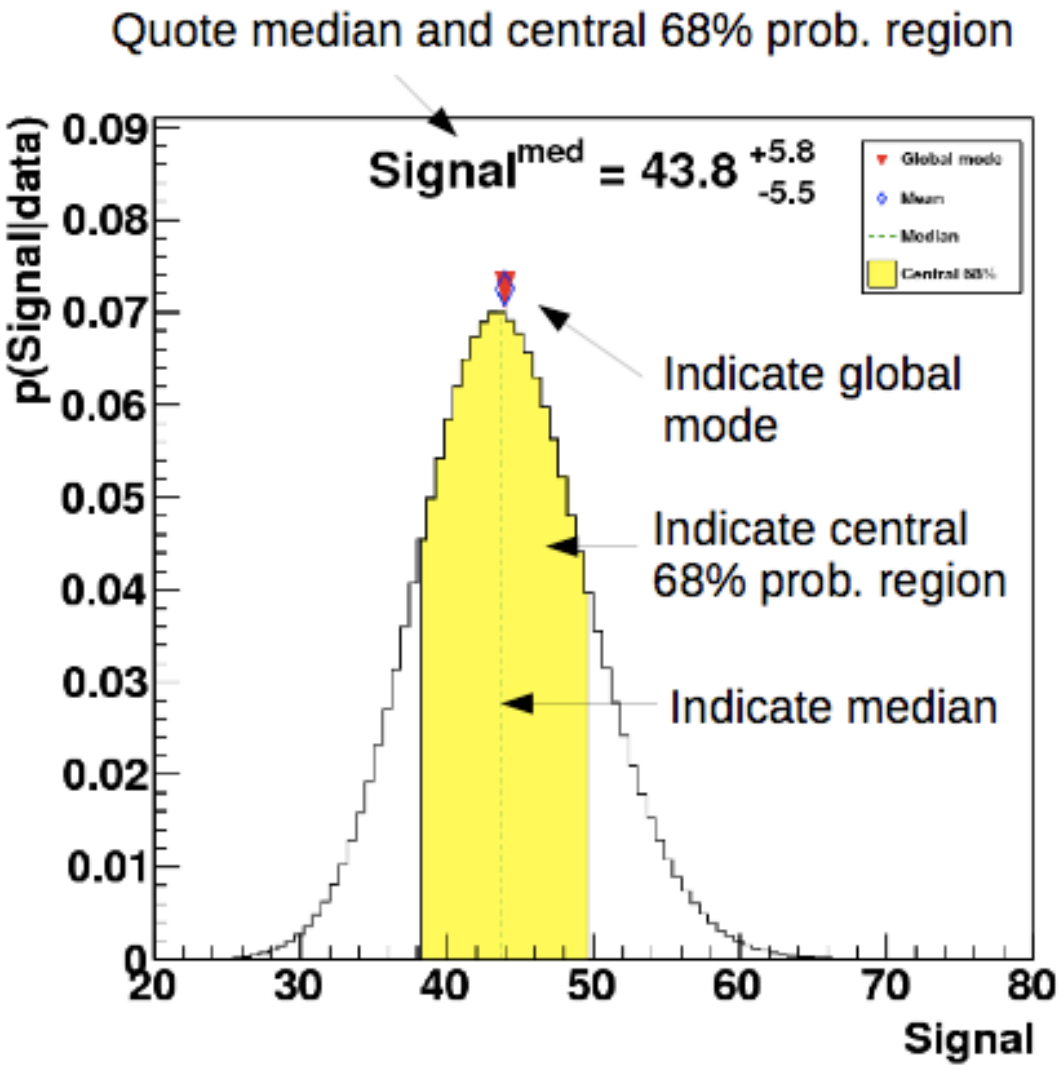
- Statistical model:
  - Bin data
  - Assume independent Poisson fluctuations in each bin

$$p(D \mid S,\mu,\sigma,B) = \prod_{i=1}^{N_{bins}} \frac{\lambda_i^{n_i}}{n_i!} e^{-\lambda_i}$$

