Introduction to Statistical Inference

Neil J. Cornish

eXtreme Gravity Institute

MONTANA STATE UNIVERSITY

Resources

Papers/Reviews

Romano & Cornish “Detection methods for stochastic gravitational-wave backgrounds: a unified treatment”, Living Reviews in Relativity (2017) 20: 2. doi:10.1007/s41114-017-0004-1

Cornish “Black Hole Merging and Gravitational Waves”, Black Hole Formation and Growth, Saas-Fee Advanced Course 48, (2019)
https://www.dropbox.com/s/l8nusg5fd5x3ak1/2019_Book_BlackHoleFormationAndGrowth.pdf?dl=0

Books

Maggiore, “Gravitational Waves: Volume 1: Theory and Experiments”

Creighton & Anderson “Gravitational-Wave Physics and Astronomy: An Introduction to Theory, Experiment and Data Analysis”

Statistical Inference

Bayesian Inference

Reverend Thomas Bayes

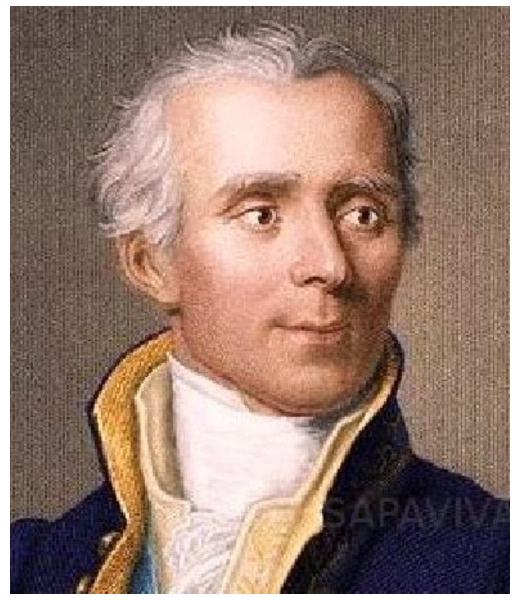
Frequentists Statistics

Sir Ronald Fisher

“there is a valid defense for using non-Bayesian methods, namely incompetence.” - Skilling

“The theory of probabilities is nothing but common sense reduced to calculus”

Pierre Simon de Laplace



Frequentist: $p(A)$ - long run relative frequency with which A occurs in identical trials

Bayesian: $p(A \mid H)$ - relative measure of proposition A given the truth of hypothesis H

Frequentist: $p(d \mid h)$ likelihood of observing the data d given the model h

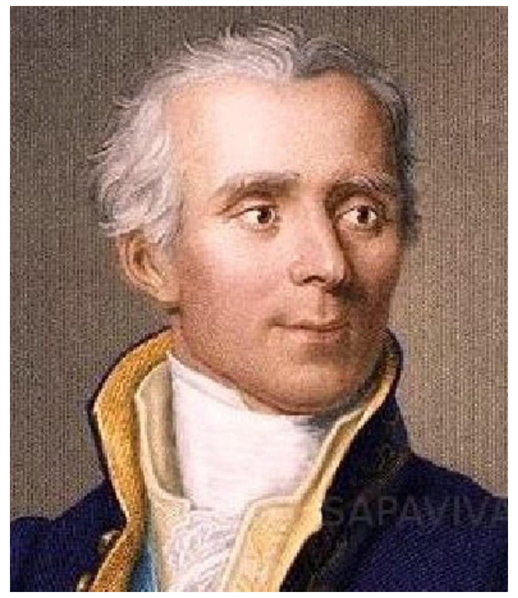
Bayesian: $p(h \mid d, H)$ likelihood of observing the model h given data d under hypothesis H

In the frequentist view, the signal is fixed and the data are considered random. In the Bayesian view the data is what you record (nothing random about it), and the signal is to be inferred in a probabilistic sense.

Frequentist analyses give sensible answers when they closely approximate the Bayesian approach

