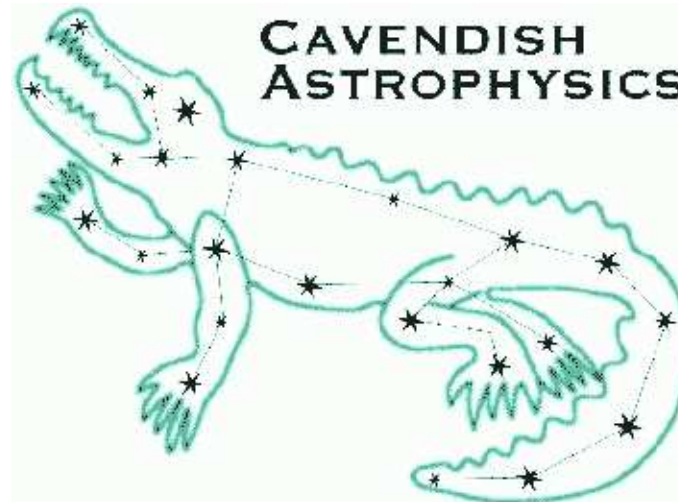# Beyond the Standard Model fitting techniques

Mike Hobson

Astrophysics Group, Cavendish Laboratory, Cambridge

Collider Phenomenology 2011 Workshop: 19th April 2011

(see Auld, Bridges, MPH, Gull – astro-ph/0608174;
Auld, Bridges, MPH – astro-ph/0703445
Feroz, MPH – arXiv:0704.3704
Feroz, MPH, Bridges – arXiv:0809.3437, . . . ,
Bridges et al. – arXiv:1011.4306)

OVERVIEW

- Review of standard BSM fitting methods

- Fast likelihood evaluation: neural networks

- Fast and reliable parameter space exploration: nested sampling

- Application to SUSY phenomenology

- The future: BAMBI

- Conclusions

- Collect a set of N data points D_i $(i = 1, 2, \ldots, N)$, which we denote collectively as the data vector $\boldsymbol{D}$.

- Propose some model (or hypothesis) H for the data, depending on a set of M parameters θ_j $(j = 1, \ldots, M)$, that we denote by the parameter vector $\boldsymbol{\theta}$.

- Apply Bayes' theorem

$$\Pr(\boldsymbol{\theta}|\boldsymbol{D}, H) = \frac{\Pr(\boldsymbol{D}|\boldsymbol{\theta}, H)\,\Pr(\boldsymbol{\theta}|H)}{\Pr(\boldsymbol{D}|H)} \quad \rightarrow \quad P(\boldsymbol{\theta}) = \frac{L(\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{E}$$

- Parameter estimation: posterior $P(\boldsymbol{\theta})$ is complete inference

- Model selection: for H_i $(i = 0, 1)$, the probability density associated with $\boldsymbol{D}$ is

$$E_i = \int L_i(\boldsymbol{\theta})\pi_i(\boldsymbol{\theta})\, d\boldsymbol{\theta}$$

then consider ratio

$$\frac{\Pr(H_1|\boldsymbol{d})}{\Pr(H_0|\boldsymbol{d})} = \frac{E_1}{E_0}\frac{\Pr(H_1)}{\Pr(H_0)}$$

- Consider restricted class of SUSY models with certain universality assumptions regarding SUSY breaking parameters: cMSSM

- Model defined by 4 free parameters:

$$\theta = \{m_0, m_{1/2}, A_0, \tan\beta\},$$

i.e. the common scalar (m_0), gaugino $(m_{1/2})$ and tri-linear (A_0) mass parameters (all specfied at GUT scale) plus ratio of Higgs VEVs $\tan\beta$. We fix $\text{sign}(\mu)$ to be positive, where μ is Higgs/higgsino mass parameter whose square is computed from conditions of radiative EWSB.

- Also have 4 'nuisance' parameters from Standard Model that enter calculation of the observable quantities:

$$\psi = \{M_t, m_b(m_b)^{\overline{MS}}, \alpha_{\text{s}}(M_Z)^{\overline{MS}}, \alpha_{\text{em}}(M_Z)^{\overline{MS}}\},$$

where M_t is pole top quark mass and $m_b(m_b)^{\overline{MS}}$ is bottom quark mass at m_b, while $\alpha_{\text{s}}(M_Z)^{\overline{MS}}$ and $\alpha_{\text{em}}(M_Z)^{\overline{MS}}$ are strong and electromagnetic coupling constants at Z pole mass M_Z (evaluated in MS scheme).

- Denote full 8D parameter space by $\Theta = (\theta, \psi)$

3

- Slow likelihood evaluation

  Likelihood evaluation in SUSY particle model analysis involves computation of mass spectrum via iterative solution of renormalization group equations (RGEs).

  Current public numerical codes reach $\sim$ few % precision and are implemented up to two-loop level, with calculation of physical masses involving, at least, full one-loop radiative corrections
  $\Rightarrow$ typical run time for spectrum calculator $\sim$ few secs per model point

  Also, parameter spaces have unphysical regions (tachyonic solutions and/or EWSB not fufilled), evenly spread across some projections of θ parameter space
  $\Rightarrow$ testing and discarding unphysical points leads to large timing ineffciencies

4
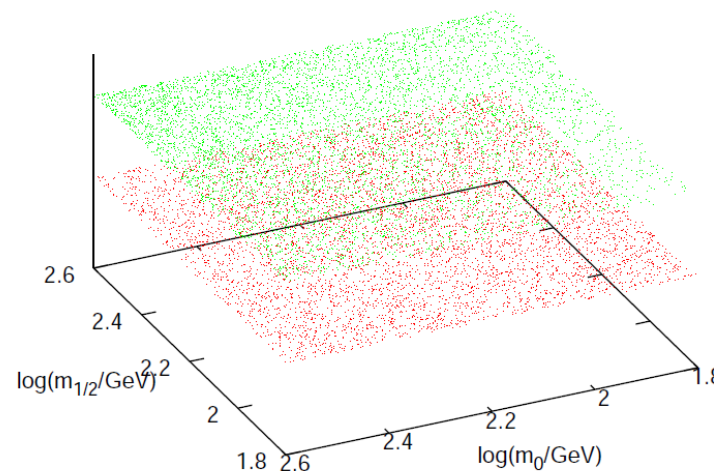
- Slow exploration of parameter space

  Traditional procedure is to evaluate likelihood function on a fixed grid, often encompassing only 2 or 3 dimensions at the time

  $\Rightarrow$ number of likelihood evaluations scales exponentially with no. of parameters

  $\Rightarrow$ impractical for full exploration of cMSSM parameter space

  More recently new approaches based on both Frequentist and Bayesian statistics coupled with Markov Chain Monte Carlo (MCMC) methodology have been applied. The effciency of the MCMC techniques allow for a full exploration of multidimensional models, but still require 10^5–10^7 parameter points sampled

 $\Rightarrow$ full analysis requires $\sim$ few $\times\, 100$ days CPU time

  $\Rightarrow$ perform analysis in $\sim$ 1–2 weeks on supercomputer depending on N_{CPU} available ($\times 2 - 3$ for 'naughty user ranking', queues, etc. . .)

  AND. . . $\times \sim 10$ for model selection using MCMC thermodynamic integration

- **Incomplete exploration of parameter space**
  Likelihood function of these models is complex and multimodal with many narrow features, making the exploration task with conventional MCMC methods challenging often with low sampling effciency
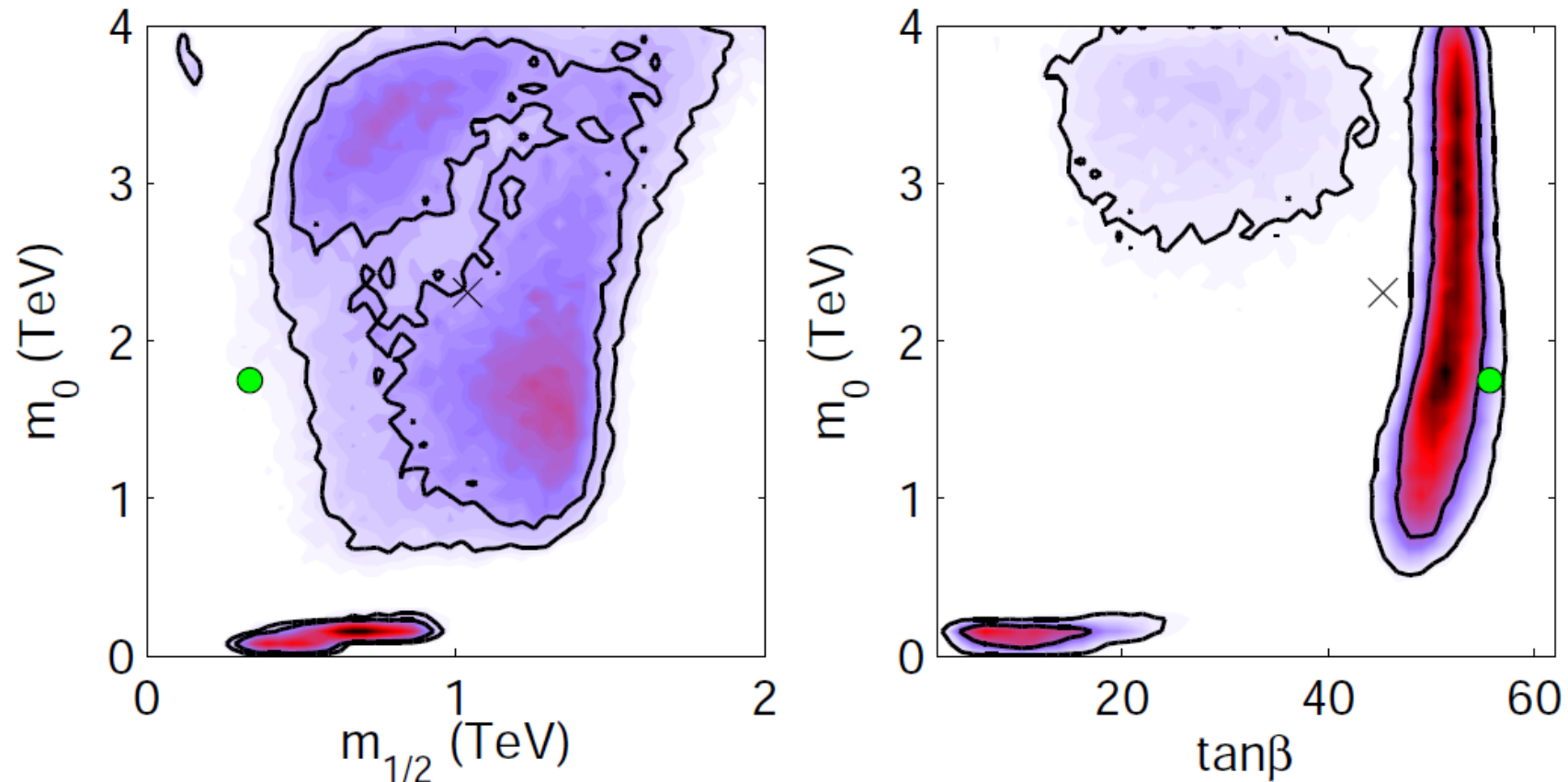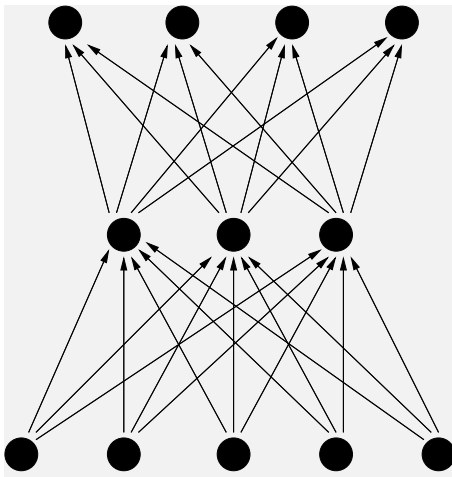
1: Neural networks: fast likelihood evaluation

- MLP = feed-forward network composed of ordered layers of perceptrons

- Consider 3-layer MLP here: input layer, hidden layer and output layer

hidden layer: $\quad h_j = g^{(1)}(f_j^{(1)}); \quad f_j^{(1)} = \sum_l w_{jl}^{(1)} x_l + b_j^{(1)},$

output layer: $\quad y_i = g^{(2)}(f_i^{(2)}); \quad f_i^{(2)} = \sum_l w_{ij}^{(2)} h_j + b_i^{(2)},$

- Use non-linear activation function ($g_1(x) = \tanh x$) on outputs of all hidden layer neurons; use $g_2(x) = x$

- Any L_2-function $f : \Re^n \to \Re^m$, can be approximated to arbitrary mean square error accuracy by a 3-layer MLP

NEURAL NETWORK APPROACH TO BSM PHYSICS

- Any analysis of cMSSM parameter space must relate model parameters $\ominus$ to observable quantities, such as the sparticle mass spectrum m at the LHC

- One can view RGEs (e.g. SOFTSUSY) simply as a mapping from $\ominus \to m$ and engineer a computationally efficient representation of this function

- Vast literature on multivariate function approximation, including neural networks, radial basis functions, support vector machines and regression trees

- Neural networks accurate and easy: random training data, scale linearly with dimension

- Regression 3-layer neural network: input layer $x = \ominus$; output layer $y = m$

- 8-parameter model in neighbourhood of SUSY benchmark point

CMSSM parameters: $\bar{\theta}$
$\log(50) < \log m_0, \log m_{1/2}\ (\text{GeV}) < \log(500)$
$-4 < A_0\ (\text{TeV}) < 4$
$2 < \tan\beta < 62$
SM nuisance parameters: ψ
$3.92 < M_t < 4.48$
$163.7 < m_b(m_b)^{\overline{MS}} < 178.1$
$0.1096 < \alpha_{em}(M_Z)^{\overline{MS}} < 0.1256$
$127.846 < \alpha_s(M_Z)^{\overline{(MS)}} < 127.99$

- Easily applicable to other regions of parameter space, e.g. focus point

- Now extending representation to wider parameter ranges

- Training data: $\mathcal{D} = \{\Theta^{(k)}, m^{(k)}\}$
  - randomly select ~ 4000 points $\Theta^{(k)}$ in box in cMSSM parameter space
  - calculate corresponding sparticle mass spectra $m^{(k)}$ using SOFTSUSY

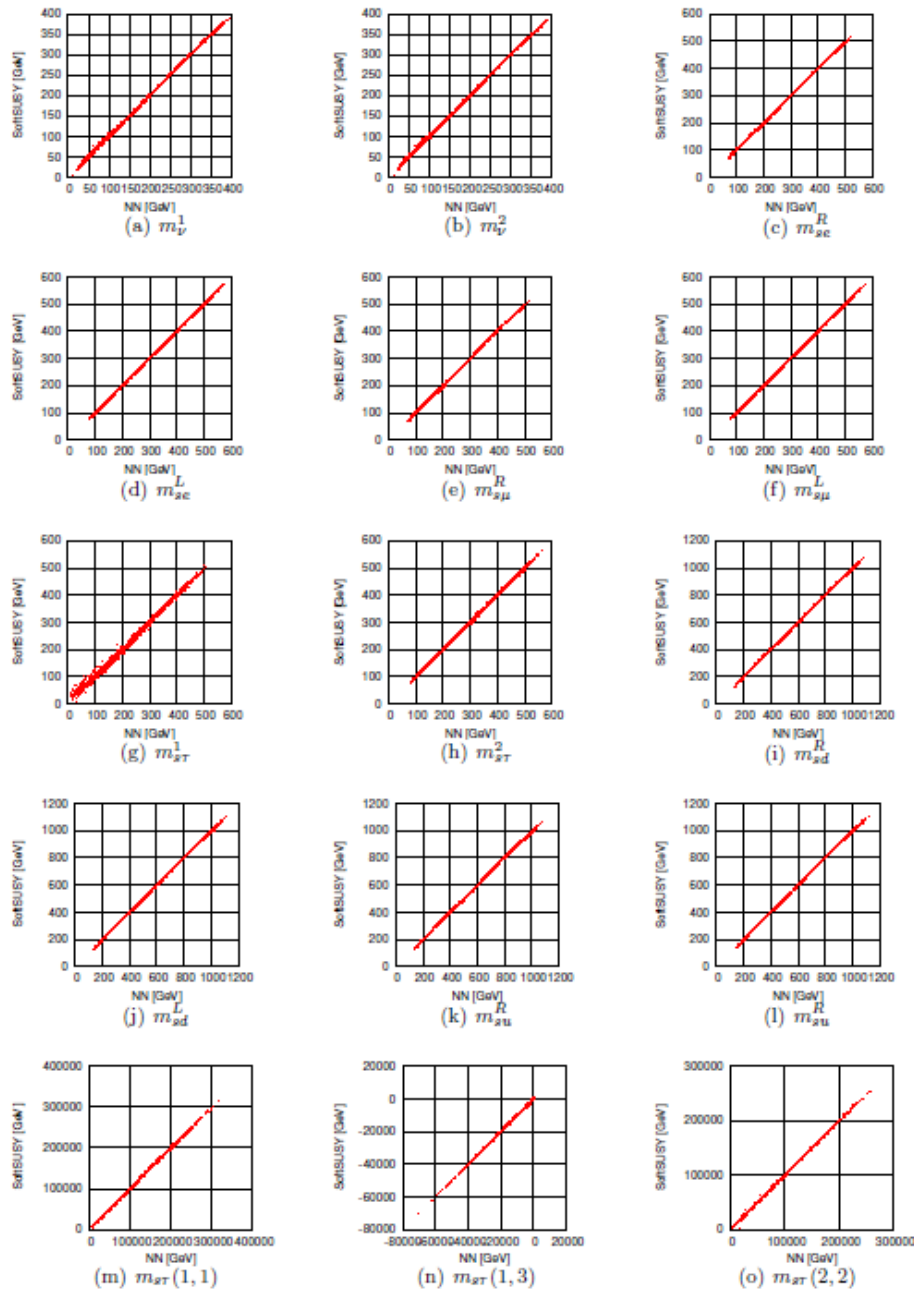- Maximise log-likelihood (misfit) with respect to network parameters $a = (w, b)$:

$$\mathcal{L}(a) = -\tfrac{1}{2} \sum_k \sum_i \left[m_i^{(k)} - y_i(\Theta^{(k)}; a) \right]^2$$