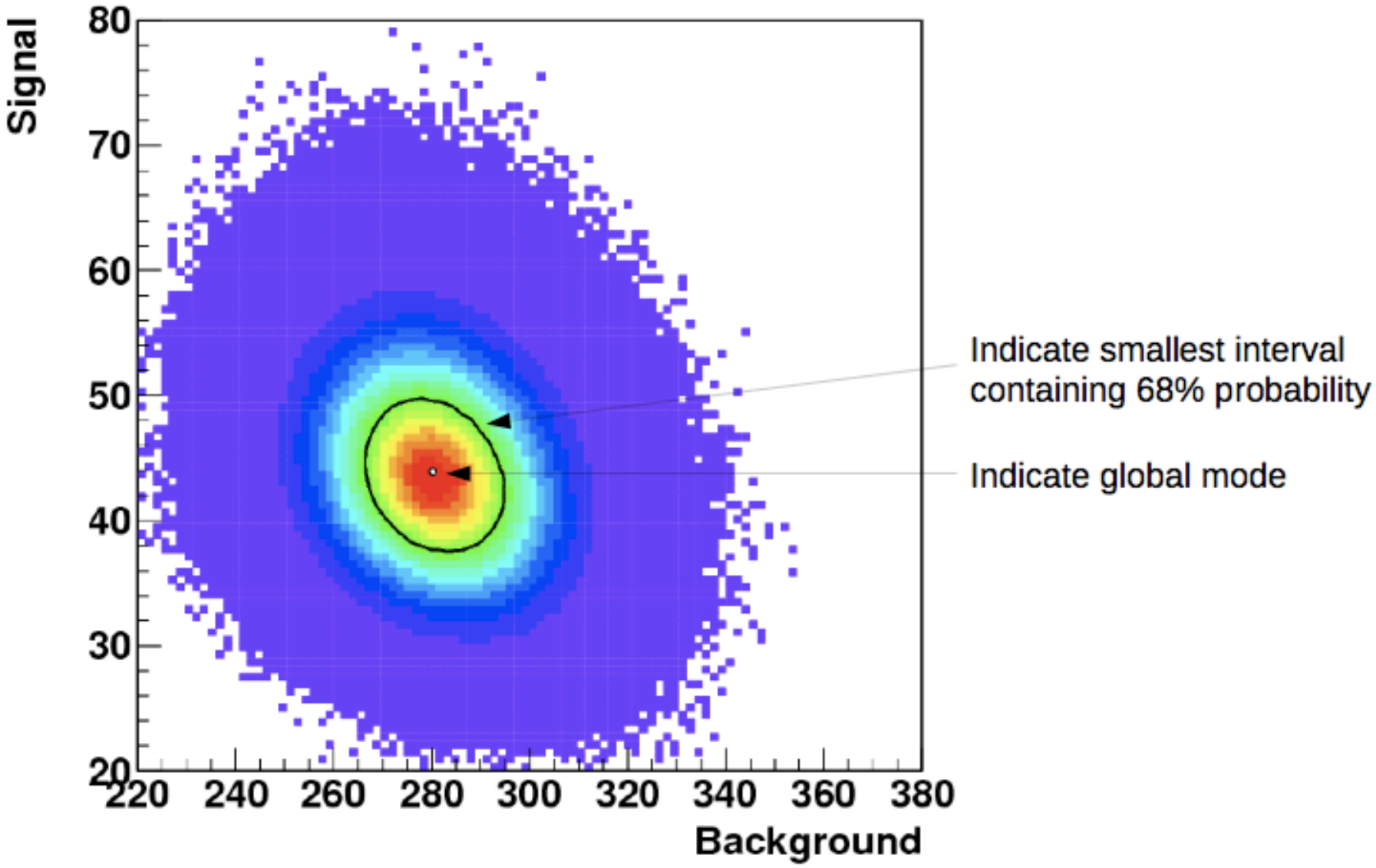
$$\lambda_i = \int_{\Delta_i} \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}} dx + \frac{B}{\Delta_i}$$