Data Analysis 101 - Laplace & Gauss

Pierre Simon de Laplace

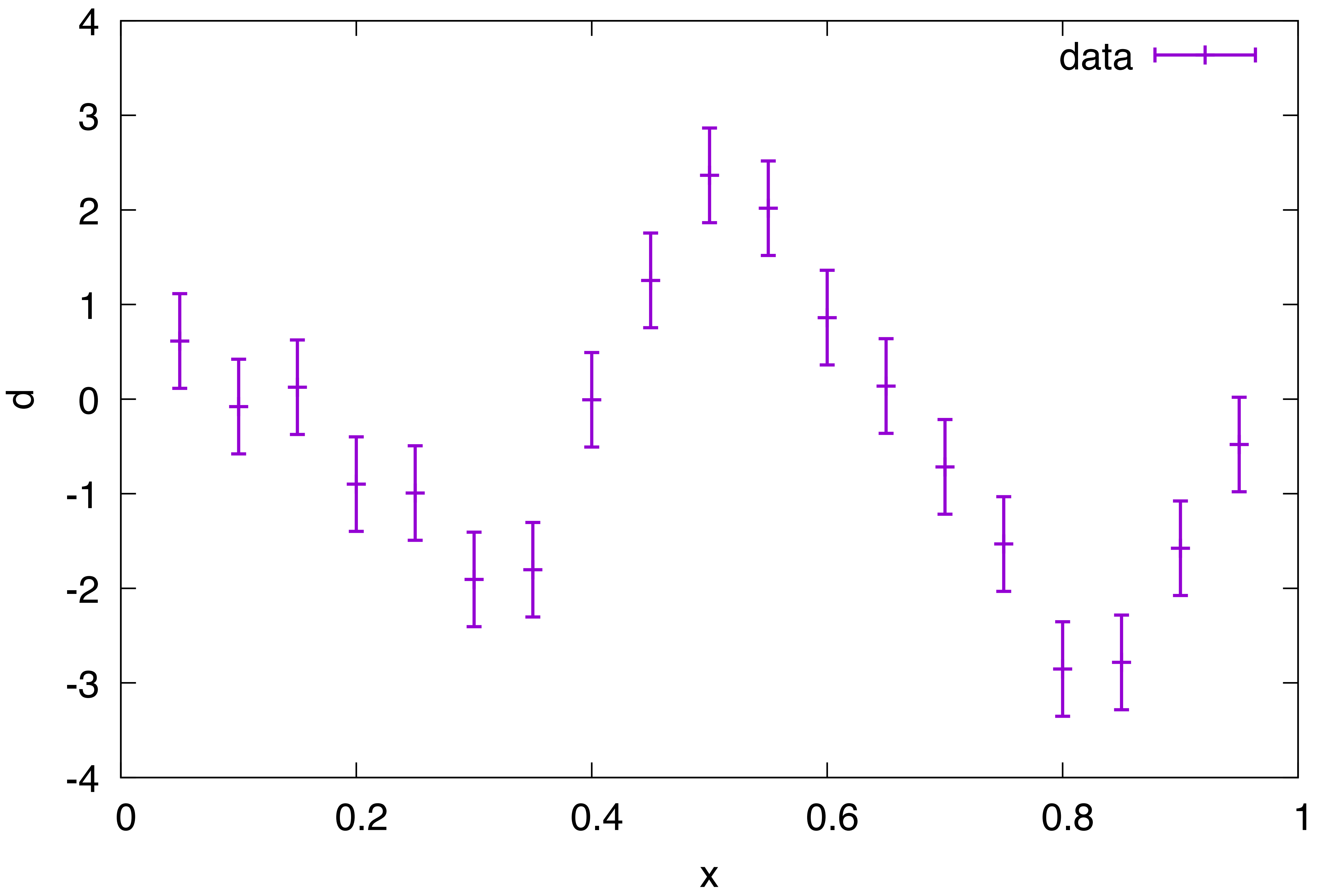


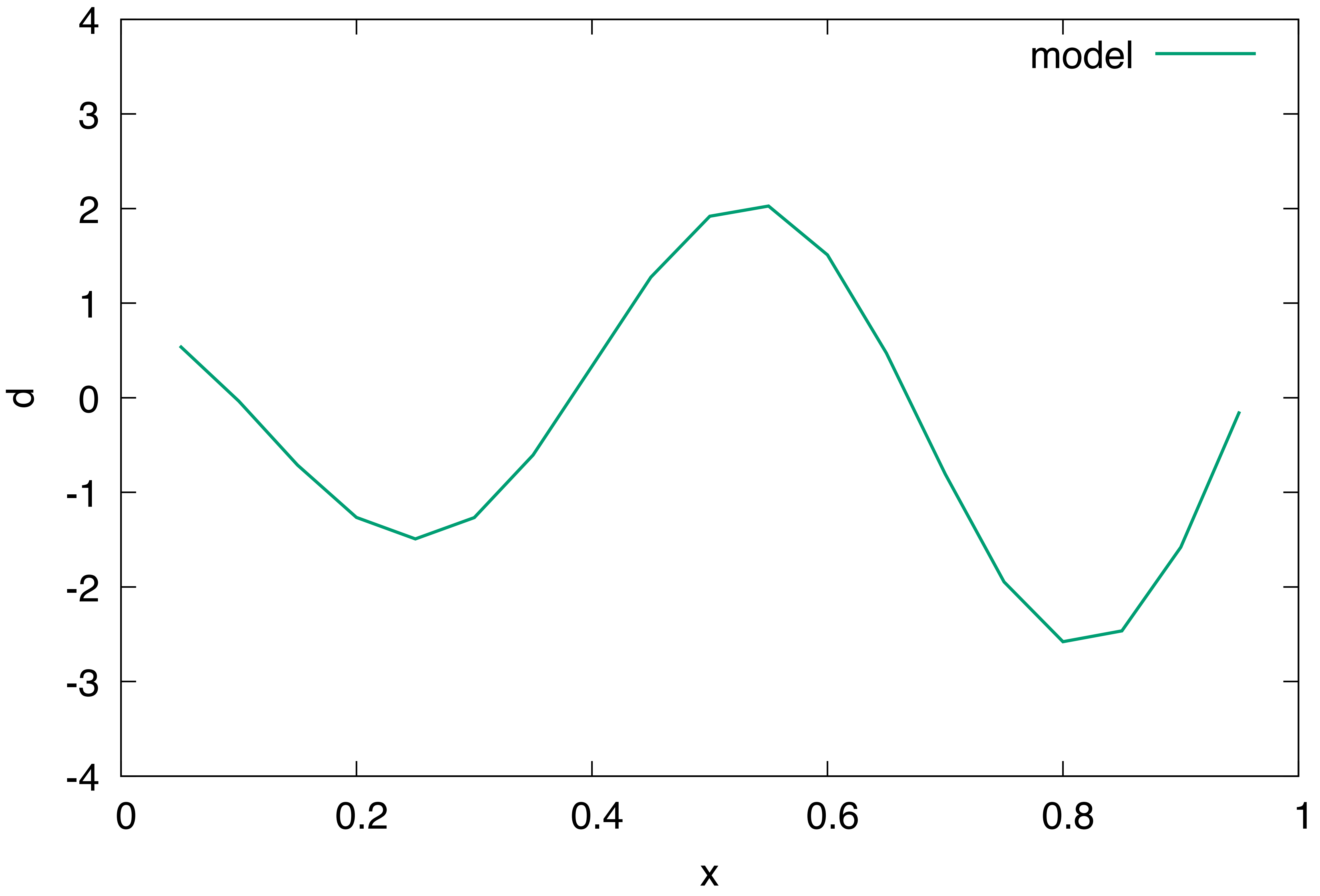
Memoir on the Probability of Causes of Events (1774)
Analytical Theory of Probability (1812)

Carl Friedrich Gauss

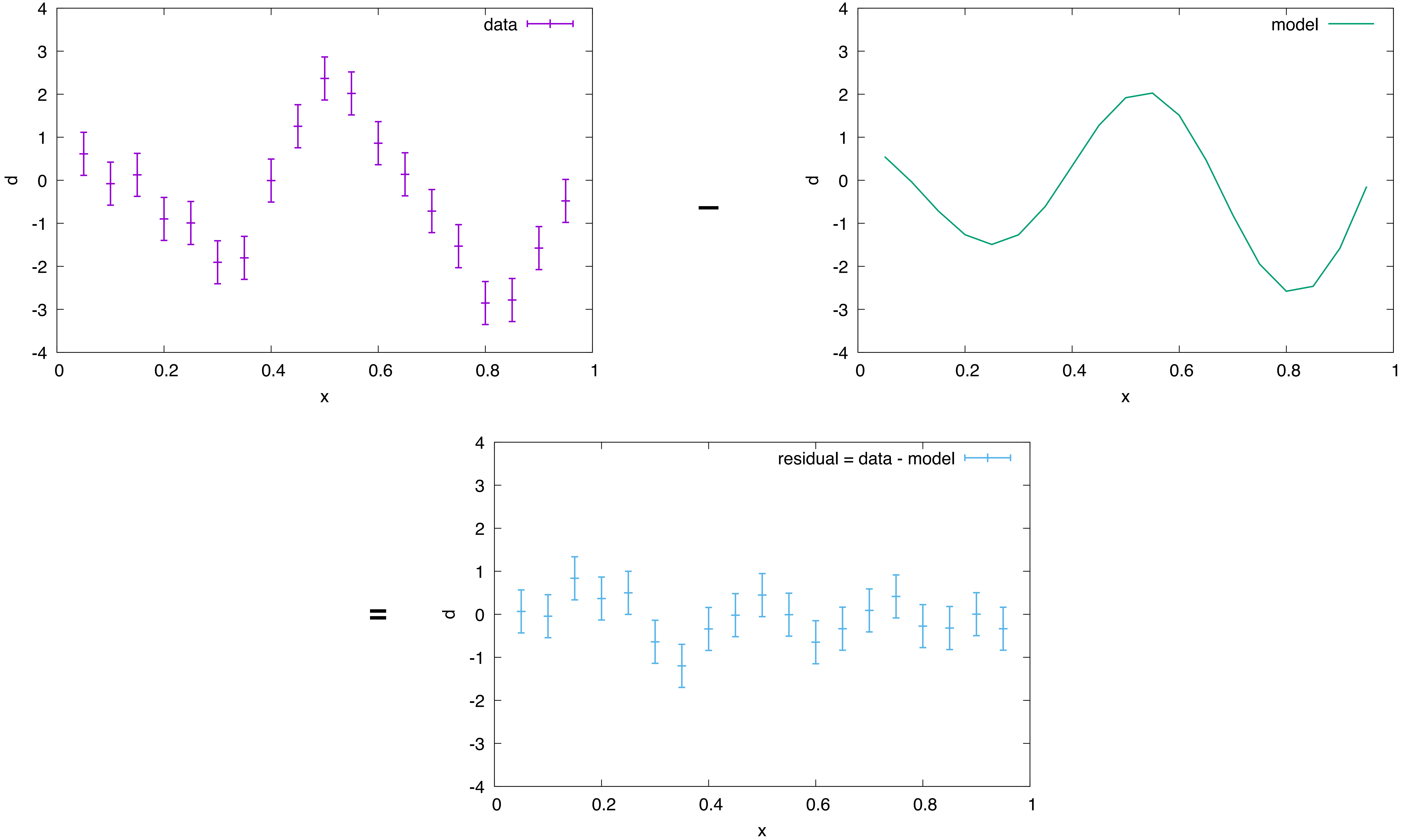
Z. Astronom. Verwandte Wiss. 1 185, (1816)

Laplace developed Bayesian Inference. Gauss developed maximum likelihood estimation.
Gauss introduced the normal distribution, Laplace explained its ubiquity (CLT).