- Highly non-linear function in 1000s of dimensions $\Rightarrow$ use MEMSYS optimiser on:

$$F(a) = \mathcal{L}(a) + \alpha S(a)$$

- Increments α down the maximum entropy trajectory (starting from $\alpha = \infty$) until the error term dominates; trains in ~ 10 mins with 10 hidden nodes (max evidence)

- Create separate test data to evaluate accuracy

(a) m_ν^1 (b) m_ν^2 (c) m_{sc}^R

(d) m_{sc}^L (e) $m_{s\mu}^R$ (f) $m_{s\mu}^L$

(g) $m_{s\tau}^1$ (h) $m_{s\tau}^2$ (i) m_{sd}^R

(j) m_{sd}^L (k) m_{su}^R (l) m_{su}^R

(m) $m_{s\tau}(1,1)$ (n) $m_{s\tau}(1,3)$ (o) $m_{s\tau}(2,2)$

- Comparison of SOFTSUSY and NN mass spectra
- All mass spectrum components have correlation coefficients > 0.9999
- Typical CPU time for each NN mass spectrum calculation $\sim$ few $\times 10^{-4}$ s
- Factor of $\sim 10^4$ speed-up over SOFTSUSY full RGE solution

12

- cMSSM parameter space has unphysical regions spread evenly across it
⇒ testing and discarding unphysical points leads to large timing ineffciencies
⇒ train classification neural network to identify physical and unphysical points

- Train 3-layer NN: input layer $x = \Theta$; output layer $p_i = \frac{e^{y_i}}{\sum_j e^{y_j}}$ (SoftMax)
⇒ output p_i gives probability of input vector belonging to ith-class

- Training data: $\mathcal{D} = \{\Theta^{(k)}, t^{(k)}\}$
– kth target vector $t^{(k)}$ has unity in class of kth input vector $\Theta^{(k)}$ and zeroes elsewhere
– randomly select ~ 30000 points $\Theta^{(k)}$ in box in cMSSM parameter space

- Network training as before but using $\mathcal{L}(a) = \sum_k \sum_i \left[t_i^{(k)} \ln p_i(\Theta^{(k)}; a) \right]$
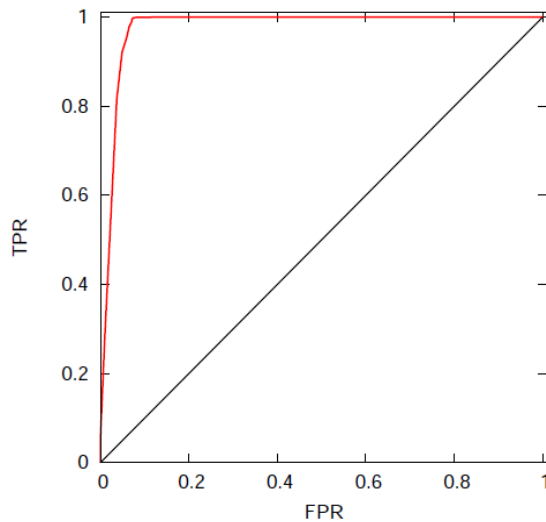
- ROC curve plots true positive rate (TPR) vs false positive rate (FPR) as a function of classifier threshold p_{th}.
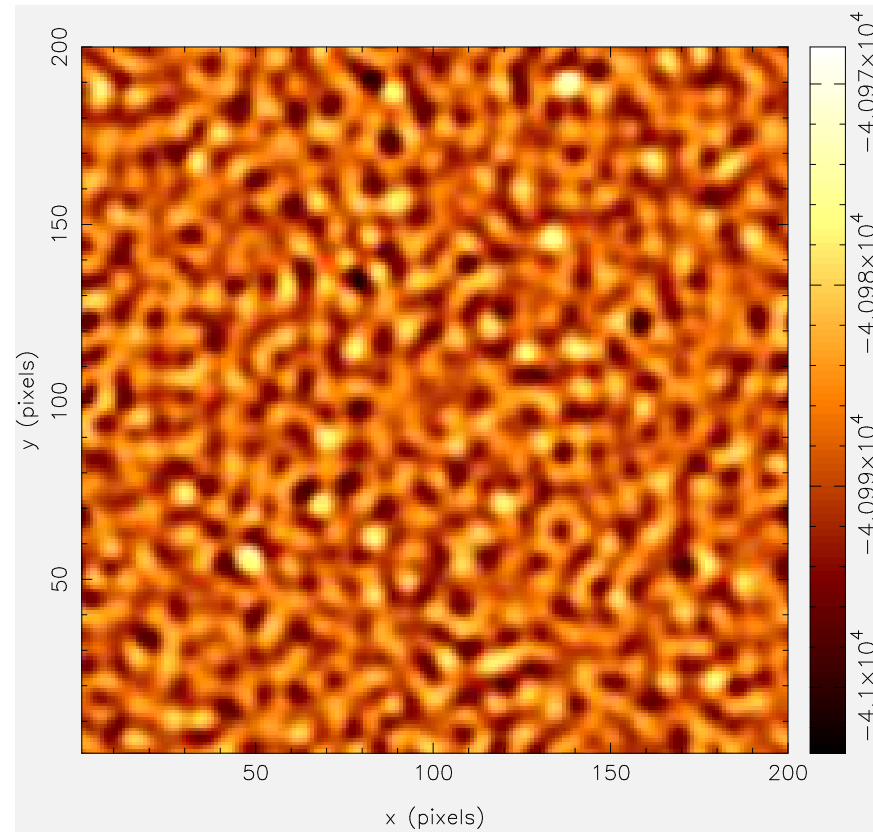- Defines (high) quality of the resulting NN binary classifier

13

2: Nested sampling: fast and reliable parameter space exploration

- Some posteriors are nice, others are nasty
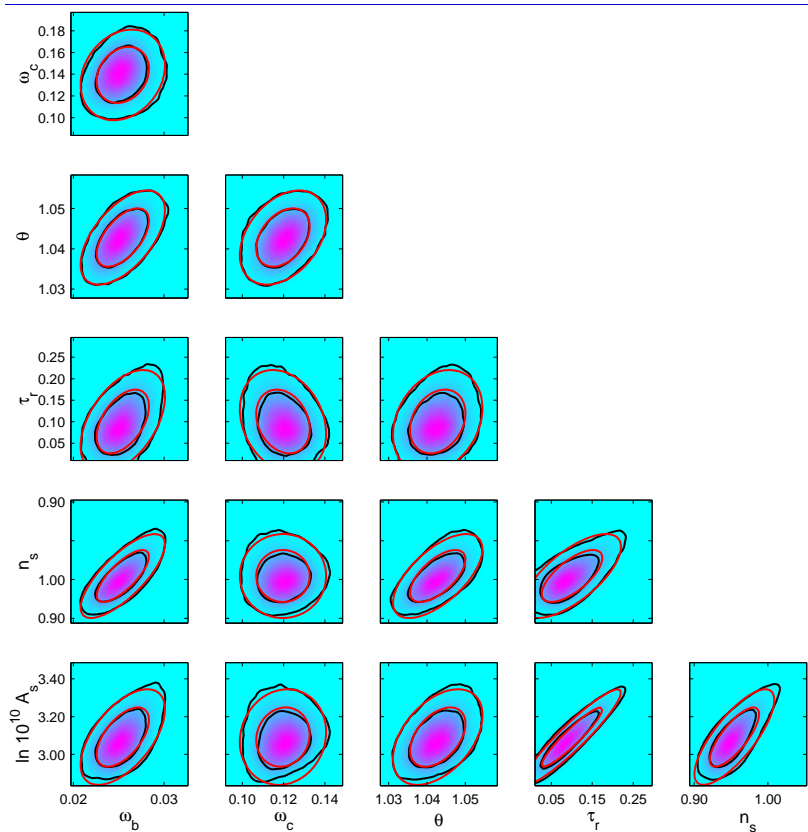
ΛCDM: $\boldsymbol{\theta} = (\omega_b, \omega_c, \theta, \tau, \ln A, n_s)$
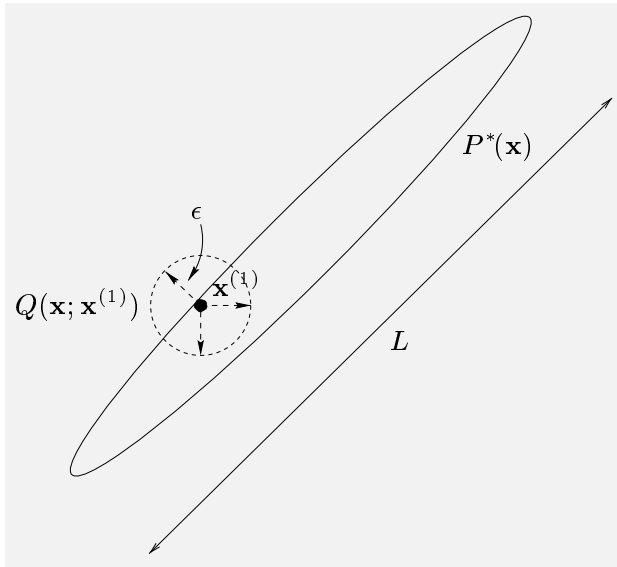using CMB+SDSS+HST data
(Trotta 2004)

Detecting SZ clusters in CMB:
$\boldsymbol{\theta} = (X, Y, A, R)$
(Hobson & McLachlan 2003)

- Posterior exploration (parameter estimation) and integration (model selection) traditionally performed using MCMC sampling

15

- Metropolis–Hastings algorithm to sample $P(\boldsymbol{\theta})$:
    - start at arbitrary point $\boldsymbol{\theta}_0$
    - at each step draw trial point $\boldsymbol{\theta}' \leftarrow Q(\boldsymbol{\theta}'|\boldsymbol{\theta}_n)$ from proposal distribution
    - calculate ratio $r = P(\boldsymbol{\theta}')Q(\boldsymbol{\theta}_n|\boldsymbol{\theta}')/P(\boldsymbol{\theta}_n)Q(\boldsymbol{\theta}'|\boldsymbol{\theta}_n)$
    - if $r \geq 1$ accept $\boldsymbol{\theta}_{n+1} = \boldsymbol{\theta}'$;
      if $r < 1$ accept with probability r, else $\boldsymbol{\theta}_{n+1} = \boldsymbol{\theta}_n$

- Implementation of basic MH algorithm is trivial:

    Initialise $\boldsymbol{\theta}_0$; set $n = 0$
    Repeat [
        Sample a point $\boldsymbol{\theta}'$ from $Q(\cdot|\boldsymbol{\theta}_n)$
        Sample a uniform [0,1] random variable U
        If $U \leq \alpha(\boldsymbol{\theta}', \boldsymbol{\theta}_n)$ set $\boldsymbol{\theta}_{n+1} = \boldsymbol{\theta}'$, else $\boldsymbol{\theta}_{n+1} = \boldsymbol{\theta}_n$
   Increment n]

- After initial burn-in period, any (positive) proposal $Q \Rightarrow$ convergence to $P(\boldsymbol{\theta})$

- Common choice for Q is multivariate Gaussian centred on $\boldsymbol{\theta}_n$ (CosmoMC)

- But... choice of Q strongly affects rate of convergence and sampling efficiency.
- Large proposal width $\epsilon \Rightarrow$ trial points rarely accepted
- Small proposal width $\epsilon \Rightarrow$ chain explores $P(\boldsymbol{\theta})$ by a random walk – very slow
- If largest scale of $P(\boldsymbol{\theta})$ is L

  $\Rightarrow$ typical diffusion time $t \sim (L/\epsilon)^2$
- If smallest scale of $P(\boldsymbol{\theta})$ is ℓ

  $\Rightarrow$ need $\epsilon \sim \ell \Rightarrow$ diffusion time $t \sim (L/\ell)^2$
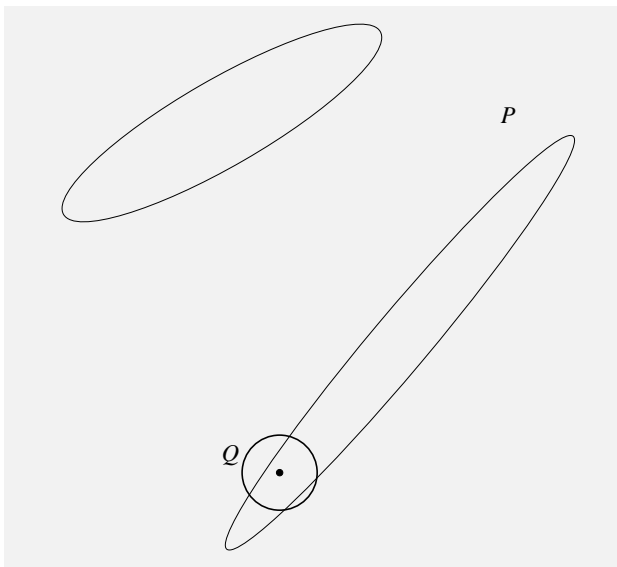
- Particularly bad for multimodal distributions
- Transitions between distant modes very rare
- No choice of proposal width ϵ works
- Standard convergence tests will suggest converged, but actually only true in a subset of modes

17

- Set proposal width ϵ by trial and error to achieve acceptance ratio ~ 0.5, or dynamically during burn-in, but must fix thereafter

- Multiple (non-interacting) chains sometimes useful

- Annealing schedules or multi-temperature chains

- Several sequential proposals: each updating only some parameters

- Innovative proposals, e.g Gibbs, Hamiltonian, slice sampling, genetic algorithms, …

- Compound proposal: multiple proposals Q_i each chosen at random with probability p_i

- Use of multiple *interacting* chains, e.g.
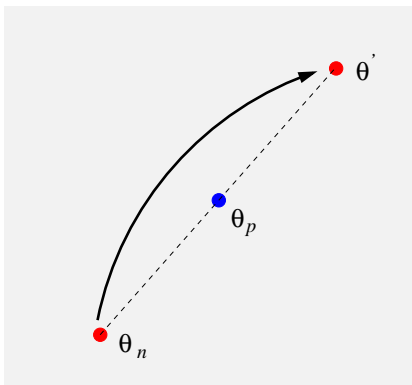
leapfrog
$$\theta' = 2\theta_{\mathrm{p}} - \theta_n$$
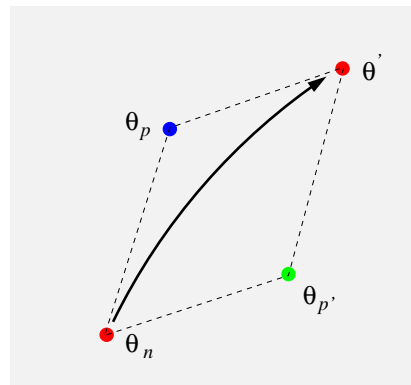
cross-walk
$$\theta' = \theta_{\mathrm{p}} + \theta_{\mathrm{p}'} - \theta_n$$

guided-walk
$$\theta' = \theta_n + (\theta_{\mathrm{p}} - \theta_{\mathrm{p}'})$$

18
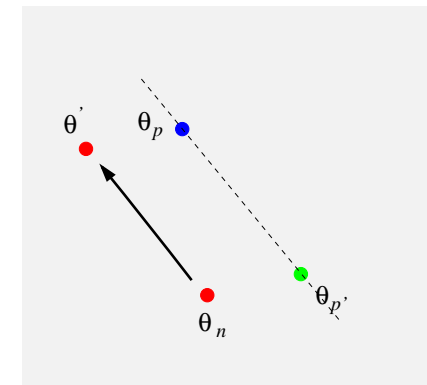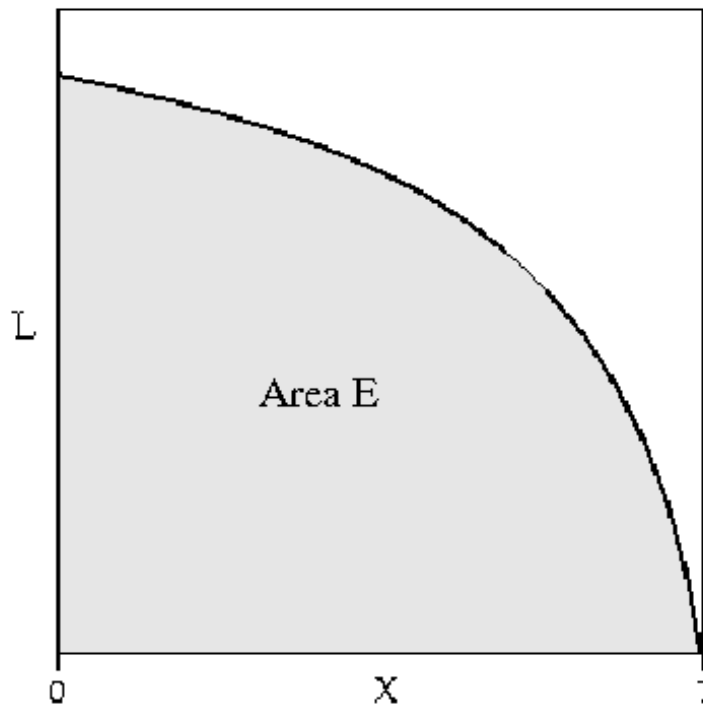
- New technique for efficient evidence evaluation (and posterior samples) (Skilling 2004)