An Example: Marginalized distributions

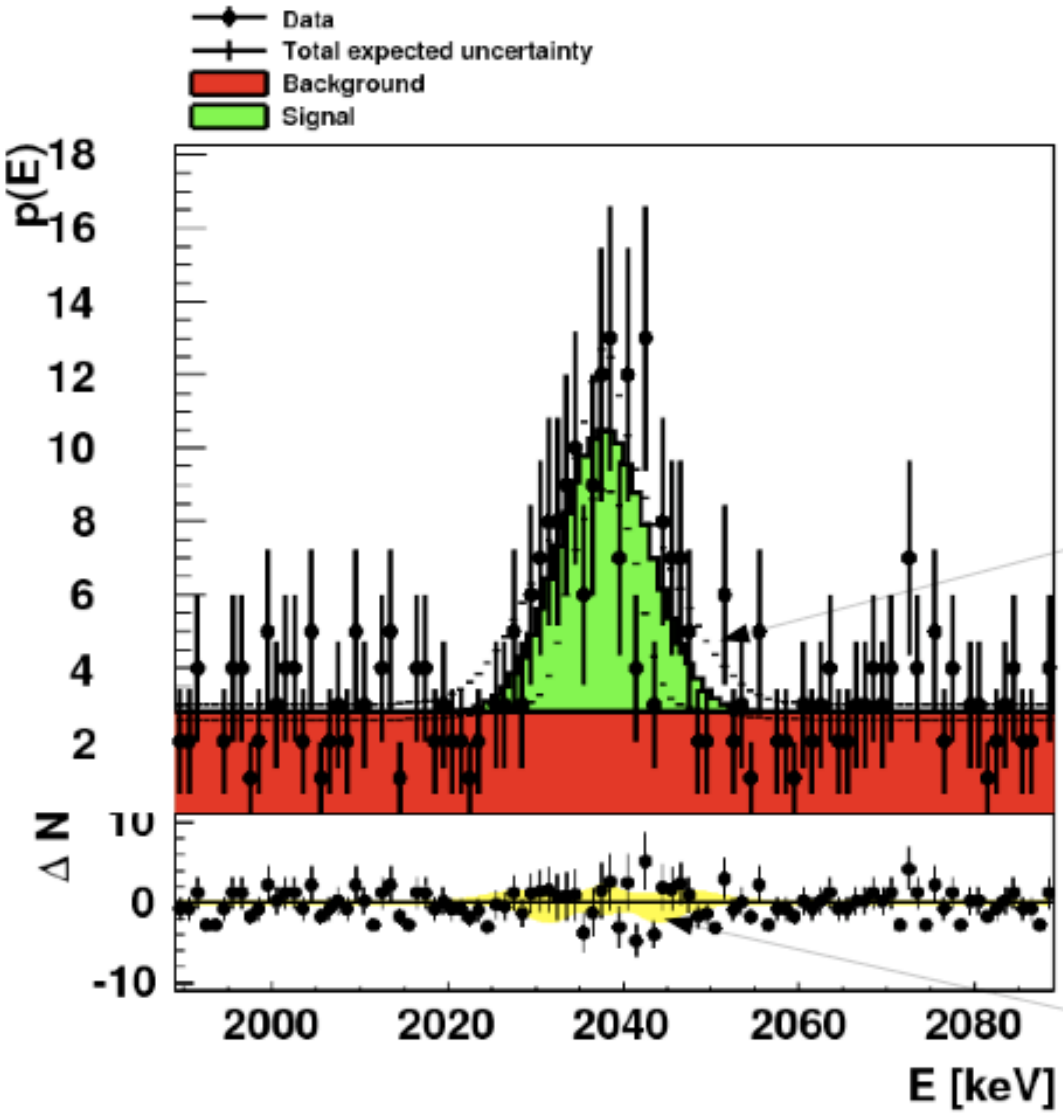
- Project posterior onto one parameter axis, i.e., integrate over all other parameters
- Global mode and mode of marginalized distribution do not have to coincide
- Full (correlated) information in Markov Chain
- Default output:
  - Mean +- std. deviation
  - Median and central int.
  - Mode and smallest int.
- All 1-D and 2-D distributions are written out during main run