Data Analysis 101



Data Analysis 101

For additive noise (linear response)

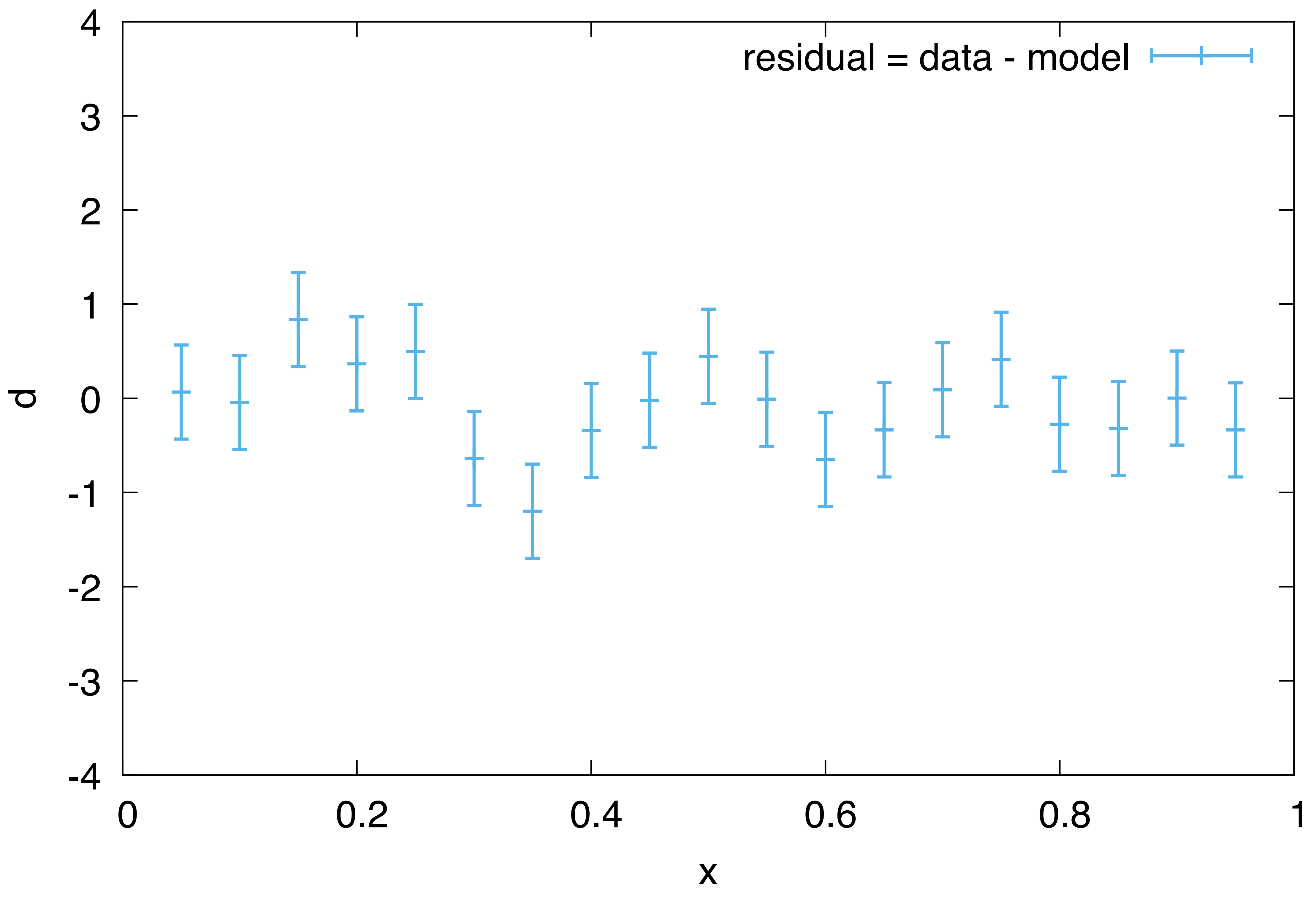
$$d = h + n \qquad \Rightarrow \qquad d - h = n$$

The residuals should follow the noise distribution

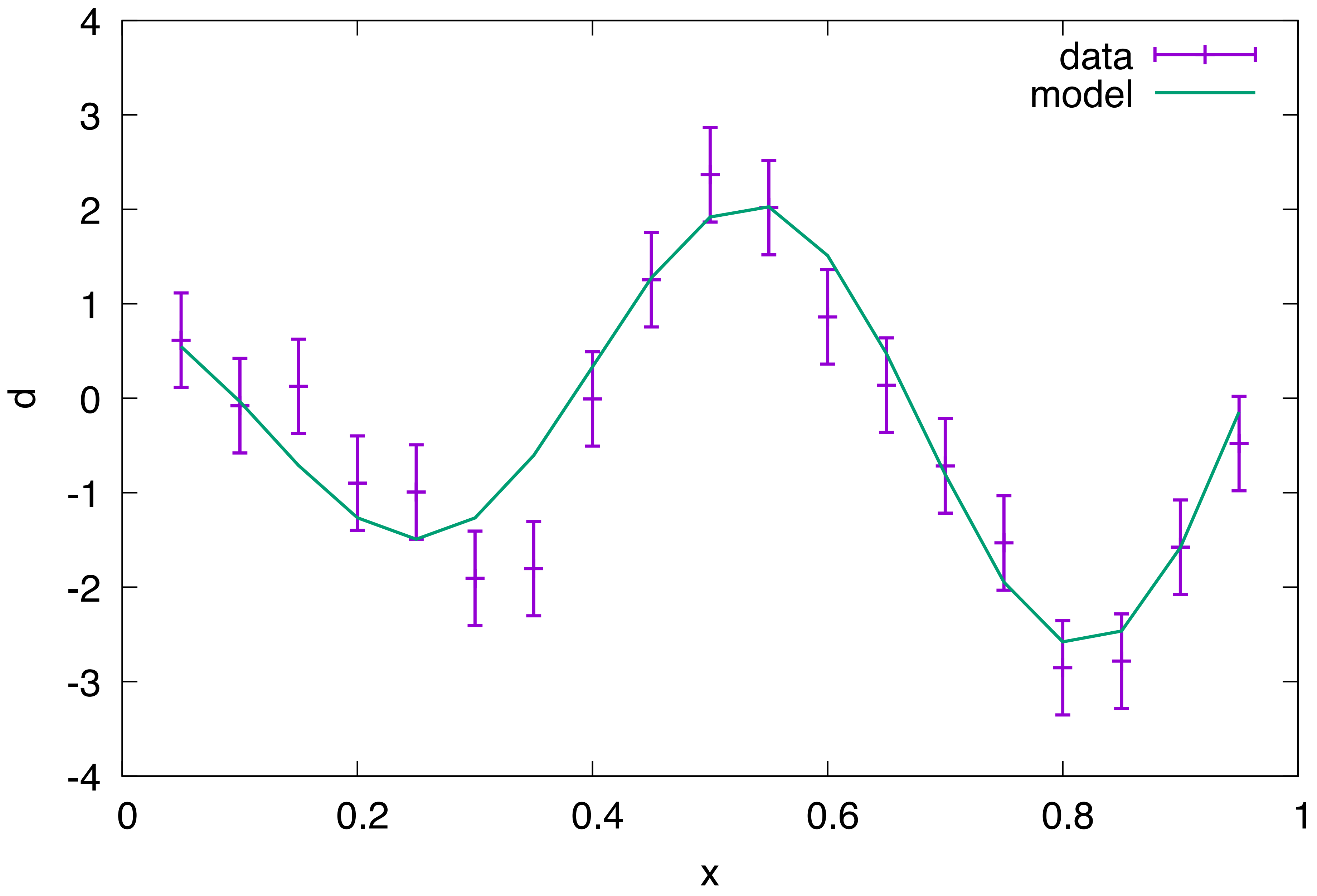
In this example the noise was uncorrelated between samples and draw from a Gaussian distribution with a fixed standard deviation ("stationary white noise")