- Define $X(\lambda) = \int_{L(\boldsymbol{\theta}) > \lambda} \pi(\boldsymbol{\theta}) \, d\boldsymbol{\theta}$

- Write inverse $L(X)$, i.e. $L(X(\lambda)) = \lambda$

- Evidence becomes one-dimensional integral

$$E = \int L(\boldsymbol{\theta})\pi(\boldsymbol{\theta}) \, d\boldsymbol{\theta} = \int_0^1 L(X) \, dX$$

- Suppose can evaluate $L_j = L(X_j)$ where $0 < X_m < \cdots < X_2 < X_1 < 1$
$\Rightarrow$ estimate E by any numerical method

$$E = \sum_{j=1}^{m} L_j w_j$$

$(w_j = \frac{1}{2}(X_{j-1} - X_{j+1})$ for trapezium rule$)$

Area E

Nested sampling approach to summation:
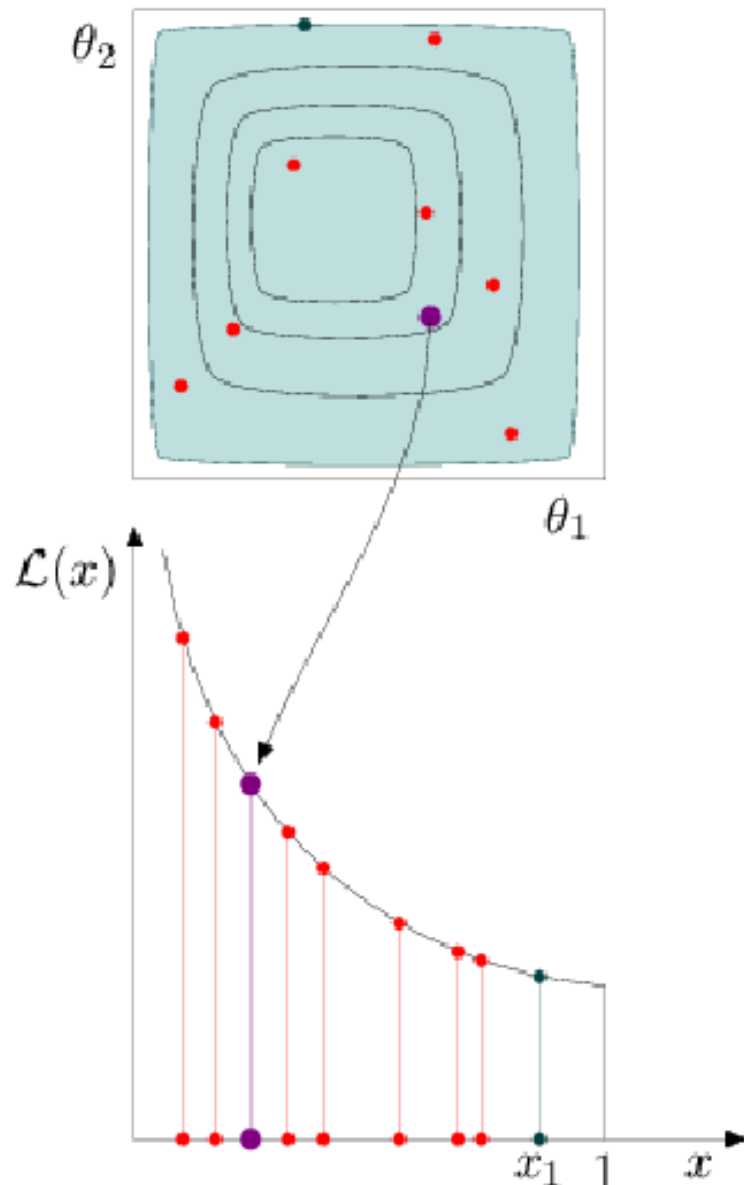
1. Set $i = 0$; initially $X_0 = 1$, $E = 0$

2. Sample N points $\{\boldsymbol{\theta}_j\}$ randomly from $\pi(\boldsymbol{\theta})$ and calculate their likelihoods

3. Set $i \rightarrow i + 1$

4. Find point with lowest likelihood value (L_i)

5. Remaining prior volume $X_i = t_i X_{i-1}$ where $\Pr(t_i | N) = N t_i^{N-1}$; or just use $\langle t_i \rangle = N/(N+1)$

6. Increment evidence $E \rightarrow E + L_i w_i$

7. Remove lowest point from active set

8. Replace with new point sampled from $\pi(\boldsymbol{\theta})$ within hard-edged region $L(\boldsymbol{\theta}) > L_i$

9. If $L_{\mathsf{max}} X_i < \alpha E$ (where some tolerance)
   $\Rightarrow E \rightarrow E + X_i \sum_{j=1}^{N} L(\boldsymbol{\theta}_j)/N$; stop

   else goto 3

- Advantages:
  - typically requires around few 100 times fewer samples than thermodynamic integration to calculate evidence to same accuracy (plus error estimate)

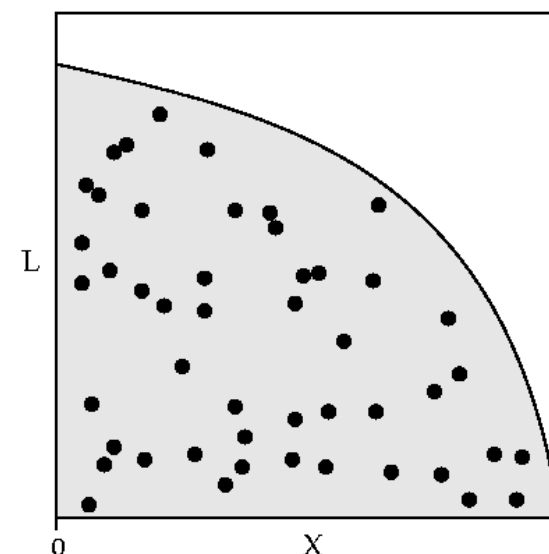 - does not get stuck at phase changes like thermodynamic integration

(a)

(b)
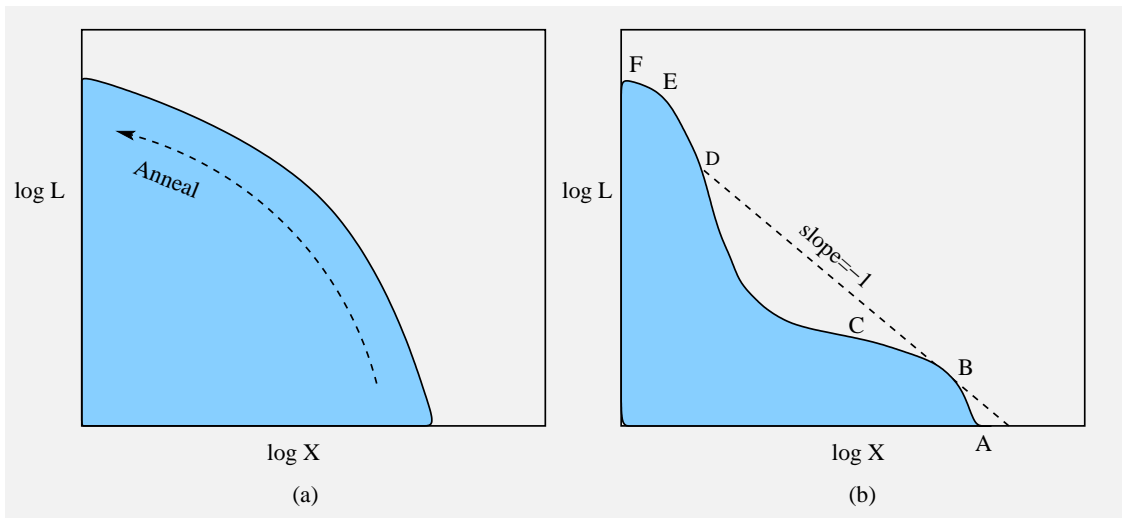
- As $\lambda : 0 \to 1$ annealing should track along curve

- But $\frac{d \log L}{d \log X} = -\frac{1}{\lambda}$, so annealing schedule cannot navigate convex regions (phase changes)

- Bonus: posterior samples easily obtained as a by-product. Simply take full sequence of sampled points $\boldsymbol{\theta}_j$ and weight jth sample by $p_j = L_j w_j / E$, e.g.

$$\mu_Q = \sum_j p_j Q(\boldsymbol{\theta}_j),$$

$$\sigma_Q^2 = \sum_j (p_j Q(\boldsymbol{\theta}_j) - \mu_Q)^2$$


21

- Most challenging task: at each iteration i must replace removed point with one sampled from $\pi(\boldsymbol{\theta})$ within complicated, hard-edged region $L(\boldsymbol{\theta}) > L_i$

- Simple MCMC using Metropolis–Hastings possible, but can be inefficient

- Mukherjee et al. (2005) fit ellipsoid to active points, enlarge to try to account for non-ellipsoidal likelihood contour, and sample within it using simple, exact method
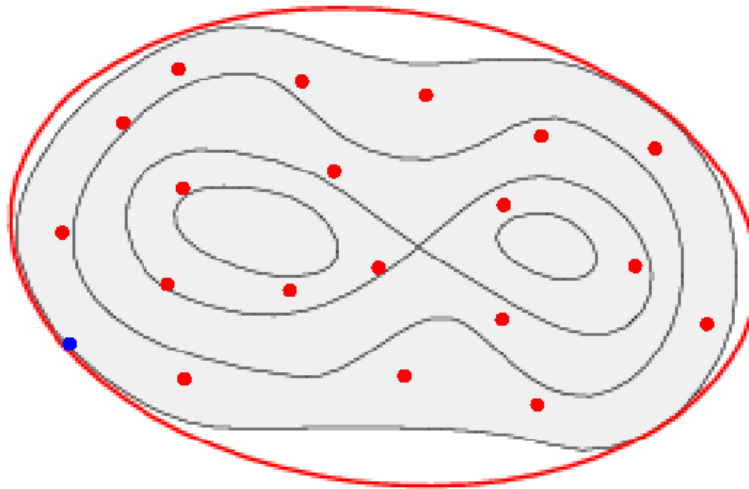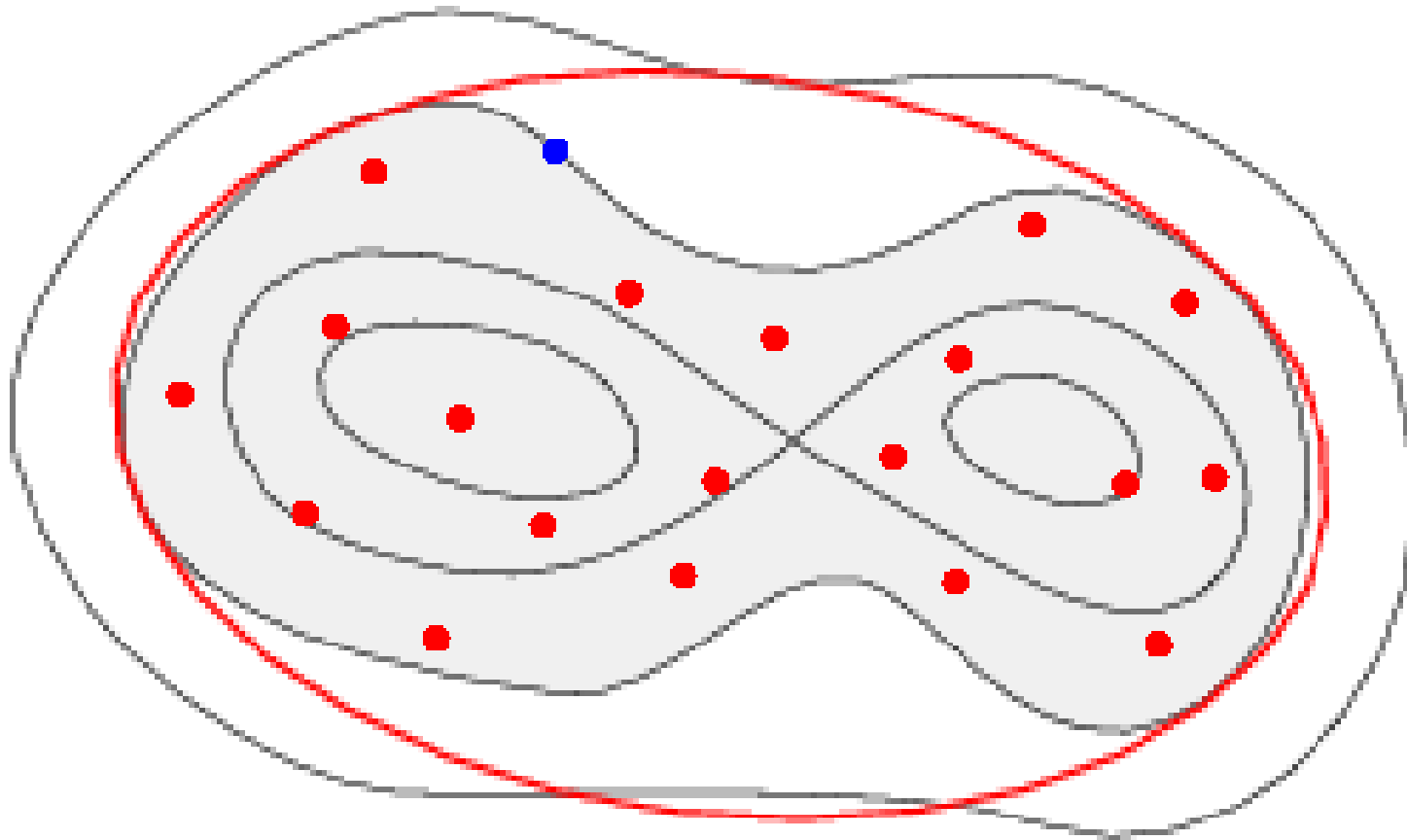

- Demonstrated high-efficiency and robustness on simple unimodal cosmological posteriors (~ 100 times faster evidence evaluation cf. thermodynamic integration)