An Example: Marginalized distributions



An Example: Stack Plot



Uncertainty of prediction

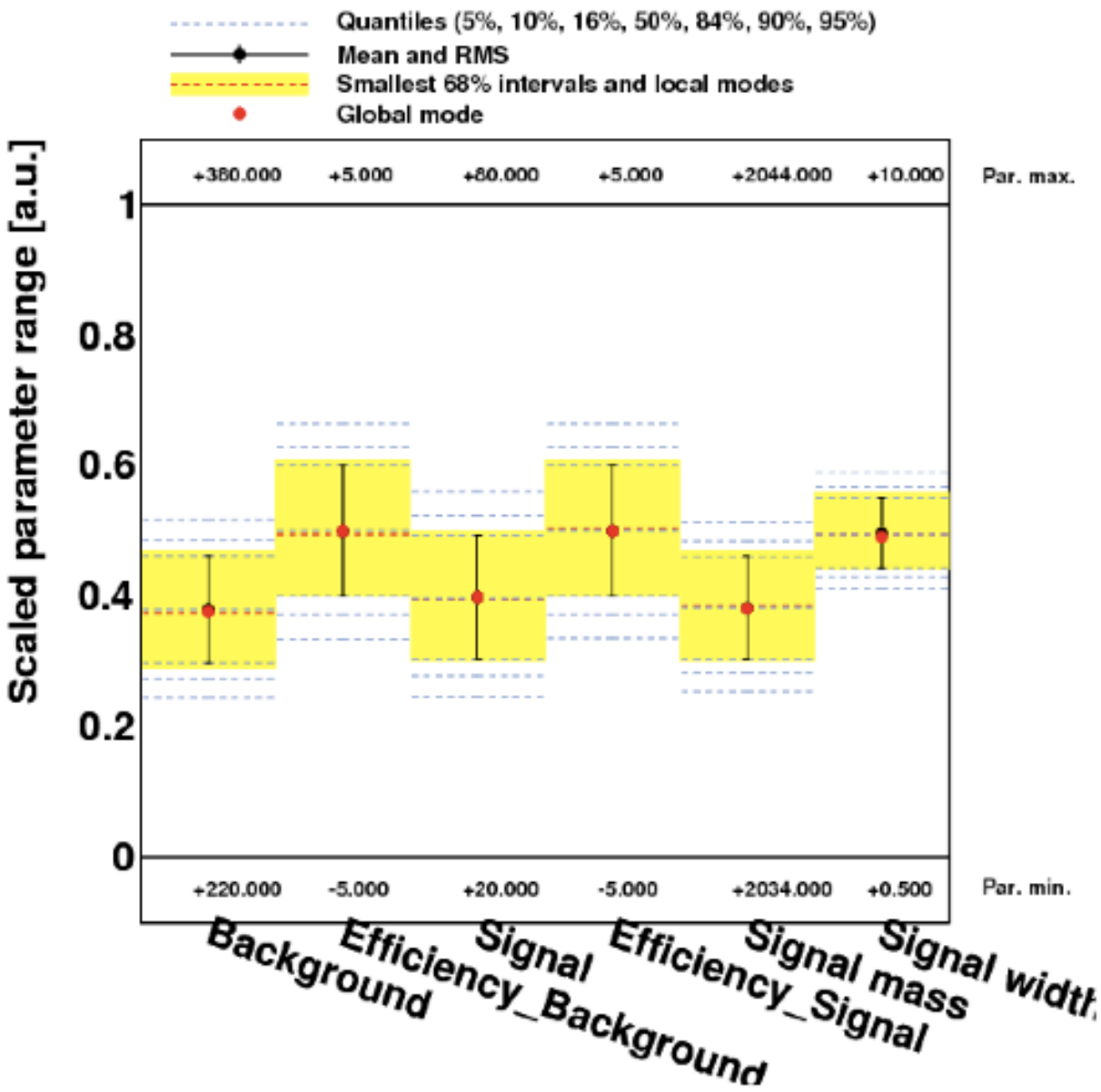
Uncertainty band

An Example: Text Results

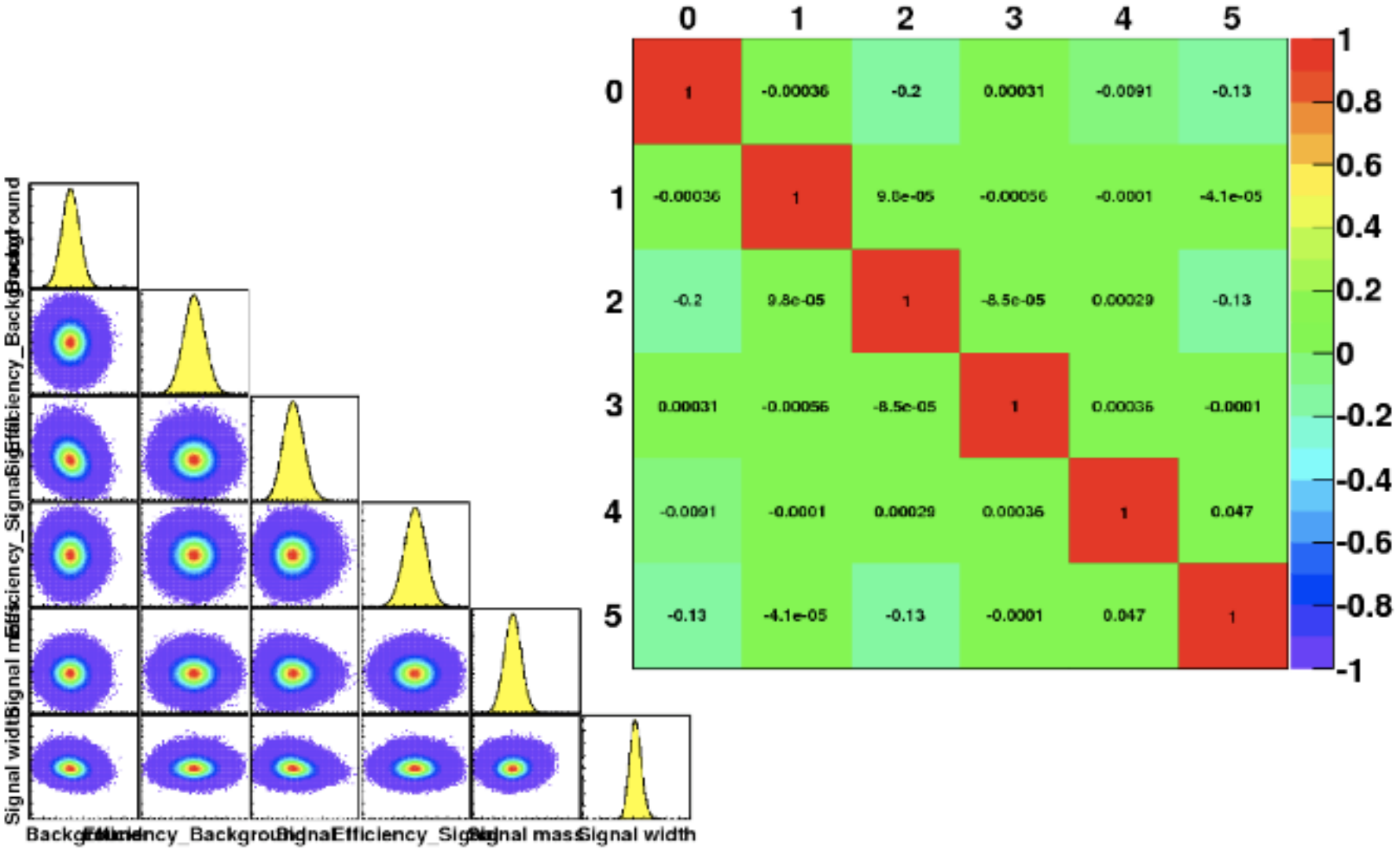
```
Results of the marginalization
==============================
List of parameters and properties of the marginalized
distributions:
 (0) Parameter "Background":
     Mean +- sqrt(V):              280.8 +- 13.16
     Median +- central 68% interval: 280.7 +  13.2 - 13.02
     (Marginalized) mode:          280
      5% quantile:                259.2
     10% quantile:                263.9
     16% quantile:                267.7
     84% quantile:                294.4
     90% quantile:                297.7
     95% quantile:                302.6
     Smallest interval(s) containing 68% and local modes:
      (266.4, 295.2) (local mode at 280 with rel. height 1; rel. area 0.6978)