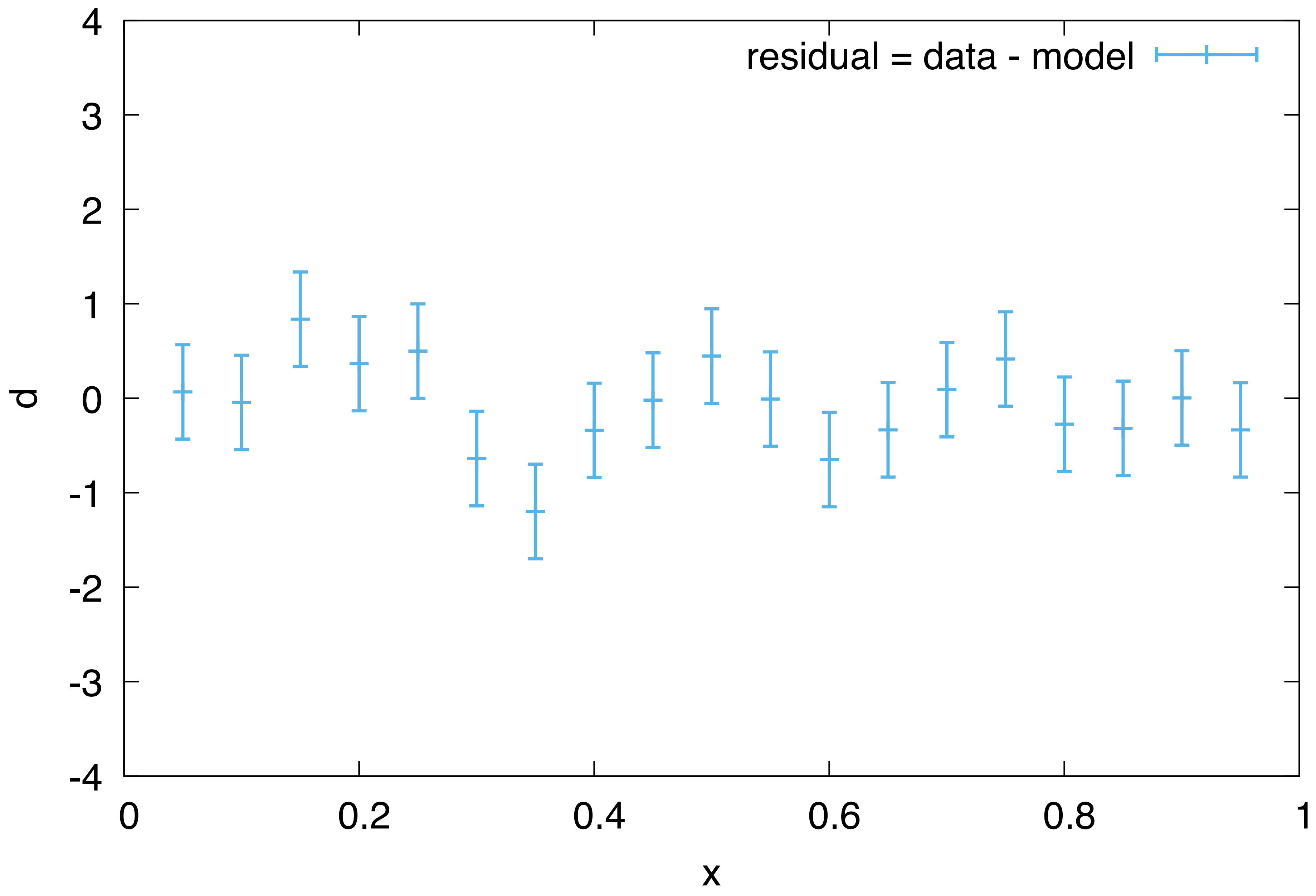
$$p(n_i) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{n_i^2}{2\sigma^2}}$$

$$p(n) = \prod_i p(n_i) = \frac{1}{(2\pi)^{N/2}\sigma^N} e^{-\sum_{i=1}^{N} \frac{n_i^2}{2\sigma^2}}$$



Data Analysis 101





The likelihood of observing the data d given the model h is simply

$$p(d - h) = \frac{1}{(2\pi)^{N/2}\sigma^N} e^{-\sum_{i=1}^{N} \frac{(d_i - h_i)^2}{2\sigma^2}}$$

Data Analysis 101

If the noise is correlated between data samples ("colored noise"), and/or if the amplitude of the noise changes from sample to sample ("heteroskedastic" aka "non-stationary"), then we need to generalize the Gaussian likelihood:

$$p(d-h) = \frac{1}{(\det(2\pi\mathbf{C}))^{N/2}}\, e^{-\frac{1}{2}(d_i - h_i)C^{-1}_{ij}(d_j - h_j)}$$

In the previous example the noise correlation matrix was promotional to the identity matrix: $C_{ij}^{\mathrm{SWN}} = \delta_{ij}\,\sigma^2$

The quantity in the exponent is the chi squared goodness of fit

$$\chi^2 = (d_i - h_i)C^{-1}_{ij}(d_j - h_j) \equiv (d - h \,|\, d - h)$$

Here I have introduced the noise weighted inner product

$$(a\,|\,b) = a_i\, C^{-1}_{ij}\, b_j$$

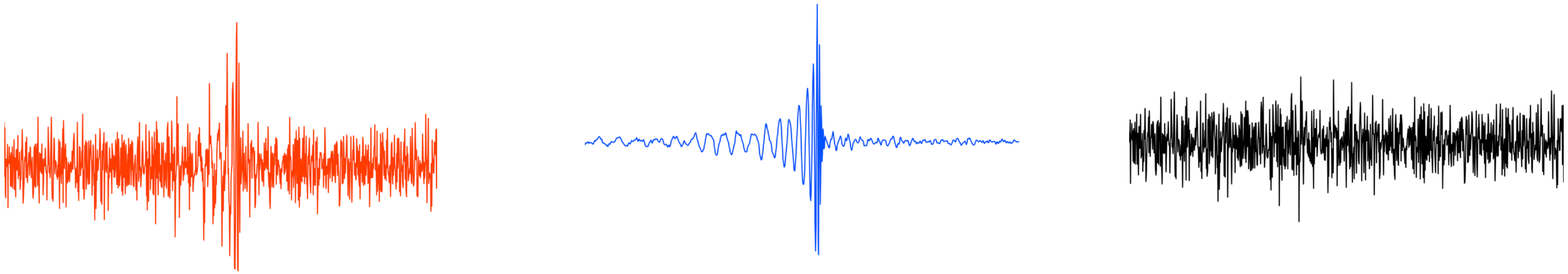
Data Analysis 101

Gravitational wave data comes in the form of a time series. Computing the noise-weighted inner product in the time domain is costly since the matrix is not diagonal. If the noise is *stationary,* i.e. has statistical properties that are unchain with time, then the noise correlation matrix is diagonal in the frequency domain. That is why most GW analysis is done in the frequency domain.