- But. . . still problematic for multimodal/ degenerate posteriors
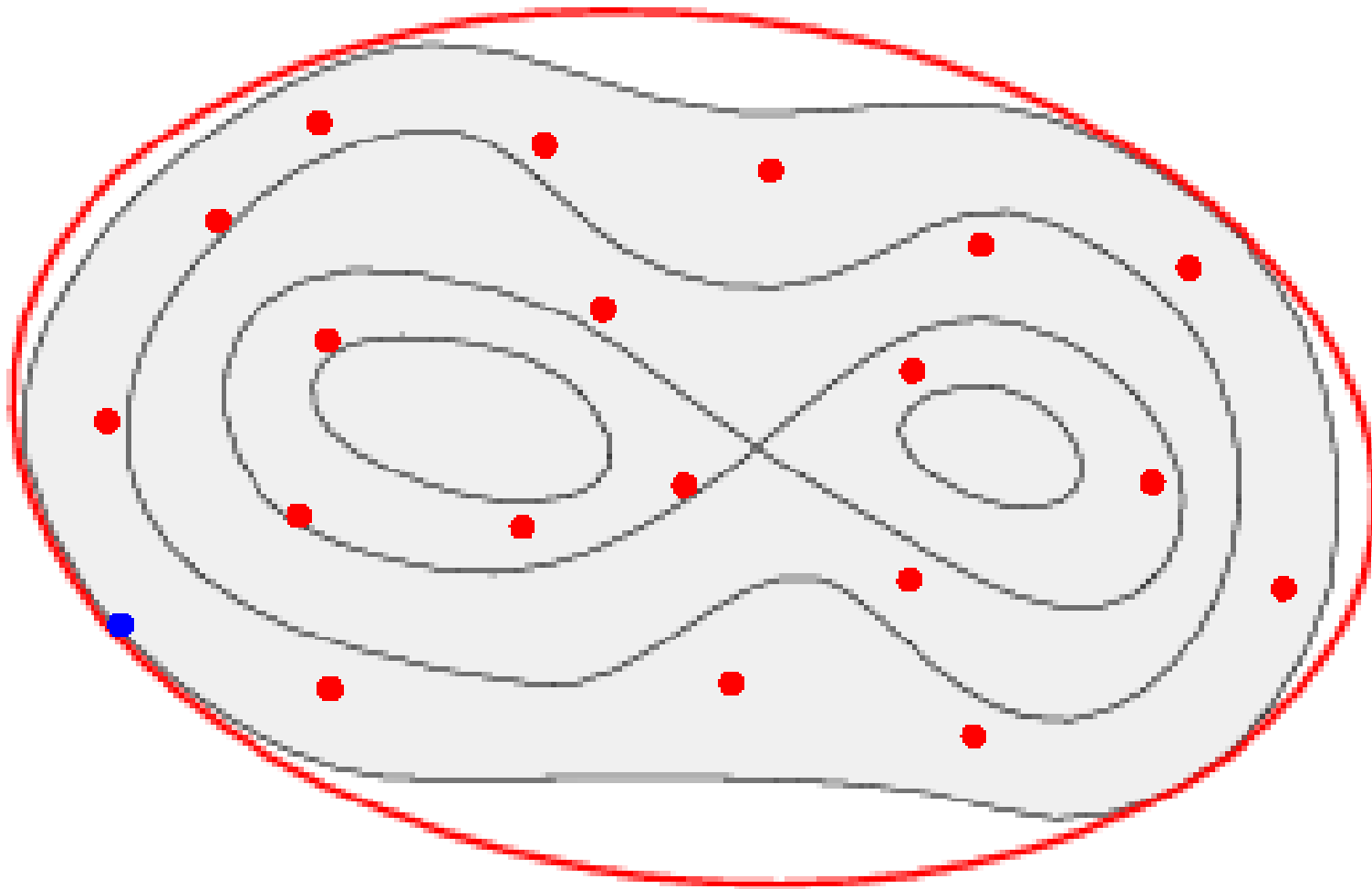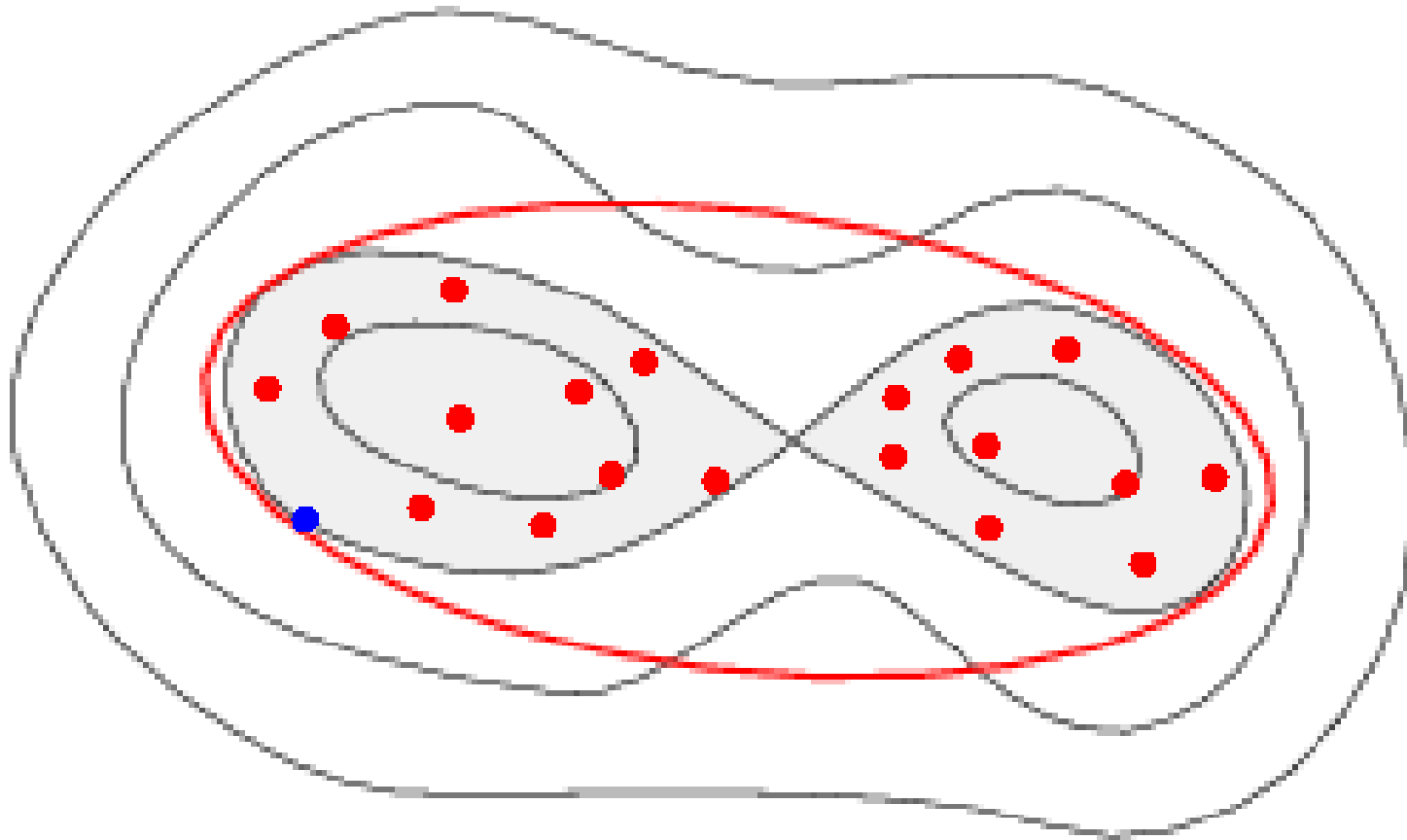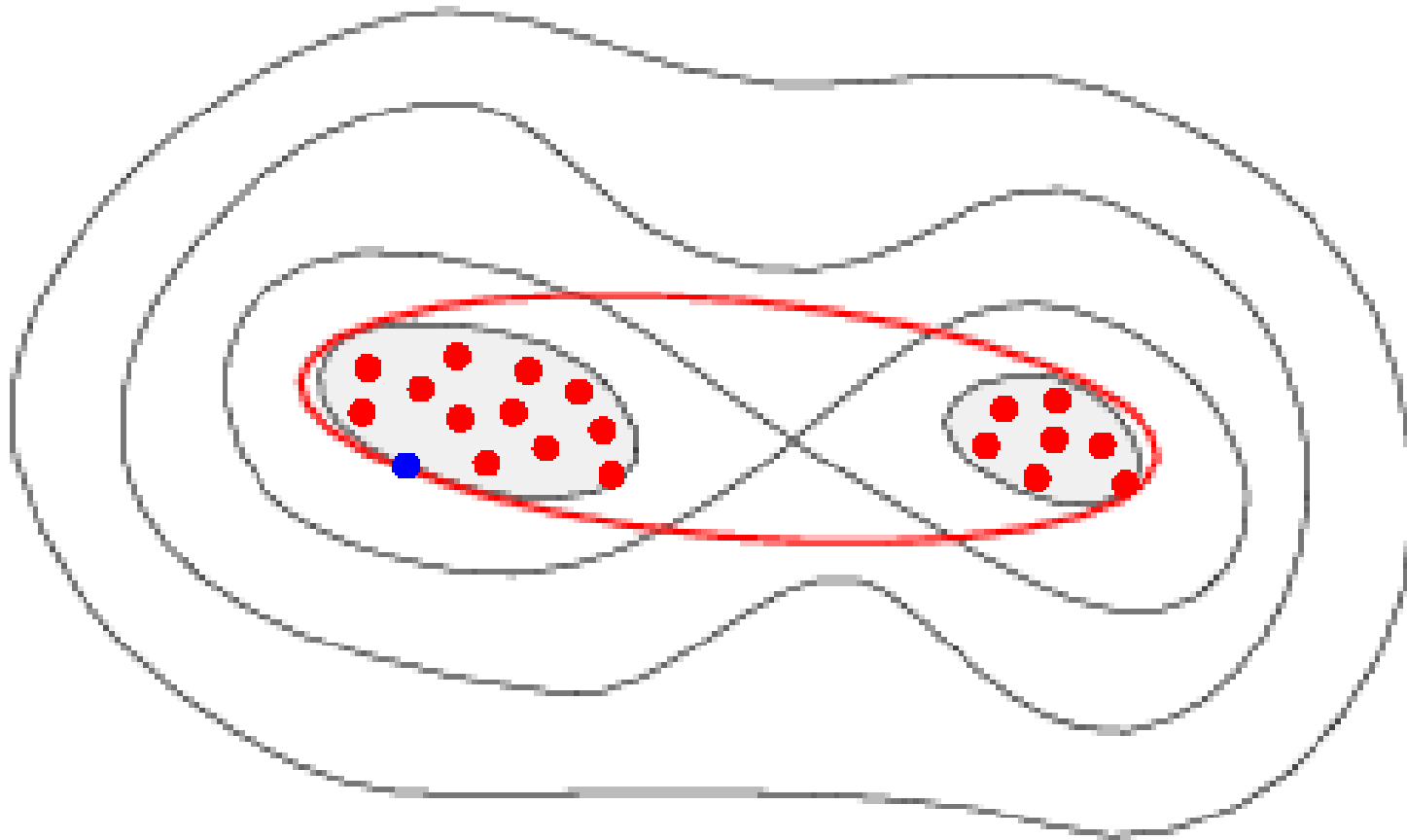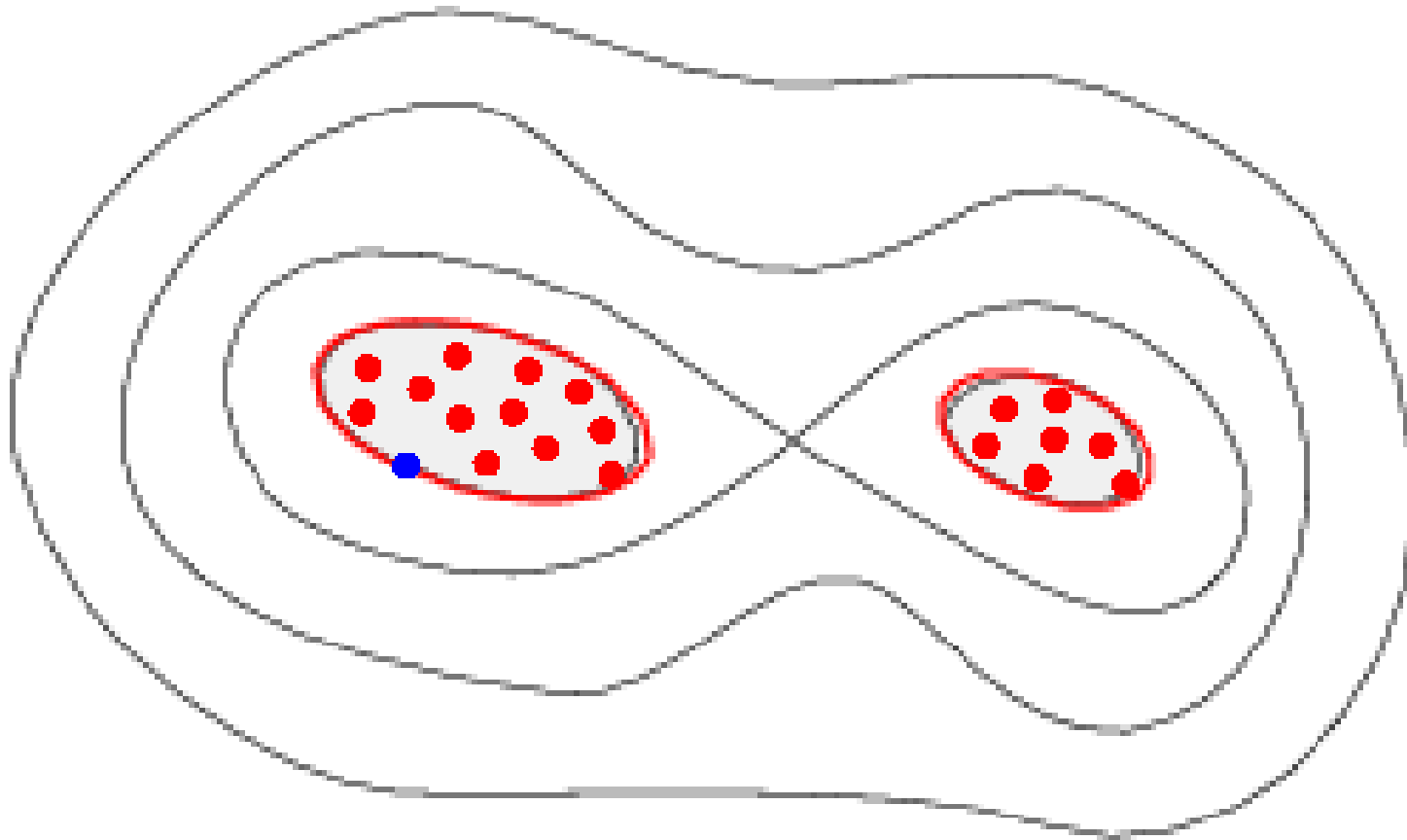
22

Problem with elliptical region sampling ($N = 20$):

Problem with elliptical region sampling ($N = 20$):

**Problem with elliptical region sampling ($N = 20$):**

Problem with elliptical region sampling ($N = 20$):

Problem with elliptical region sampling ($N = 20$):

- Introduced by Feroz & MPH (2008), refined by Feroz, MPH & Bridges (2008)

- At each nested sampling iteration i:
- construct optimal multi-ellipsoidal bound for each cluster (variable ellipsoid number), or evolve existing decomposition via scaling (fast)
- determine ellipsoid overlaps using cheap exact algorithm (Alfano et al. 2003)
- remove point with lowest L_i from active points; increment evidence
- pick ellipsoid randomly and sample new point with $L > L_i$, accounting for overlaps
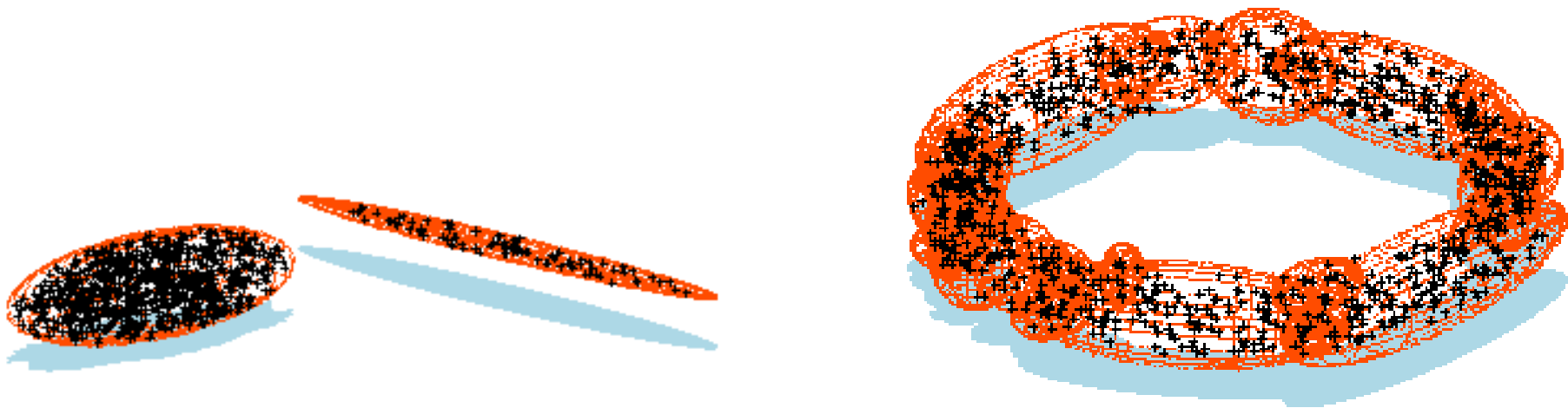

- MULTINEST algorithm usefully (and easily) parallelized
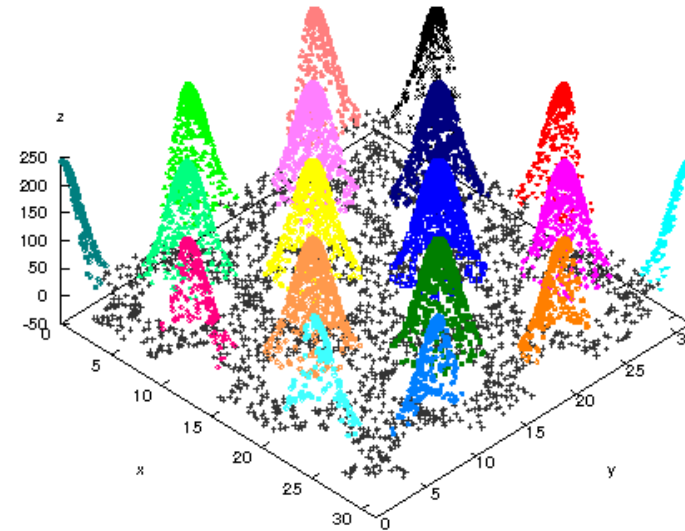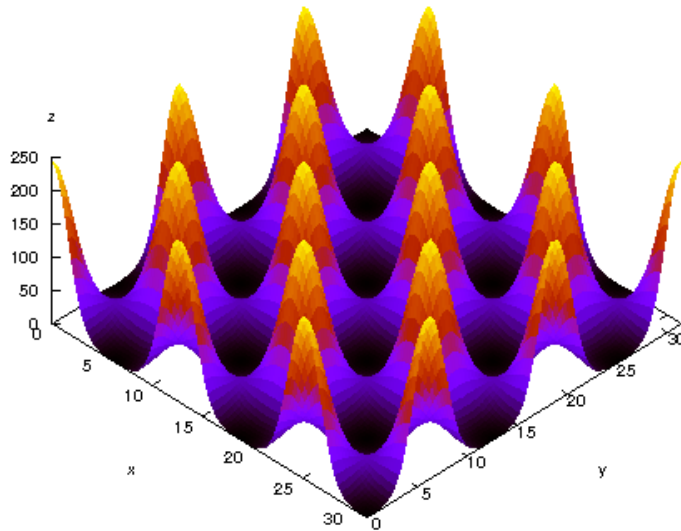
- For multimodal posteriors, useful to identify which samples 'belong' to which mode

- For well-defined 'isolated' modes:
  – can make reasonable estimate of posterior mass each contains ('local' evidence)
  – can construct posterior parameter constraints associated with each mode

- Partitioning and ellipsoids construction algorithm described above provides efficient and reliable method for performing mode identification
  ⇒ 'local' evidence and parameter constraints for each isolated mode
  ⇒ sum of local evidences equals 'global' evidence

29

3: Application to SUSY phenomenology

- MULTINEST applied to cMSSM and pMSSM (by us)
  (see arXiv:0807.4512, arXiv:0809.3792, arXiv:0903.2487, arXiv0904.2548,
  arXiv0906.0957, arXiv:1101.3296)