 (2) Parameter "Signal":
     Mean +- sqrt(V):              43.94 +- 5.724
     Median +- central 68% interval: 43.78 +  5.849 - 5.532
     (Marginalized) mode:          43.7
      5% quantile:                34.8
     10% quantile:                36.71
     16% quantile:                38.25
     84% quantile:                49.88
     90% quantile:                51.38
     95% quantile:                53.62
     Smallest interval(s) containing 68% and local modes:
      (38, 50) (local mode at 43.7 with rel. height 1; rel. area 0.6821)

 (4) Parameter "Signal mass":
     Mean +- sqrt(V):              2038 +- 0.7871
     Median +- central 68% interval: 2038 +  0.7806 - 0.7781
     (Marginalized) mode:          2038
      5% quantile:                2037
     10% quantile:                2037
     16% quantile:                2037
     84% quantile:                2039
     90% quantile:                2039
     95% quantile:                2039
     Smallest interval(s) containing 68% and local modes:
      (2037, 2039) (local mode at 2038 with rel. height 1; rel. area 0.693)
...
```

```
Results of the optimization
===========================
Optimization algorithm used:Metropolis MCMC
List of parameters and global mode:
 (0) Parameter "Background": 280.2 +- 13.08
 (2) Parameter "Signal": 43.94 +- 5.674
 (4) Parameter "Signal mass": 2038 +- 0.7652
 (5) Parameter "Signal width": 5.159 +- 0.5012

Status of the MCMC
==================
Convergence reached:                yes
Number of iterations until convergence: 24000
Number of chains:                   10
Number of iterations per chain:     10000000
Average efficiencies:
 (0) Parameter "Background": 20.03%
 (2) Parameter "Signal": 17.35%
 (4) Parameter "Signal mass": 24.52%
 (5) Parameter "Signal width": 19.56%
```