$$(a\,|\,b) = a_i\, C^{-1}_{ij}\, b_j = 2\sum_f \frac{\tilde{a}(f)\tilde{b}^*(f) + \tilde{a}^*(f)\tilde{b}(f)}{S_n(f)}$$

The factor of 2 is because we only sum over positive frequencies. The quantity $S_n(f)$ is the one-sided noise spectral density (PSD)

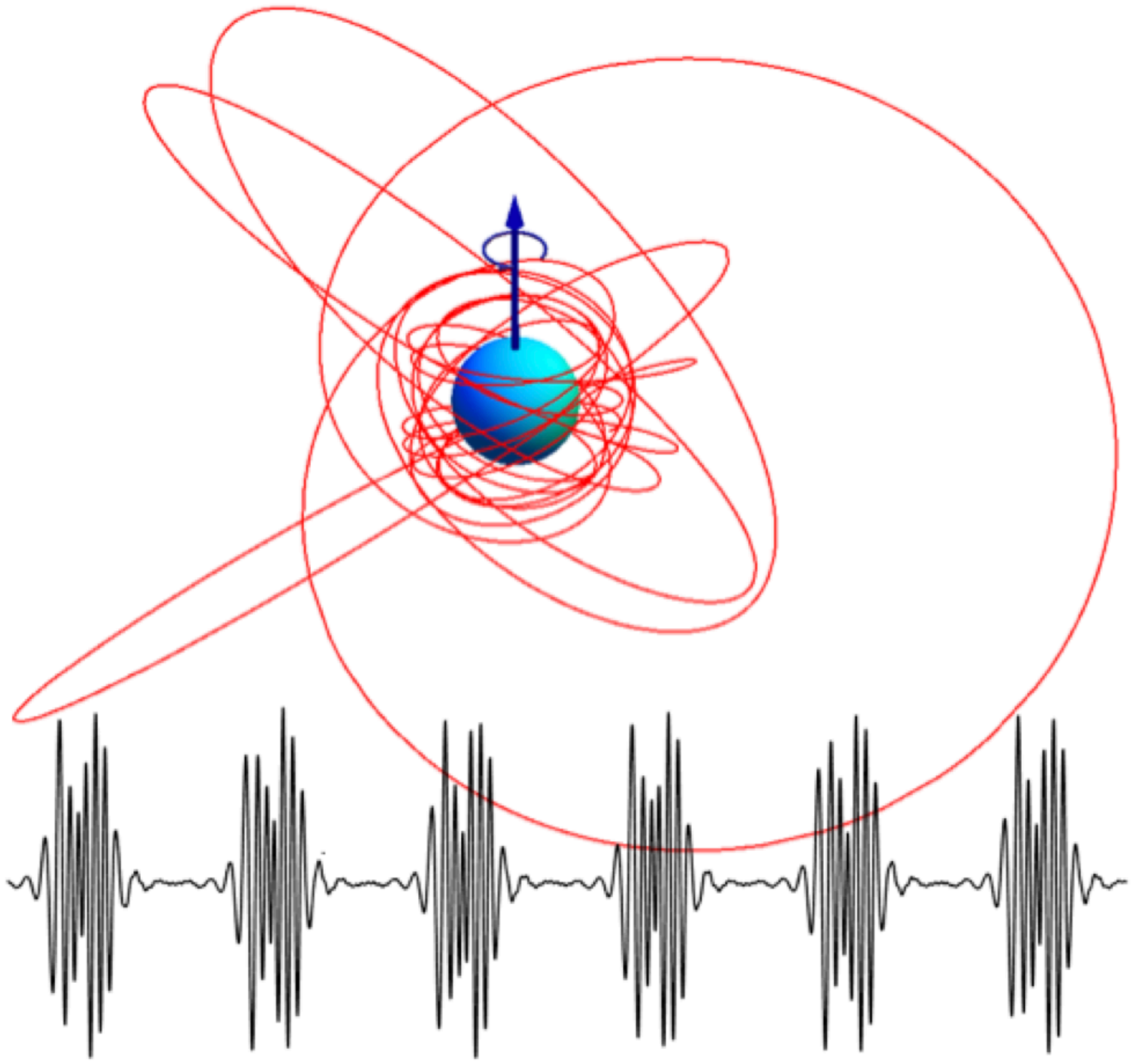
We often talk about “whitened” data or waveforms. This is simply $\tilde{a}^W(f) = \tilde{a}(f)/S_n(f)^{1/2}$. Can transform this back to the time domain:

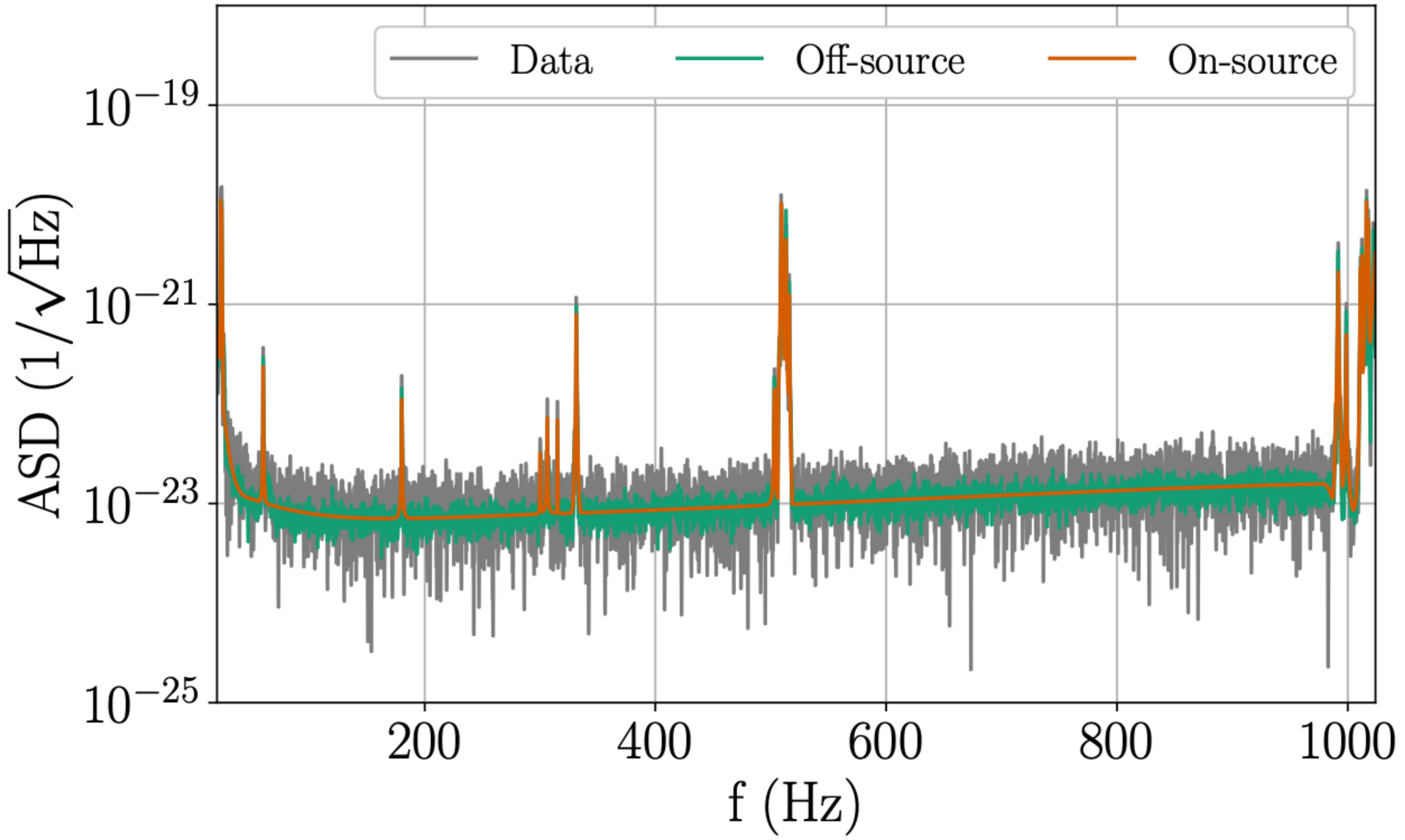


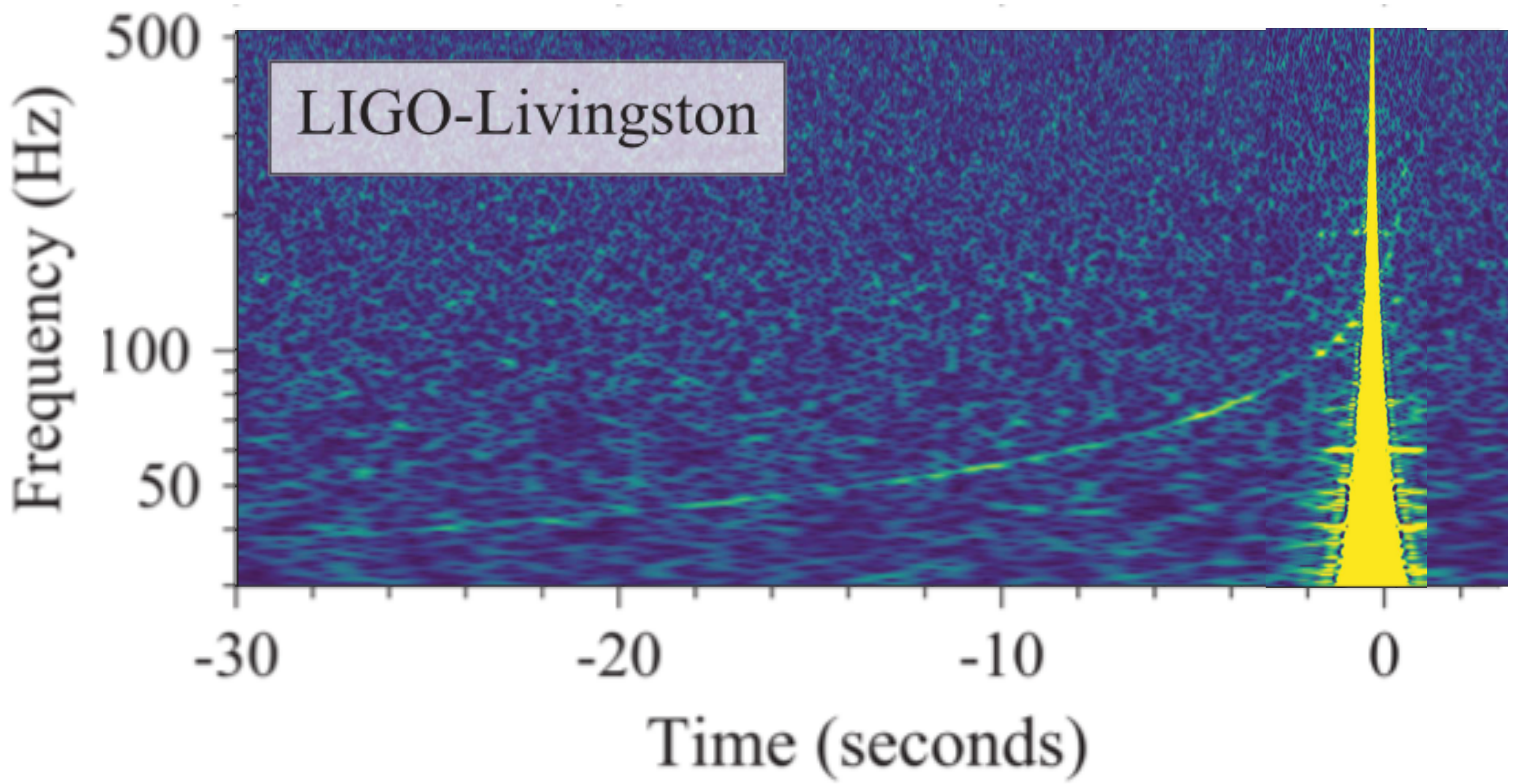
data - signal = noise

Challenges for Gravitational Wave Data Analysis

- Complicated waveforms, as many as 17 parameters
- Noise properties have to be estimated along with the signals
- Non-Gaussian noise transients have to be modeled/mitigated







The Bayesian Way

Likelihood
(noise model)

Prior
(signal model)