- In all cases, MULTINEST is few $\times$ 100 more efficient than MCMC

- NNs and MULTINEST in cMSSM around benchmark point (arXiv:1011.4306)
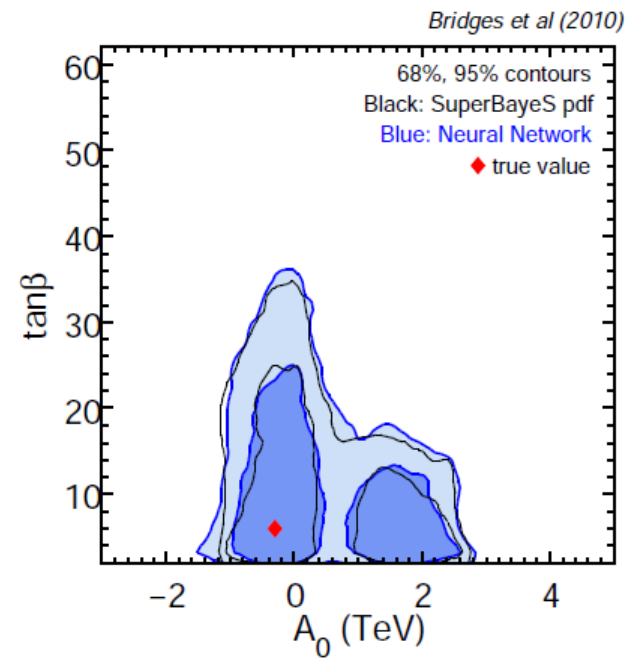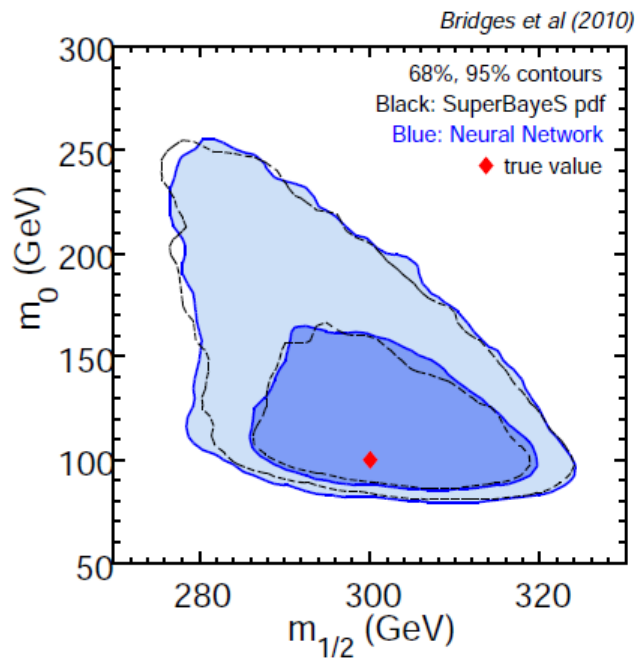
- Speeds up analysis by factor $\sim 10^6$ (NN by $\sim 10^4$ and MULTINEST by $\sim 10^2$)
  $\Rightarrow$ original SOFTSUSY + MCMC = 720 CPU days; NN + MULTINEST = 1 minute

- Bayesians believe in posterior mass
  $\Rightarrow$ MULTINEST produces samples drawn according to full 8D posterior $\Theta = (\boldsymbol{\theta}, \boldsymbol{\psi})$
  $\Rightarrow$ calculate marginalised posteriors for parameters of interest

$$P(\boldsymbol{\theta}) = \int P(\boldsymbol{\theta}, \boldsymbol{\psi}) \, d\psi$$

- Frequentists believe in likelihood global maximum
  $\Rightarrow$ calculate profile likelihood for parameters of interest

$$\lambda(\boldsymbol{\theta}) \equiv \frac{L(\boldsymbol{\theta}, \hat{\hat{\boldsymbol{\psi}}})}{L(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\psi}})}$$

where $\hat{\hat{\boldsymbol{\psi}}}$ is conditional MLE with $\boldsymbol{\theta}$ fixed, and $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\psi}}$ are unconditional MLEs

- Standard MULTINEST configuration good for accurate reconstruction of Bayesian posterior (spikes unimportant), BUT profile likelihood poorly approximated

- Require more live points and later termination criterion
  $\Rightarrow$ allows MULTINEST to explore high-likelihood regions (including spikes)
  $\Rightarrow$ significantly better approximation to profile likelihood than latest genetic algorithm method, particularly near boundaries of the interval

● MULTINEST Bayesian marginalised posteriors and profile likelihoods in cMSSM
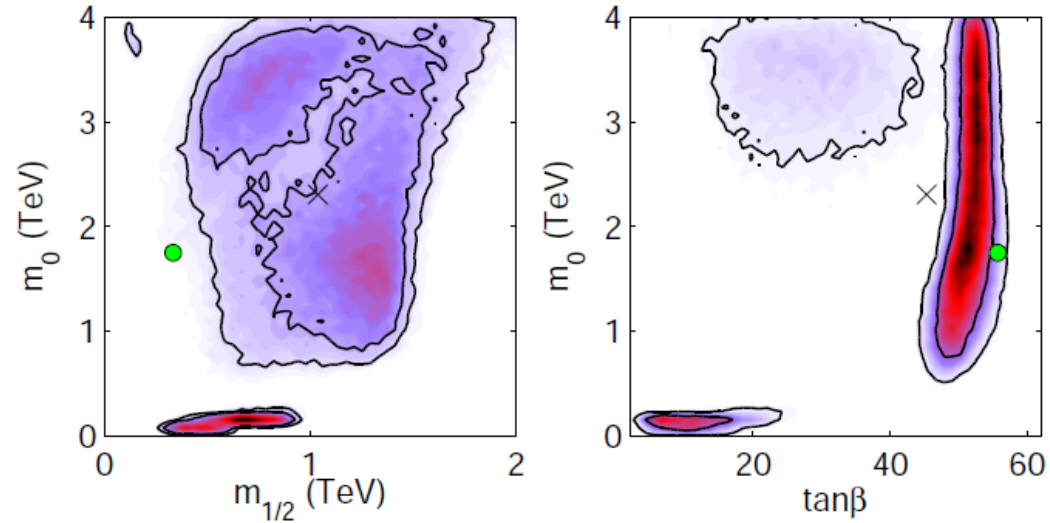
(a)
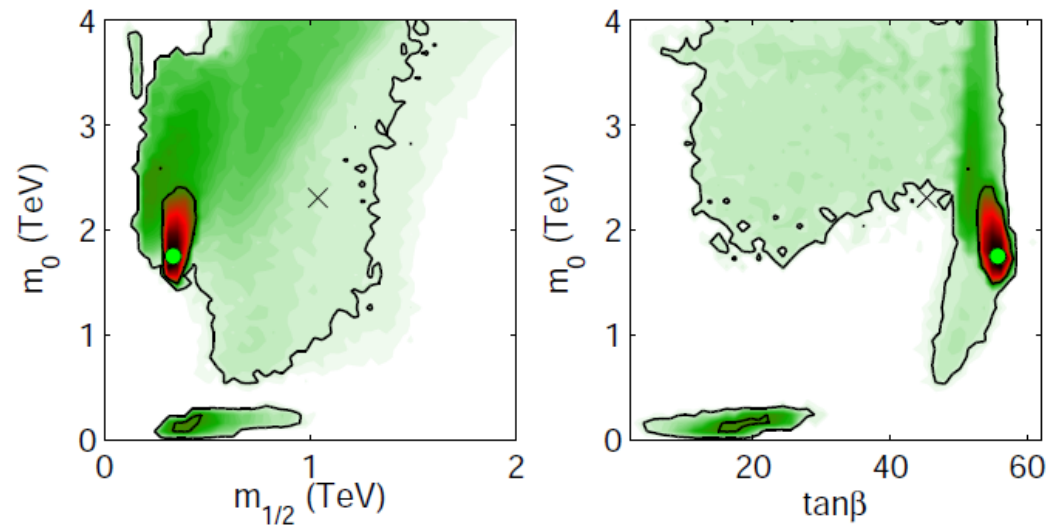
4: The future: BAMBI…

BLIND ACCELERATED MUTLIMODAL BAYESIAN INFERENCE (BAMBI)

- General Bayesian inference engine with wide applicability: only requires choice of priors on the parameters in model

- Combines neural networks and nested sampling in complementary manner

- Basic idea is as follows:

  - early stage (prior-driven) nested samples $\Rightarrow$ (incremental) training data set

  - simultaneous training of neural network $\Rightarrow$ 'learn' likelihood function

  - clustering in nested sampler $\Rightarrow$ accelerates network training

  - once trained, network replaces likelihood code
    $\Rightarrow$ completes posterior sampling and evidence evaluation extremely rapidly

  - trained likelihood network available for subsequent analyses

CONCLUSIONS

- Standard Bayesian or Frequentist analyses can be very computationally intensive: days–weeks on a supercomputer

- Large speed-ups possible using neural networks for model prediction

- Efficient and robust parameter space exploration provided by nested sampling
  – MULTINEST allows sampling from multimodal/degenerate posteriors
  – typically few $\times$ 100 times more efficient than standard MCMC

- These methods should be useful in a wide range of physical inference problems; already applied in many areas

- COSMONET and MULTINEST code publically available from:
  `www.mrao.cam.ac.uk/software/cosmonet`
  `www.mrao.cam.ac.uk/software/multinest`

- BAMBI in development. . .

Supplementary slides

ADVANTAGES OF COSMONET

- **Simplicity:** provides single, simple, closed-form function for each interpolation over **entire** parameter space

- **Memory usage:** a network with N_i input nodes, N_h hidden nodes and N_o output nodes has $(N_i + 1)N_h + (N_h + 1)N_o \approx N_h N_o$ parameters. For above model, requires only ~ 50 kB of parameter memory

- **Accuracy:** excellent after only $\sim$ few mins of training on single 2GHz CPU

- **Speed:** number of calculations to perform feed-forward network mapping is $2N_i N_h + 2N_h N_o \approx 2N_h N_o$. In above example, calculation of C_ℓ spectrum in $\sim$ 20 microseconds, and WMAP likelihood in ~ 5 microseconds

- **Scaling:** N_h increases at worst **linearly** with N_i

- Algorithm for partitioning active points into clusters and constructing ellipsoidal bounds requires uniformly distributed points

- MULTINEST 'native' space = D-dimensional unit hypercube in which samples are drawn uniformly. All operations are carried out in this space (cf. BAYESYS).

- To conserve probability mass, point $\boldsymbol{u} = (u_1, u_2, \cdots, u_D)$ in unit hypercube transformed point $\boldsymbol{\Theta} = (\theta_1, \theta_2, \cdots, \theta_D)$ in 'physical' parameter space, such that

$$\int \pi(\theta_1, \theta_2, \cdots, \theta_D)\, d\theta_1\, d\theta_2 \cdots d\theta_D = \int du_1 du_2 \cdots du_D$$

- In simple case that prior separable: $\pi(\boldsymbol{\Theta}) = \pi_1(\theta_1)\pi_2(\theta_2) \cdots \pi_D(\theta_D)$, set $\pi_j(\theta_j)d\theta_j = du_j \Rightarrow$ for given u_j, find θ_j by solving

$$u_j = \int_{-\infty}^{\theta_j} \pi_j(\theta_j')d\theta_j'$$

- If prior $\pi(\Theta)$ not separable, instead write

$$\pi(\theta_1, \theta_2, \cdots, \theta_D) = \pi_1(\theta_1)\pi_2(\theta_2|\theta_1) \cdots \pi_D(\theta_D|\theta_1, \theta_2 \cdots \theta_{D-1})$$

where

$$\pi_j(\theta_j|\theta_1, \cdots, \theta_{j-1}) = \int \pi(\theta_1, \cdots, \theta_{j-1}, \theta_j, \theta_{j+1}, \cdots, \theta_D) \, d\theta_{j+1} \cdots d\theta_D$$

- Physical point Θ corresponding to point u in unit hypercube then found by using this π_j in earlier expression

- Physical parameters Θ used to calculate likelihood of point u
  For many problems, prior $\pi(\Theta)$ is uniform $\Rightarrow u$ and Θ-spaces coincide
  For many other problems, prior $\pi(\Theta)$ allows one to solve for Θ point analytically

- In all cases, can solve for Θ point numerically

- Alternatively... re-cast inference problem: for example, define new 'likelihood' $\mathcal{L}'(\Theta) \equiv \mathcal{L}(\Theta)\pi(\Theta)$ and 'prior' $\pi'(\Theta) \equiv$ constant. But potentially inefficient since lacks true prior $\pi(\Theta)$ to guide the sampling of active points

40

- At ith NS iteration, find 'optimal' ellipsoidal decomposition of N active points distributed uniformly in remaining prior volume X_i using EM approach

- Let set of N active points in unit hypercube be $S = \{u_1, u_2, \cdots, u_N\}$ and some partitioning into K clusters be $\{S_k\}_{k=1}^K$, where $K \geq 1$ and $\cup_{k=1}^K S_k = S$.

- For cluster (or subset) S_k containing n_k points, define quasi-minimum-volume bounding ellipsoid

$$E_k = \{u \in \mathcal{R}^{\mathsf{D}} | u^{\mathsf{T}} (f_k C_k)^{-1} u \leq 1\},$$

where the empirical covariance matrix of the subset is

$$C_k = \frac{1}{n_k} \sum_{j=1}^{n_k} (u_j - mu_k)(u_j - mu_k)^{\mathsf{T}}$$

and $mu_k = \sum_{j=1}^{n_k} u_j$ is its center of the mass. Enlargement factor f_k ensures E_k is a bounding ellipsoid. Note: volume of ellipsoid $V(E_k) \propto \sqrt{\det(f_k C_k)}$

41

- At ith NS iteration, volume $V(S)$ from which set S uniformly sampled is unknown remaining prior volume X_i, but use expectation value $V(S) = \exp(-i/N)$

- Define objective function

$$F(S) \equiv \frac{1}{V(S)} \sum_{k=1}^{K} V(E_k)$$

and minimise $F(S)$, subject to the constraint $F(S) \geq 1$, wrt K-partitionings $\{S_k\}_{k=1}^{K} \Rightarrow$ 'optimal' decomposition of original sampled region into K ellipsoids

- Minimisation most easily performed using EM scheme, using result (Lu et al. 2007) that, change in $F(S)$ resulting from reassigning a point $\boldsymbol{u}$ from subset S_k to $S_{k'}$ is

$$\Delta F(S)_{k,k'} \approx \gamma \left(\frac{V(E_{k'})d(\boldsymbol{u}, S_{k'})}{V(S_{k'})} - \frac{V(E_k)d(\boldsymbol{u}, S_k)}{V(S_k)} \right)$$

where γ is a constant,

$$d(\boldsymbol{u}, S_k) = (\boldsymbol{u} - \boldsymbol{mu}_k)^{\top}(f_k \mathbf{C}_k)^{-1}(\boldsymbol{u} - \boldsymbol{mu}_k)$$

is 'distance' from $\boldsymbol{u}$ to centroid $\boldsymbol{mu}_k$ of ellipsoid E_k, and

$$V(S_k) = \frac{n_k V(S)}{N}$$

may be considered the volume from which subset S_k was drawn uniformly

- In fact, impose further constraint that $V(E_k) > V(S_k)$. Easily achieved by enlarging ellipsoid E_k by factor f_k, such that $V(E_k) = \max[V(E_k), V(S_k)]$, before evaluating $F(S)$ and $\Delta F(S)_{k,k'}$

- Minimising $F(S)$ equivalent to defining

$$h_k(\boldsymbol{u}) = \frac{V(E_k)d(\boldsymbol{u}, S_k)}{V(S_k)}$$

and, for all points $\boldsymbol{u} \in S$, assigning $\boldsymbol{u} \in S_k$ to $S_{k'}$ only if $h_k(\boldsymbol{u}) < h_{k'}(\boldsymbol{u})$, $\forall \, k \neq k'$, and repeating until convergence is achieved

- To find optimal number of ellipsoids, K, use recursive scheme:
  – start by performing k-means partition with $K = 2$
  – optimise this 2-partition as outlined above,
  – recursively partition and optimise the resulting clusters
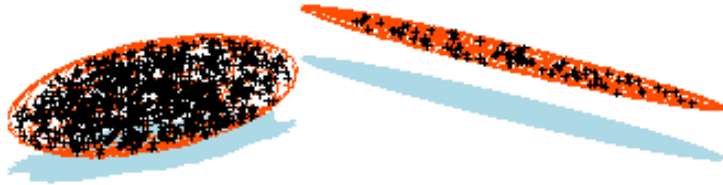
1000 points drawn from two ellipsoids

1. For S, calculate bounding ellipsoid E and $V(E)$

2. Enlarge E so that $V(E) = \max[V(E), V(S)]$

3. Partition S into S_1 and S_2 containing n_1 and n_2 points using $k-$means with $K = 2$

4. Calculate E_1, E_2 and volumes $V(E_1)$, $V(E_2)$

5. Enlarge E_k $(k = 1, 2)$ so that $V(E_k) = \max[V(E_k), V(S_k)]$.

6. For all $u \in S$, assign u to S_k such that $h_k(u) = \min[h_1(x), h_2(x)]$

7. If no point reassigned goto 8; else goto 4

8. If $V(E_1) + V(E_2) < V(E)$ or $V(E) > 2V(S)$
   – partition S into S_1 and S_2
   – repeat entire algortihm for each subset S_1 and S_2
   else
   – return E as the optimal ellipsoid of the point set S

1000 points drawn from a torus

- EM algorithm quite computationally expensive, especially in high dimensions

- But... MULTINEST need not perform full partitioning at each NS iteration

- Ellipsoids can be evolved through scaling at subsequent NS iterations $i + i'$ such that $V(E_k) = \max[V(E_k), X_{i+i'} n_k / N]$

- Ellipsoidal decomposition calculated at iteration i becomes less optimal as i' grows $\Rightarrow$ perform full re-partitioning of active points if $F(S) \geq h$ (typically $h = 1.1$)