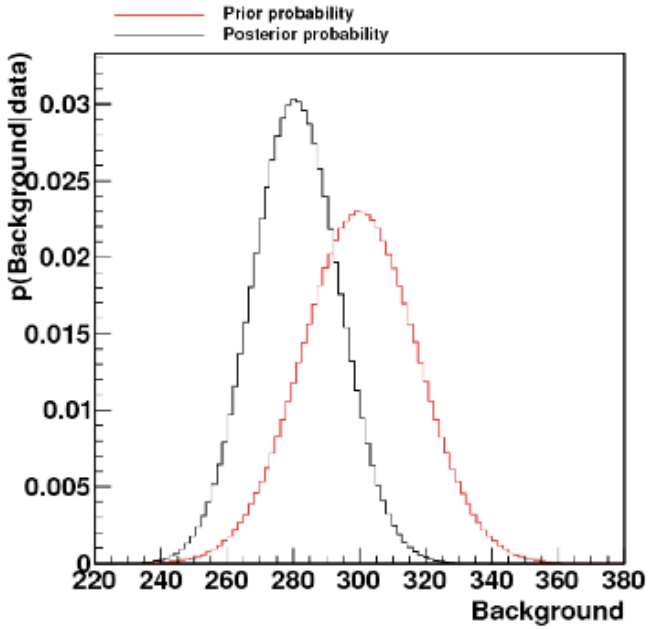
An Example: Parameter Summary

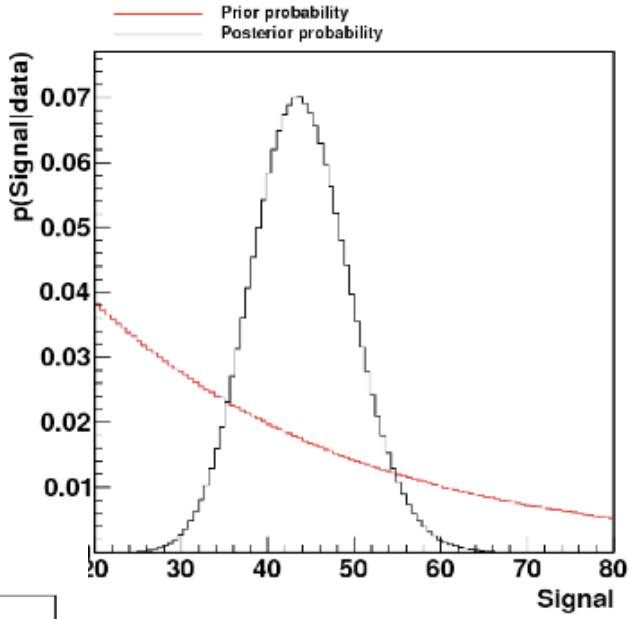


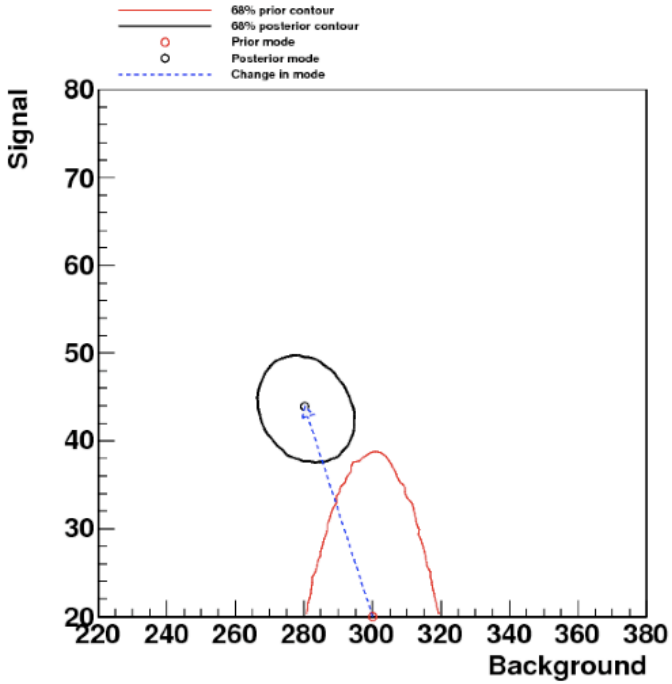
An Example: Correlations



An Example: Knowledge Update







Why BAT?

- Easy to use
- Nice graphical output
- Extensive test suite for proper MCMC sampling
- MCMC output allows for flexible use of posterior
- Simplified error propagation
- Handling very complicated problems with a large number of parameters

- Doing all the hard numerical calculations for model selection and hypothesis testing
- Includes a broad array of sophisticated numerical packages for fitting, integration, …

Tutorial

- Tutorials are online at: http://mpp.mpg.de/bat/?page=tutorials
- A short presentation
- Description on what steps to take
- Solutions are provided as a link
- Tar files of the code are also available, however, please do not use them during the tutorial sessions, you do not need them actually
- Please don't be shy to ask questions! We'll be more than happy to help…

Starting with BAT on the VM

- Suggestion for easier viewing of the results:

  sudo apt-get install gv

- Setup the BAT environment:

  source /statistics-school/BAT-0.4.2/forTuturial/setupBAT.sh

  cd $BATINSTALL/tool

Our Program here

- Counting experiment (today)
- Hypothesis testing (tomorrow)

Summary

- The Bayesian Analysis Toolkit have been introduced
  - Bayesian inference requires some computational effort (e.g., nuisance parameters)
  - Markov Chain Monte Carlo is the key tool
- The philosophy behind it and some of its capabilities are presented
- MCMC implementation and performance in BAT is shown
- An example of data analysis is given