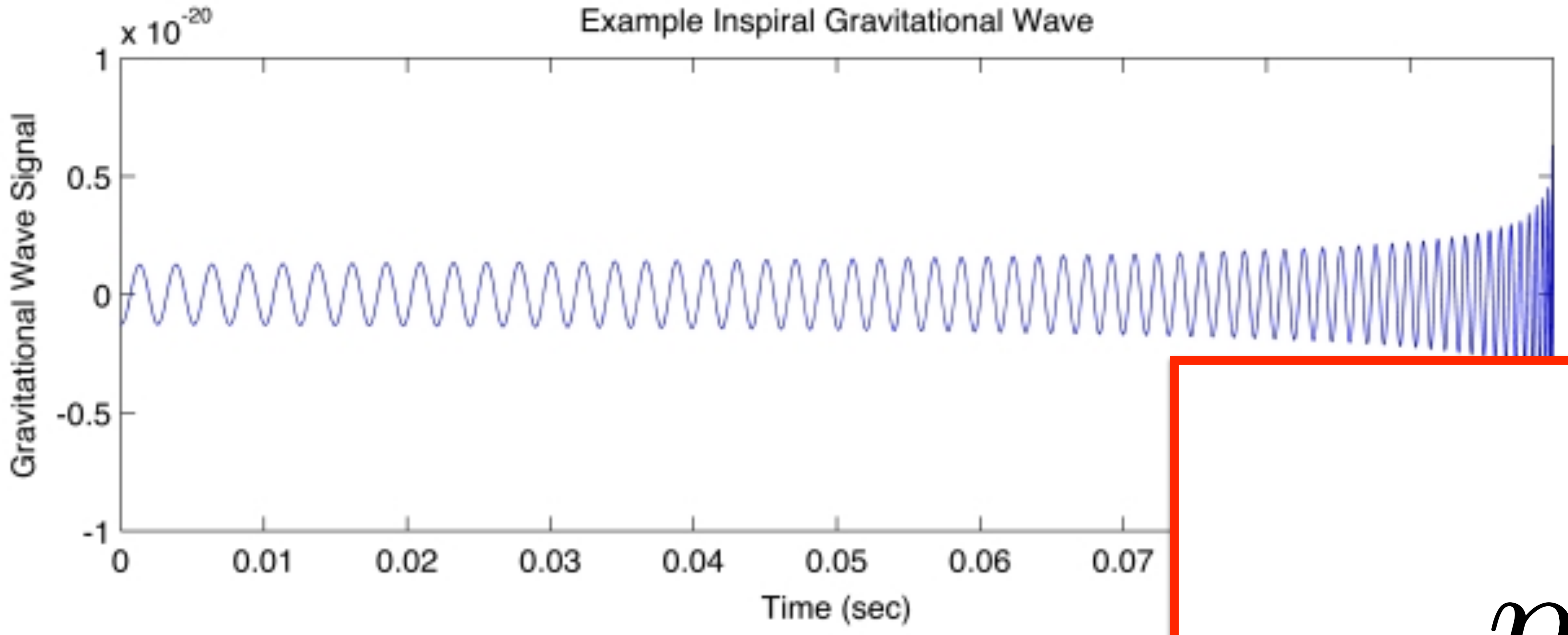
Posterior
probability for
waveform h

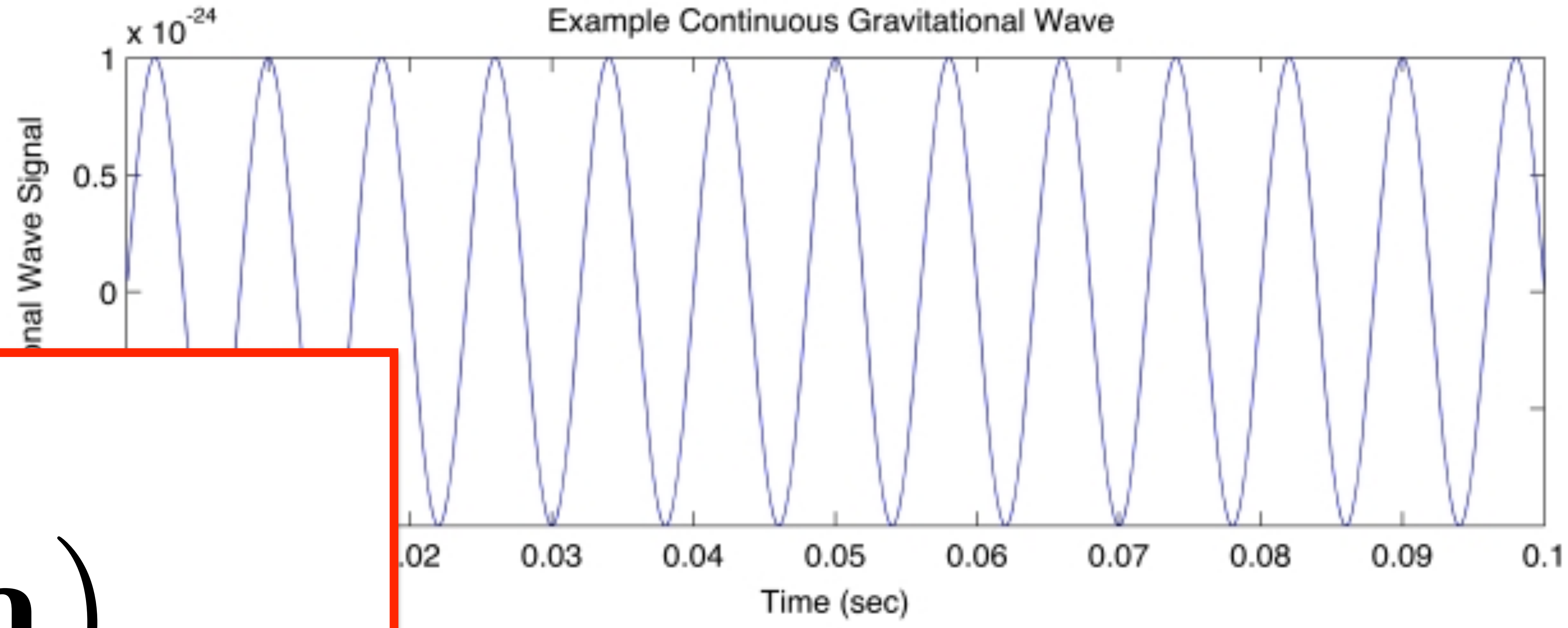
$$p(h|d) = \frac{p(d|h)p(h)}{p(d)}$$

Normalization - model evidence

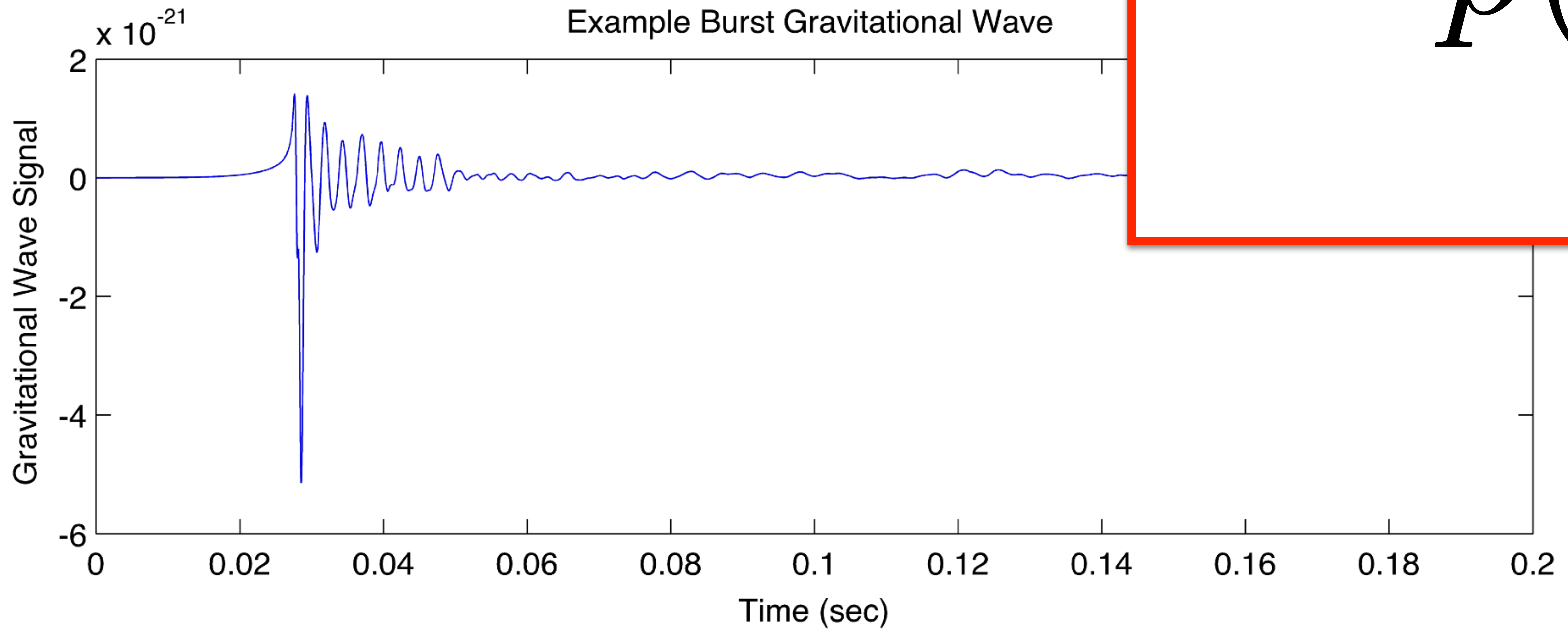
Gravitational wave signal types

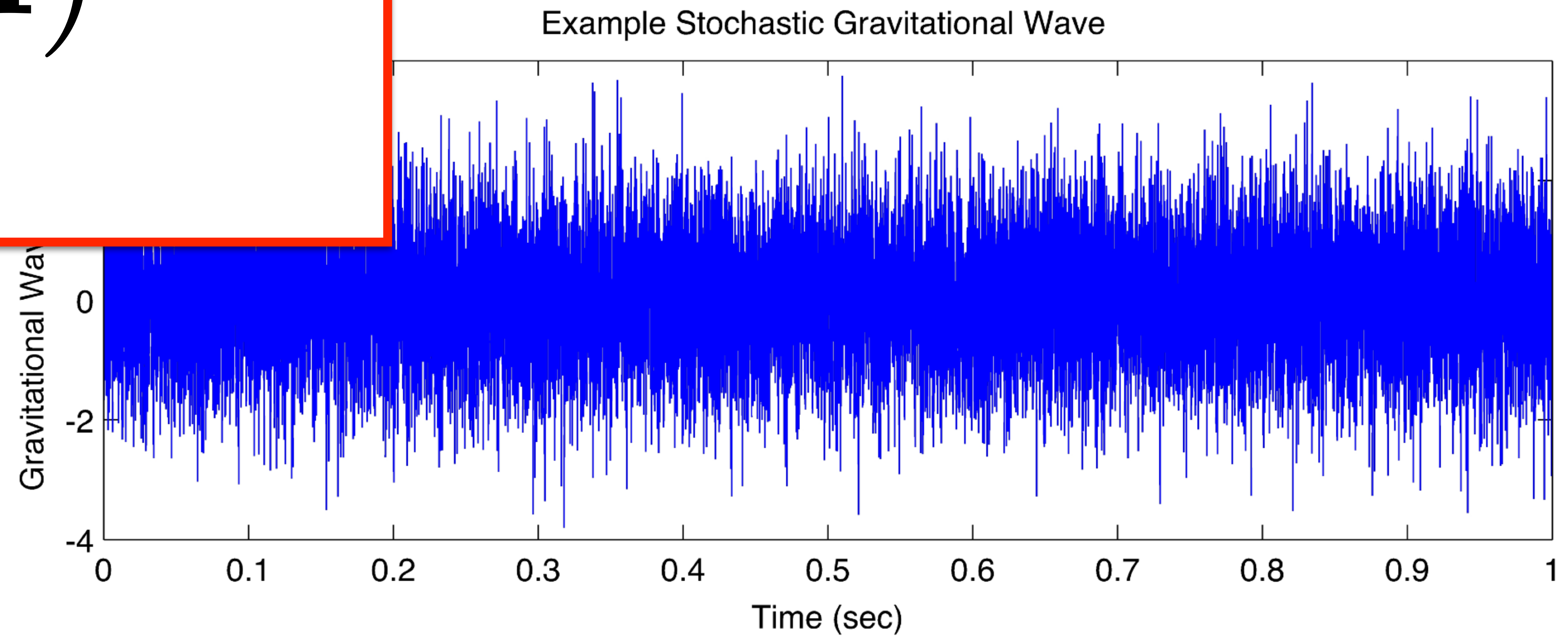
Well modeled - e.g. binary inspiral and merger





$$p(\mathbf{h})$$



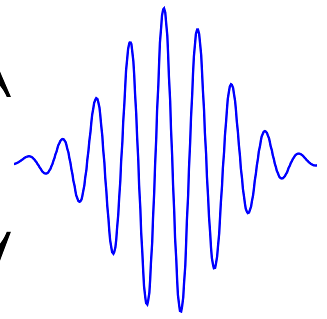
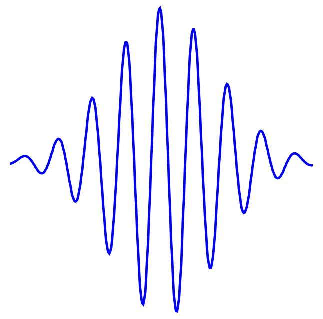