- Possible that ellipsoids might not enclose the entire iso-likelihood contour, even though sum of their volumes must exceed prior volume $X \Rightarrow$ safer to set desired minimum volume as eX, where e is an enlargement factor

- Note: regardless of e-value, always ensure that E_k is a bounding ellipsoid of subset S_k.
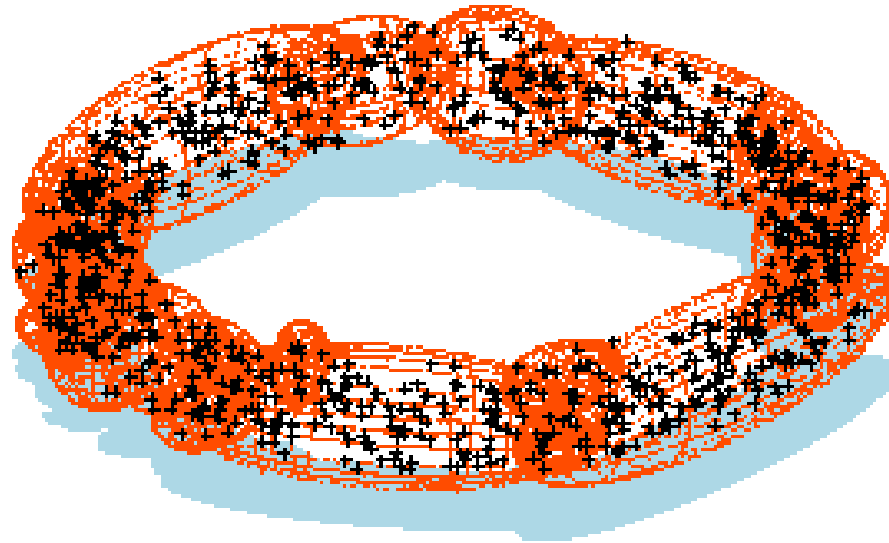
- At each NS iteration, need to draw a new point uniformly from union of ellipsoids

- k Suppose K ellipsoids $\{E_k\}$, where kth one has volume $V(E_k)$

- Choose one ellipsoid with probability $p_k = V_k/V_{\text{tot}}$

- Sample from chosen ellipsoid within hard constraint $L > L_i$

- Find number n_e of ellipsoids in which sample lies; accept with probability $1/n_e$

TRIVIAL PARALLELIZATION

- Typical sampling efficiency less than unity since
  – ellipsoidal approximation to iso-likelihood surface not perfect
  – ellipsoids may overlap (as discussed above)

- But... MULTINEST algorithm usefully (and easily) parallelized

- At each NS iteration, draw a potential replacement point on each of N_{CPU} processors, where $1/N_{\mathsf{CPU}}$ is an estimate of the sampling efficiency

$\Rightarrow$ Effective efficiency close to unity across N_{CPU}
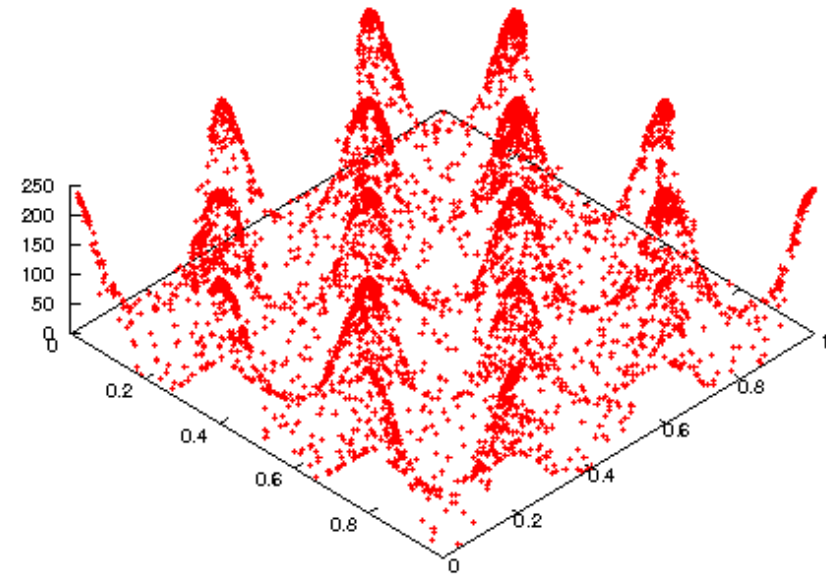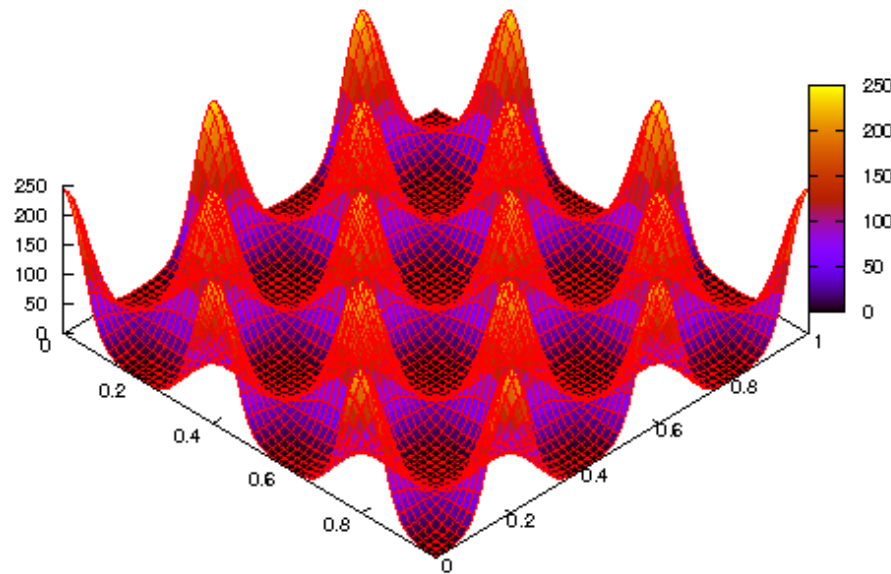
- For multimodal posteriors, useful to identify which samples 'belong' to which mode

- Some arbitrariness in this process: modes sit on top of some general 'background' of probability distribution

- Moreover, modes lying close together may only 'separate out' at relatively high likelihood levels
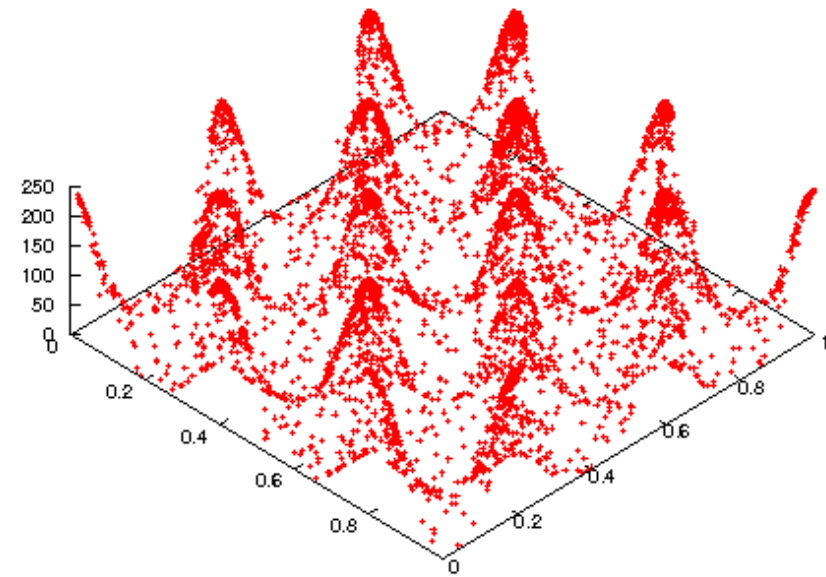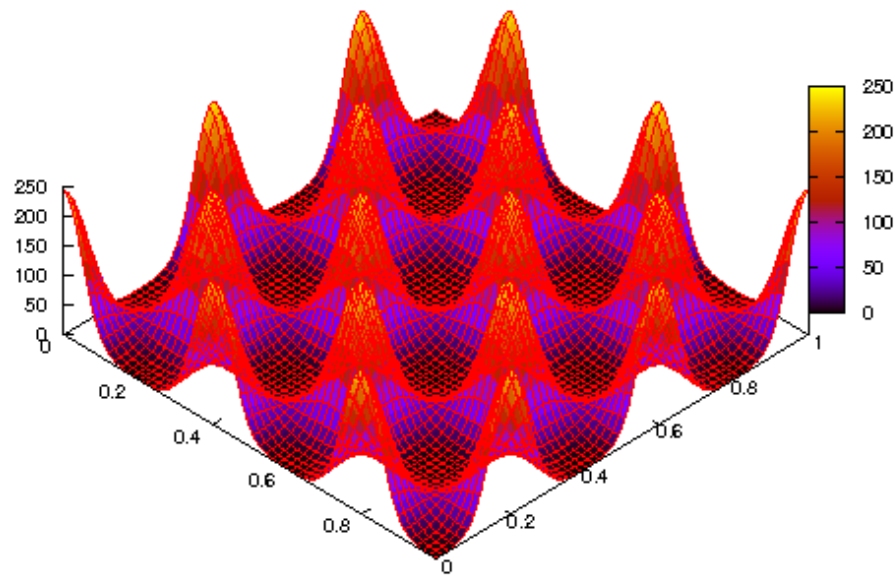
- Nonetheless, for well-defined 'isolated' modes:
  – can make reasonable estimate of posterior mass each contains ('local' evidence)
  – can construct posterior parameter constraints associated with each mode

- Once NS process reached likelihood such that 'footprint' of mode well-defined ⇒ identify at each subsequent iteration the points in active set belonging to mode

- Partitioning and ellipsoids construction algorithm described above provides efficient and reliable method for performing this identification
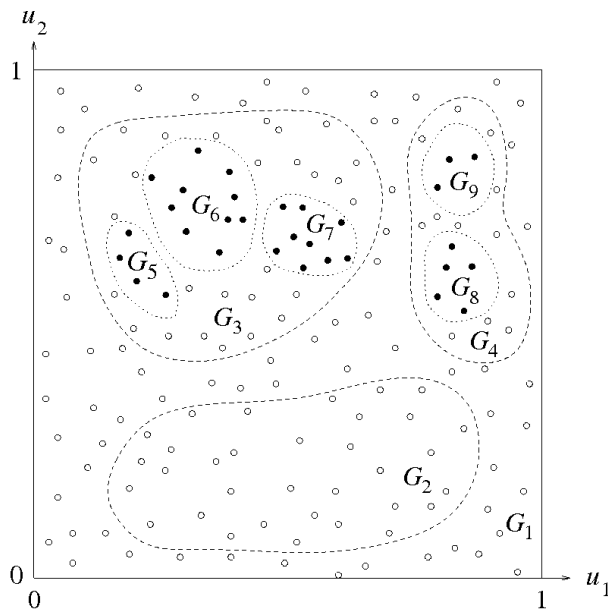
1. In first NS iteration, assign all active points to active group G_1

2. In subsequent NS iterations, pick subset S_k of G_1 at random:
   – S_k points become first members of 'temporary set' $\mathcal{T}$
   – E_k becomes first member of 'ellipsoid set' $\mathcal{E}$

3. For all $E_{k'} \notin \mathcal{E}$, determine if $E_{k'}$ intersects any ellipsoid in $\mathcal{E}$

4. If no such intersections occur:
   – goto 5
   else, for each such intersecting ellipsoid $E_{k'}$:
   – add $S_{k'}$ points to $\mathcal{T}$ and add $E_{k'}$ to $\mathcal{E}$
   – goto 3

5. If all ellipsoids are members of $\mathcal{E}$:
   – (re)assign points in $\mathcal{T}$ to G_1
   else
   – (re)assign points in $\mathcal{T}$ to new active group G_2
   – (re)assign remaining active points to new active group G_3
   – group G_1 becomes 'inactive'

6. In current NS iteration, goto 2 and repeat algorithm for each active group until no new active groups occur

7. In subsequent NS iterations, apply algorithm to each active group

50
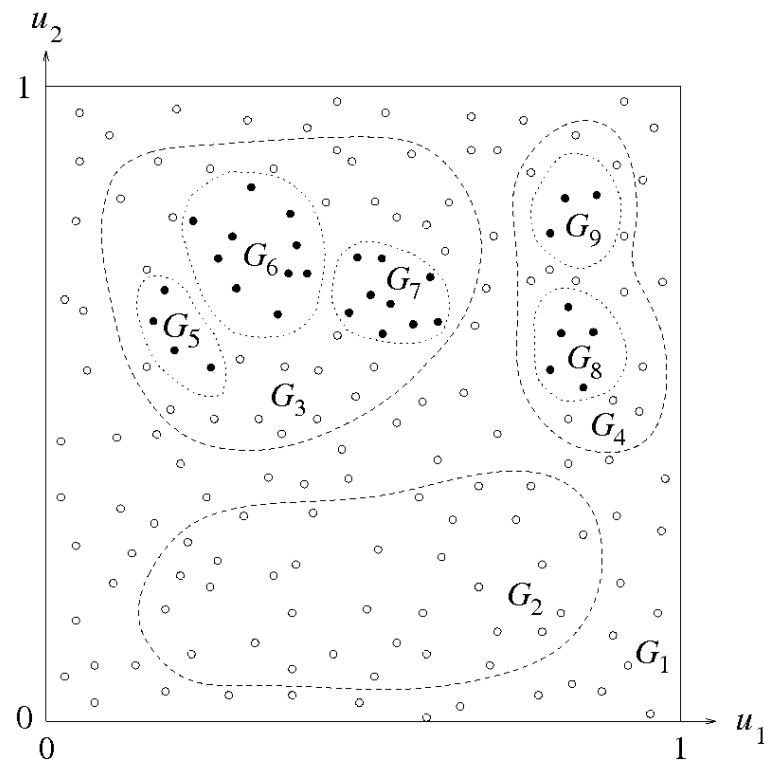
- At end of NS process $\Rightarrow$ set of inactive groups and set of active groups, which together partition the full set of (inactive and active) sample points generated

- Note: as NS process reaches higher likelihoods, number of active points in any particular active group may dwindle to zero, but... group still considered active since it remains unsplit at the end of NS run.

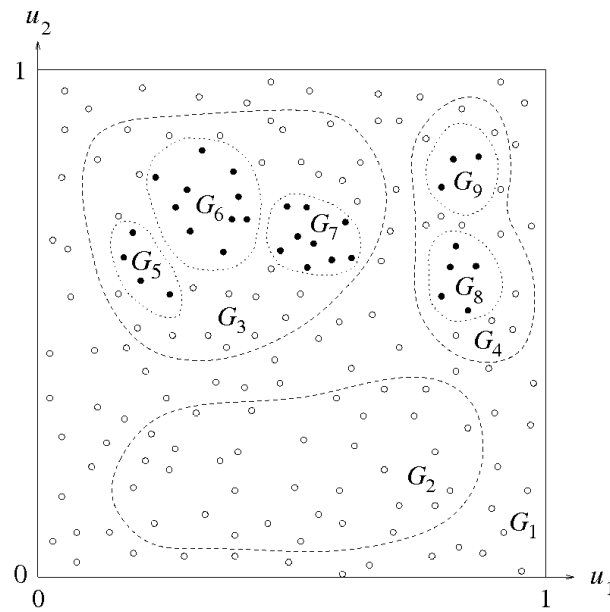- Finally, each active group is promoted to a 'mode', resulting in a set of L (say) such modes $\{M_l\}$.
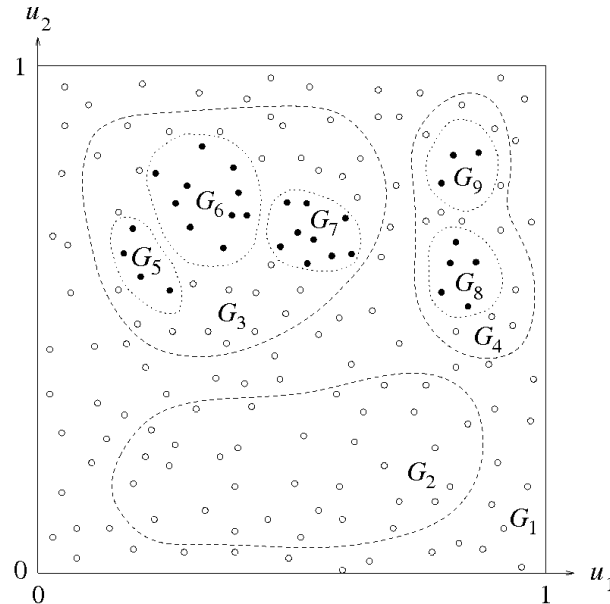
- Suppose lth mode M_l contains the points $\{\boldsymbol{u}_j\}$ $(j = 1, n_l)$

- In simplest approach, local evidence of mode is

$$\mathscr{Z}_l = \sum_{j=1}^{n_l} \mathcal{L}_j w_j$$

where $w_j = X_M/N$ for each active point in M_l and $w_j = \frac{1}{2}(X_{i-1} - X_{i+1})$ for each inactive point (i is NS iteration when inactive point was discarded).

- Similarly, posterior inferences resulting from lth mode obtained by weighting each point in M_l by $p_j = \mathcal{L}_j w_j / Z_l$.

- But... local evidence underestimated for modes lying close together – only identified as separate regions at high likelihood values

- Overcome problem by also making use of points in the inactive groups at end of NS process

- For each mode M_l, expression local evidence now reads

$$\mathcal{Z}_l = \sum_{j=1}^{n_l} \mathcal{L}_j w_j + \sum_g \mathcal{L}_g w_g \alpha_g^{(l)},$$

where sum over g includes all points in inactive groups, $w_g = \frac{1}{2}(X_{i-1} - X_{i+1})$ as above, and additional factors $\alpha_g^{(l)}$ are calculated as set out below.

- Similarly, posterior inferences from lth mode obtained by weighting each point in M_l by $p_j = \mathcal{L}_j w_j / Z_l$ and each point in inactive groups by $p_g = \mathcal{L}_g w_g \alpha_g^{(l)} / Z_l$

- Factors $\alpha_g^{(l)}$ most easily determined by essentially reversing the mode identification process

- Each mode M_l is simply a renamned active group G

- Identify inactive group G' that split to form G at the NS iteration i

- Assign all points in G' the factor

$$\alpha_g^{(l)} = \frac{n_G^{(A)}(i)}{n_{G'}^{(A)}(i)},$$

 where $n_G^{(A)}(i)$ is number of active points in G at NS iteration i; similar for $n_{G'}^{(A)}(i)$.

- Now, G' may itself have formed when an inactive group G'' split at an eariler NS iteration $i' < i$, in which case all points in G'' are assigned the factor

$$\alpha_g^{(l)} = \frac{n_G^{(A)}(i) \, n_{G'}^{(A)}(i')}{n_{G'}^{(A)}(i) \, n_{G''}^{(A)}(i')}.$$

- Process is continued until the recursion terminates

- Finally, all points in inactive groups not already assigned have $\alpha_g^{(l)} = 0$.

- Easy to show $\sum_{l=1}^{L} \mathcal{Z}_l = \mathcal{Z}$, the global evidence $\Rightarrow$ evidence exactly partitioned
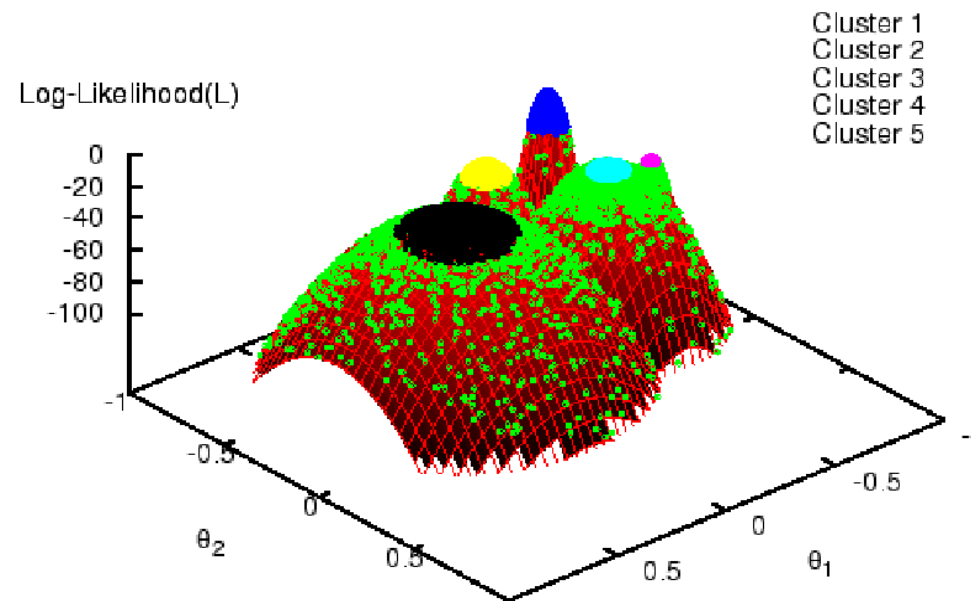
- Note: can instead use mixture model to assign factors
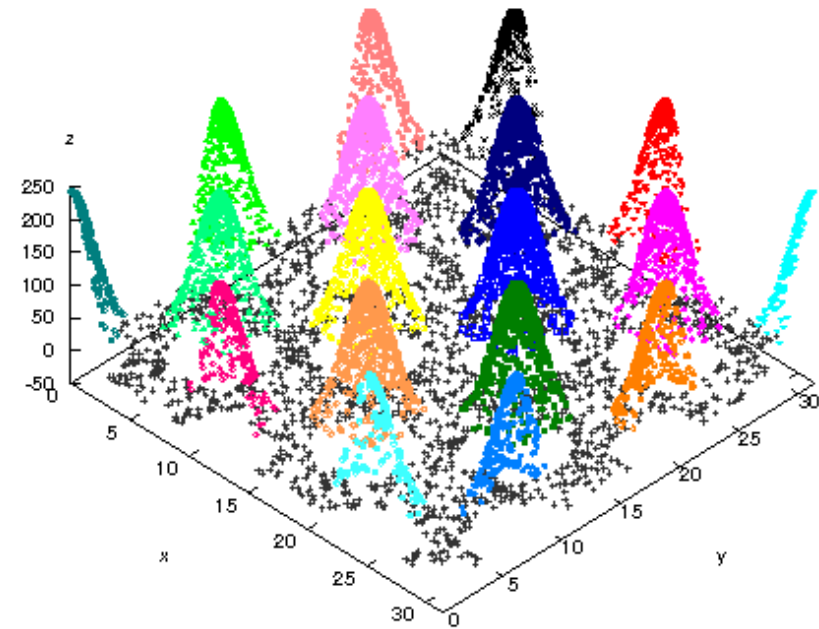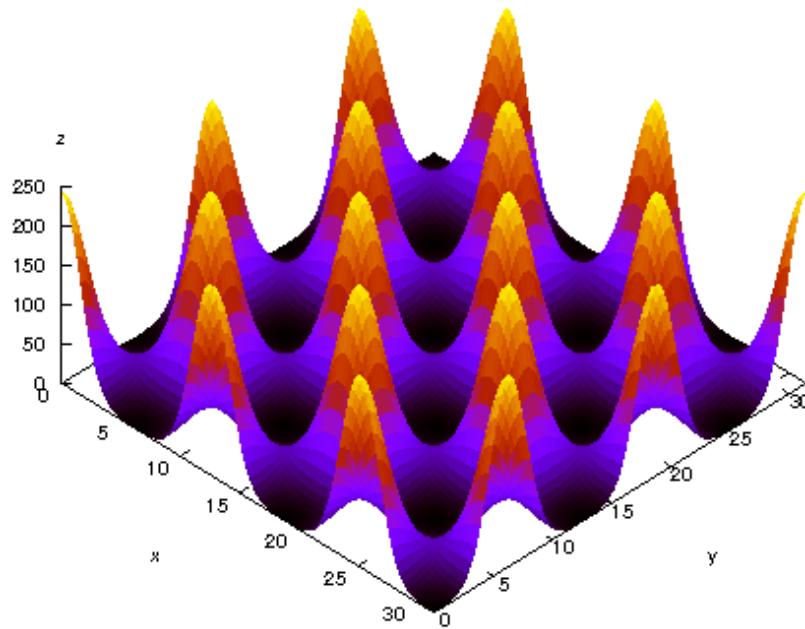
- Likelihood resembles egg-box and is given by

$$\mathcal{L}(\theta_1, \theta_2) = \exp\left[2 + \cos\left(\frac{\theta_1}{2}\right)\cos\left(\frac{\theta_2}{2}\right)\right]^5,$$

and prior is $\mathcal{U}(0, 10\pi)$ for both θ_1 and θ_2.

- Use 2000 active points $\Rightarrow \sim 30,000$ likelihood evaluations to obtain $\log \mathcal{Z} = 235.86 \pm 0.06$ (analytical $\log \mathcal{Z} = 235.88$)

- Likelihood = five 2-D Gaussians of varying widths and amplitudes; prior = uniform

- Analytic evidence integral $\log E = -5.27$

- MULTINEST: $\log E = -5.33 \pm 0.11$, $N_{\text{like}} \approx 10^4$

- Thermodynamic integration (+ error): $\log E = -5.24 \pm 0.12$, $N_{\text{like}} \approx 4 \times 10^6$

- Typical of real applications (see later): $\sim 500 \times$ efficiency of standard MCMC
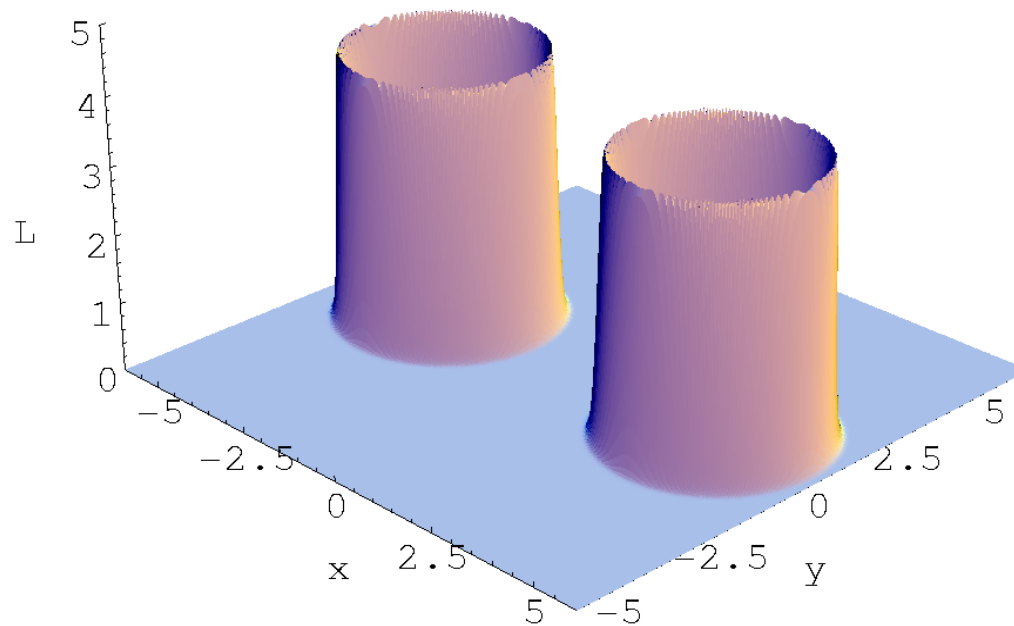
- Likelihood defined as

$$L(\boldsymbol{x}) = \text{circ}(\boldsymbol{x}; \boldsymbol{c}_1, r_1, w_1) + \text{circ}(\boldsymbol{x}; \boldsymbol{c}_2, r_2, w_2),$$

where

$$\text{circ}(\boldsymbol{x}; \boldsymbol{c}, r, w) = \frac{1}{\sqrt{2\pi w^2}} \exp\left[-\frac{(|\boldsymbol{x} - \boldsymbol{c}| - r)^2}{2w^2}\right].$$

and assuming a uniform prior

- MULTINEST results:
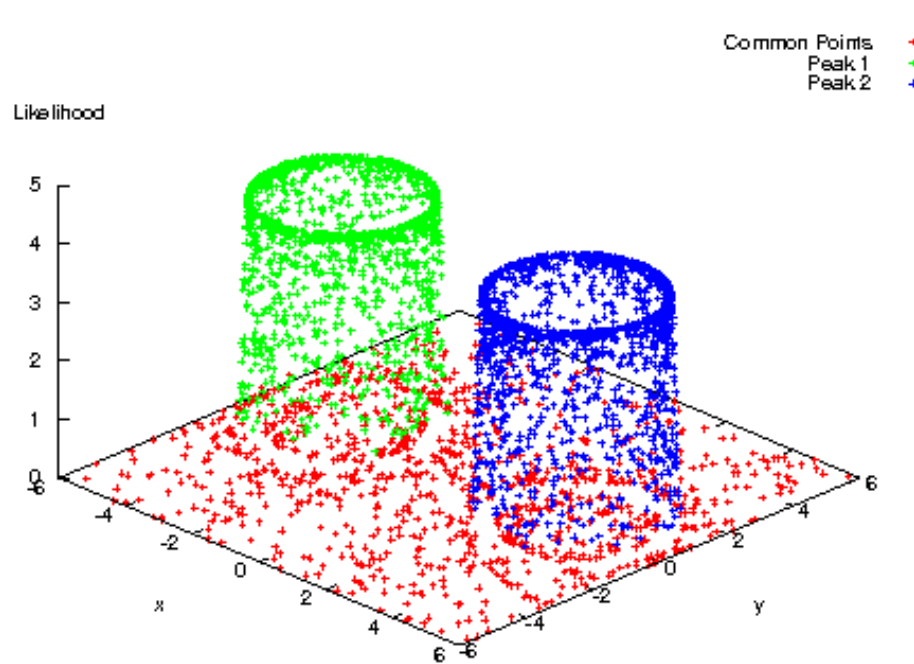
D	MULTINEST	
	N_{like}	Efficiency
2	$7,370$	70.77%
5	$17,967$	51.02%
10	$52,901$	34.28%
20	$255,092$	15.49%
30	$753,789$	8.39%

	Analytical		MULTINEST		
D	$\log(\mathcal{Z})$	local $\log(\mathcal{Z})$	$\log(\mathcal{Z})$	local $\log(\mathcal{Z}_1)$	local $\log(\mathcal{Z}_2)$
2	-1.75	-2.44	-1.72 ± 0.05	-2.28 ± 0.08	-2.56 ± 0.08
5	-5.67	-6.36	-5.75 ± 0.08	-6.34 ± 0.10	-6.57 ± 0.11
10	-14.59	-15.28	-14.69 ± 0.12	-15.41 ± 0.15	-15.36 ± 0.15
20	-36.09	-36.78	-35.93 ± 0.19	-37.13 ± 0.23	-36.28 ± 0.22
30	-60.13	-60.82	-59.94 ± 0.24	-60.70 ± 0.30	-60.57 ± 0.32

- Bank sampler (MCMC): $N_{\text{like}} \sim 10^6$ in $D = 2$ for parameter estimation alone
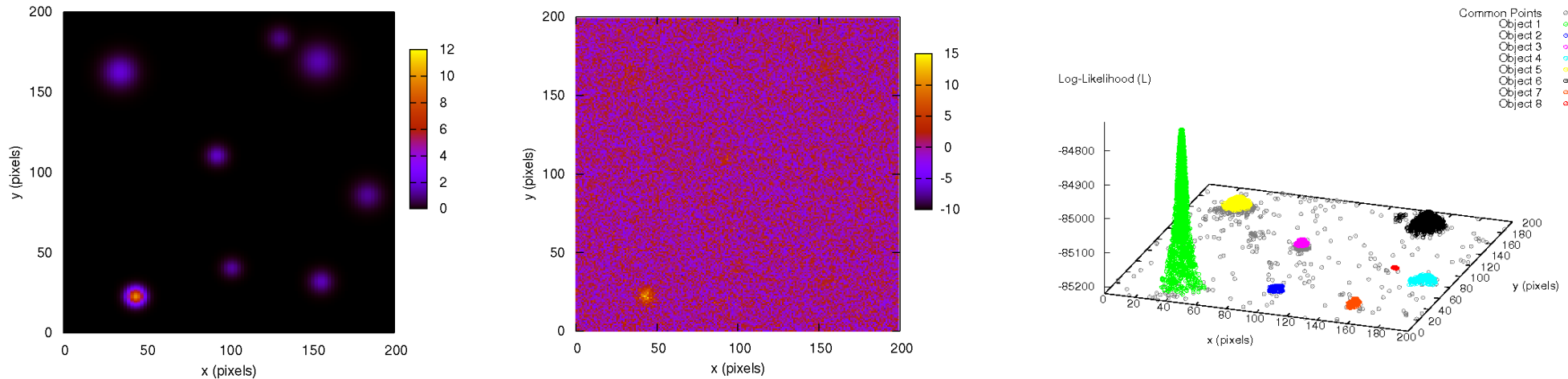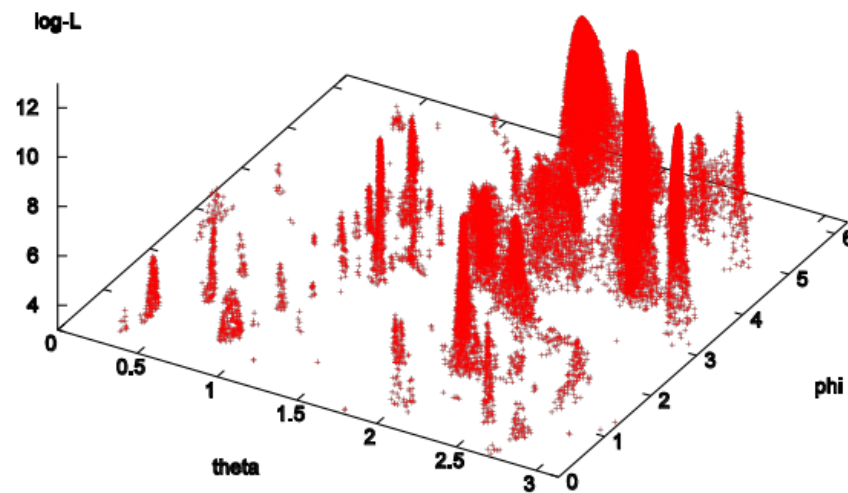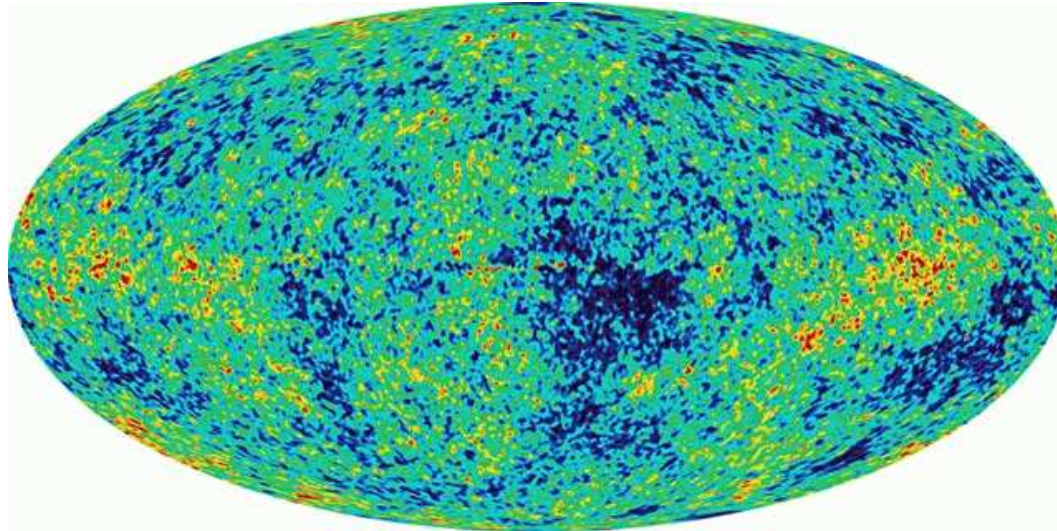
- Toy model: Gaussian objects in noise (Feroz & MPH, arXiv:0704.3704)

- Multinest: $N_{\text{like}} \sim 10^4$, run time ~ 2 CPU mins – identified all objects correctly

- BayeSys (MCMC + thermo. int.): $N_{\text{like}} \sim 5 \times 10^6$, run time ~ 16 CPU hrs
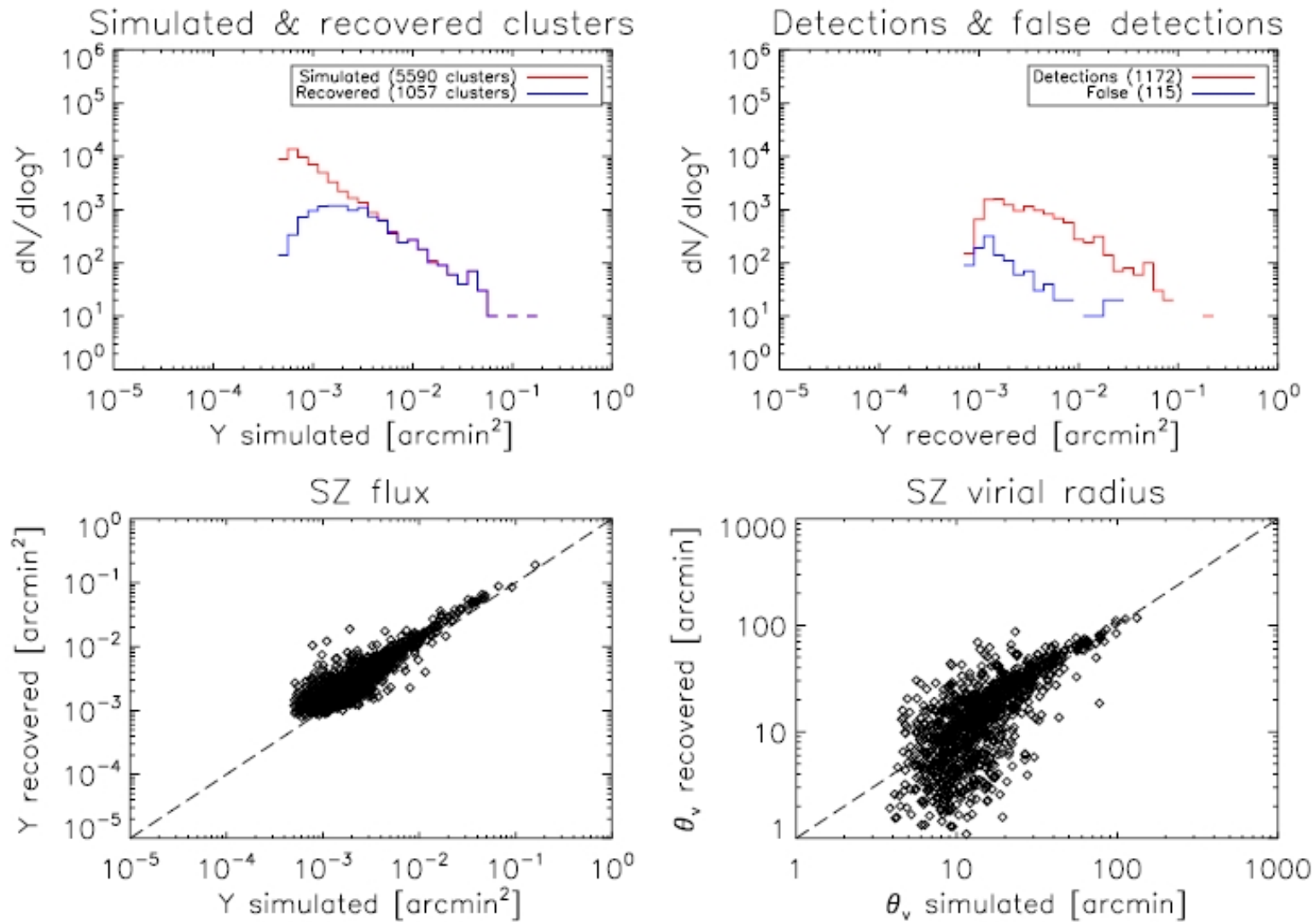 Required several object subtraction iterations to identify all objects
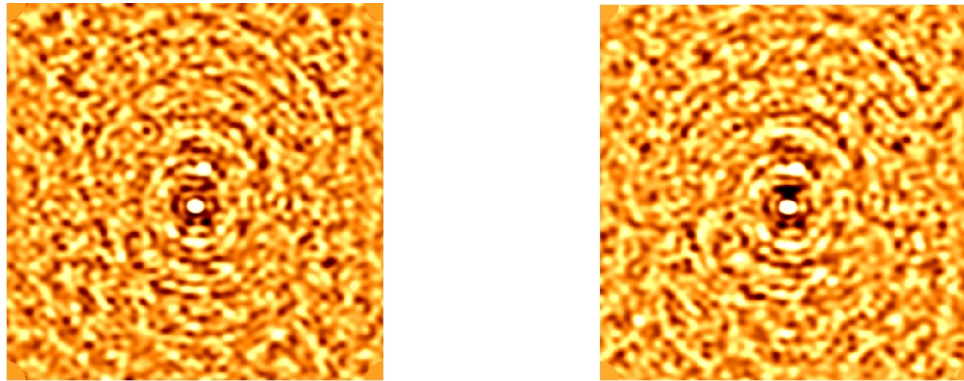
• Textures in CMB data (in preparation)

- SZ clusters in all-sky Planck data (SZ Challenge 2)

- 50×10^6 pixels, ~ 1000 recovered clusters, ~ 3 CPU hrs

- Cluster (and point sources) in interferometric SZ data (Feroz et al., arXiv:0811.1199)

- Simulations: A (left) without cluster and B (right) with cluster (β-model), including CMB, 3 point sources, confusion noise, instrumental noise

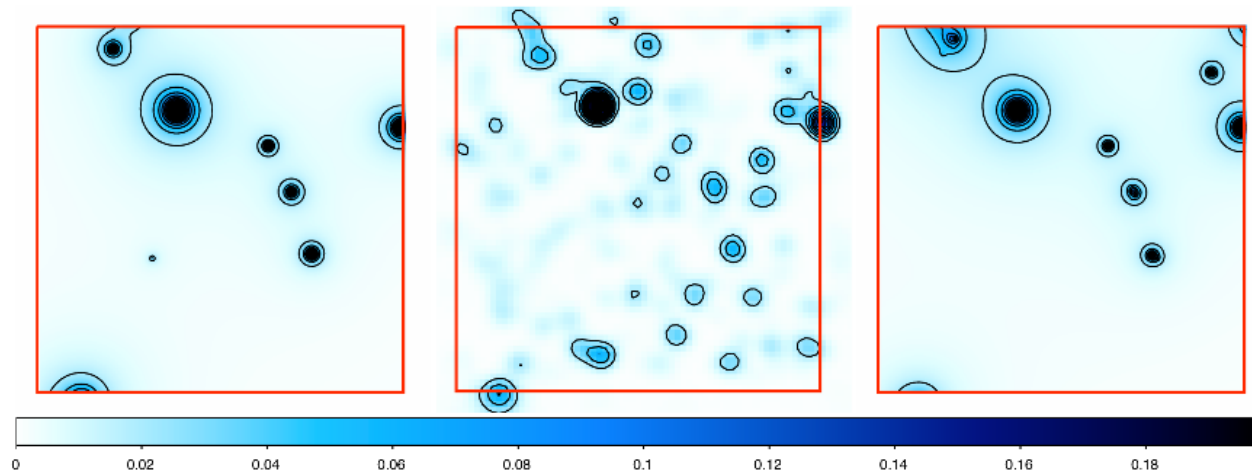

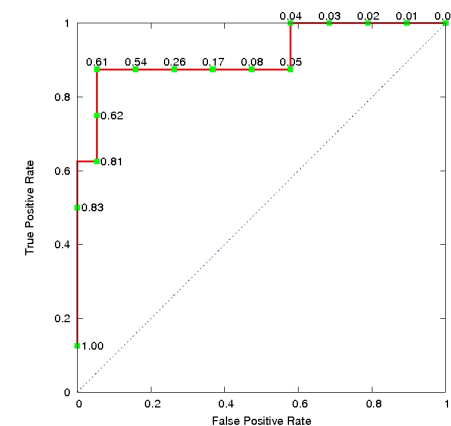- A simulation $R = 0.35 \pm 0.05$; B simulation $R \sim 10^{33}$. Parameter constraints:

62
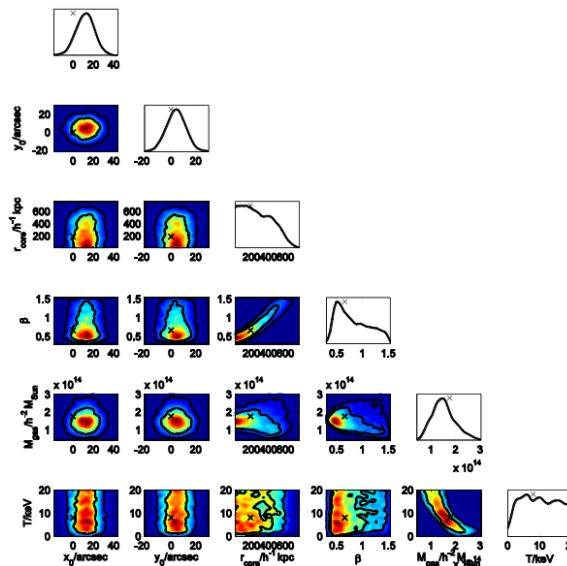
- Clusters in weak lensing surveys (Feroz, Marshall, MPH, arXiv:0810.0781)

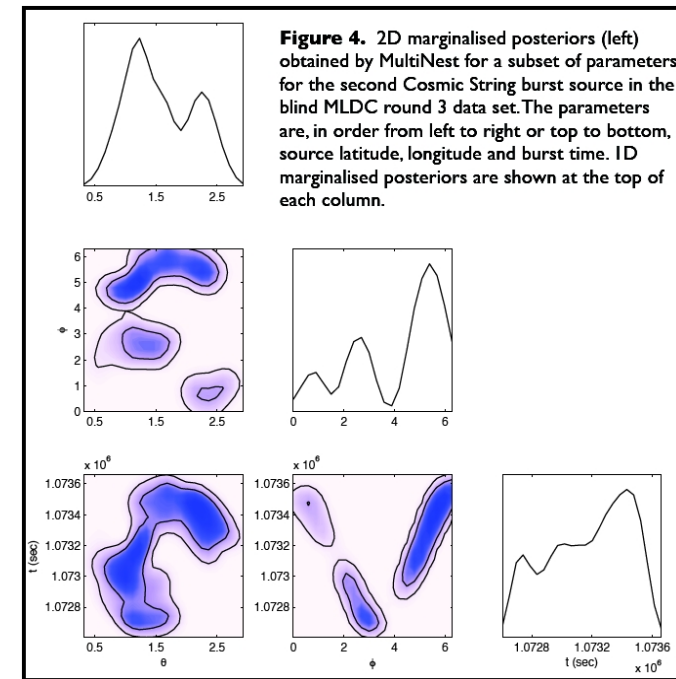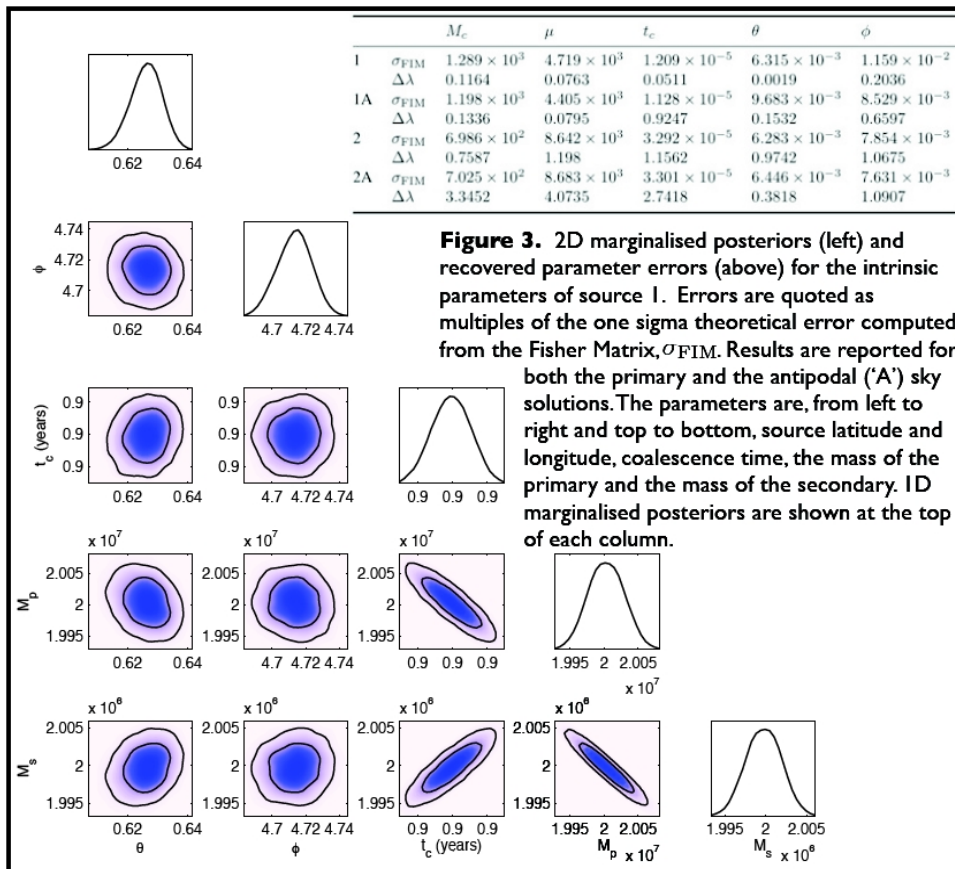- 0.5×0.5 deg^2 simulation (ΛCDM + Press–Schechter), 100 gal arcmin^{-2}, $\sigma = 0.3$

- Probability ith mode is true positive $p_i = R_i/(1 + R_i) \Rightarrow \widehat{n}_{FP} = \sum_{\substack{i=1 \\ p_i > p_{th}}}^{N} (1 - p_i)$

63
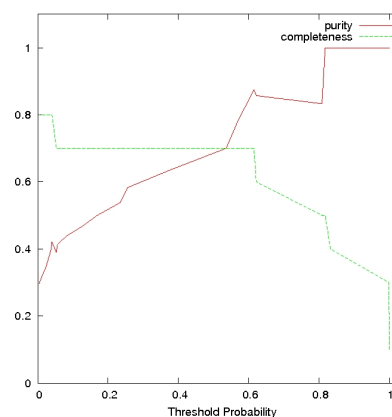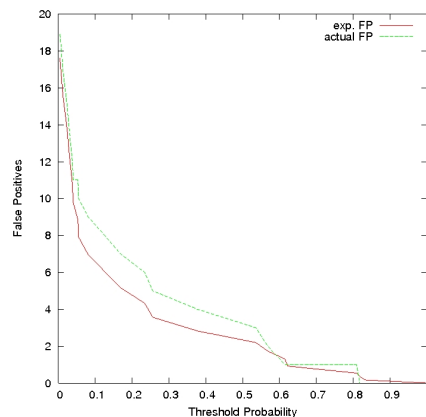
- Simulated LISA data containing two signals from non-spinning SMBH mergers. Each source has antipodal degeneracy $\Rightarrow$ at least 4 modes in posterior

- All identified and well characterized in ~ 2 CPU hrs (Feroz et al., arXiv:0904.1544)

Figure 3. 2D marginalised posteriors (left) and recovered parameter errors (above) for the intrinsic parameters of source 1. Errors are quoted as multiples of the one sigma theoretical error computed from the Fisher Matrix, σ_{FIM}. Results are reported for both the primary and the antipodal ('A') sky solutions. The parameters are, from left to right and top to bottom, source latitude and longitude, coalescence time, the mass of the primary and the mass of the secondary. 1D marginalised posteriors are shown at the top of each column.

Figure 4. 2D marginalised posteriors (left) obtained by MultiNest for a subset of parameters for the second Cosmic String burst source in the blind MLDC round 3 data set. The parameters are, in order from left to right or top to bottom, source latitude, longitude and burst time. 1D marginalised posteriors are shown at the top of each column.

- Also applied successfully in Mock LISA Data Challenge Round 3 to simulations of 5 spinning BH binary inspirals and 3 cosmic strings (Feroz et al. arXiv:0911.0288)