Poorly modeled - e.g. core collapse supernovae

Stochastic- e.g. phase transition in early universe

Gravitational wave signal models

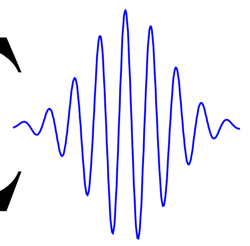
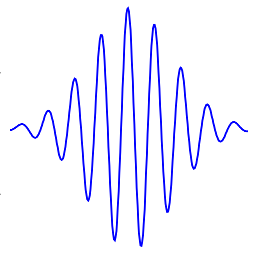
Template based $\quad p(\mathbf{h}) = \delta(\mathbf{h} - \mathbf{h}(\vec{\lambda})), \quad p(\vec{\lambda})$

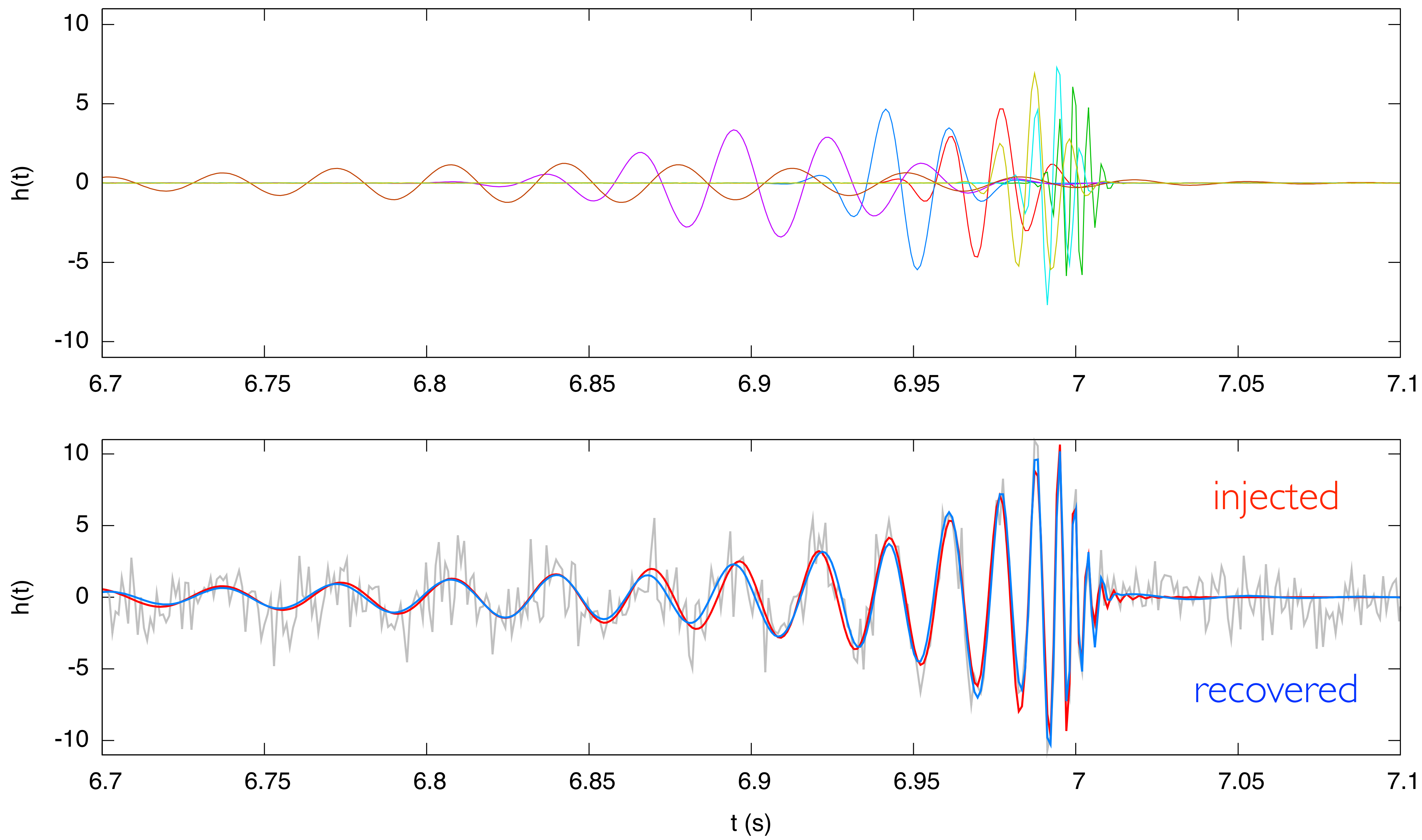
Burst signals $\quad p(\mathbf{h}) = \delta\Big(\mathbf{h} - \sum$  $\Big), \quad p\Big($  $\Big)$

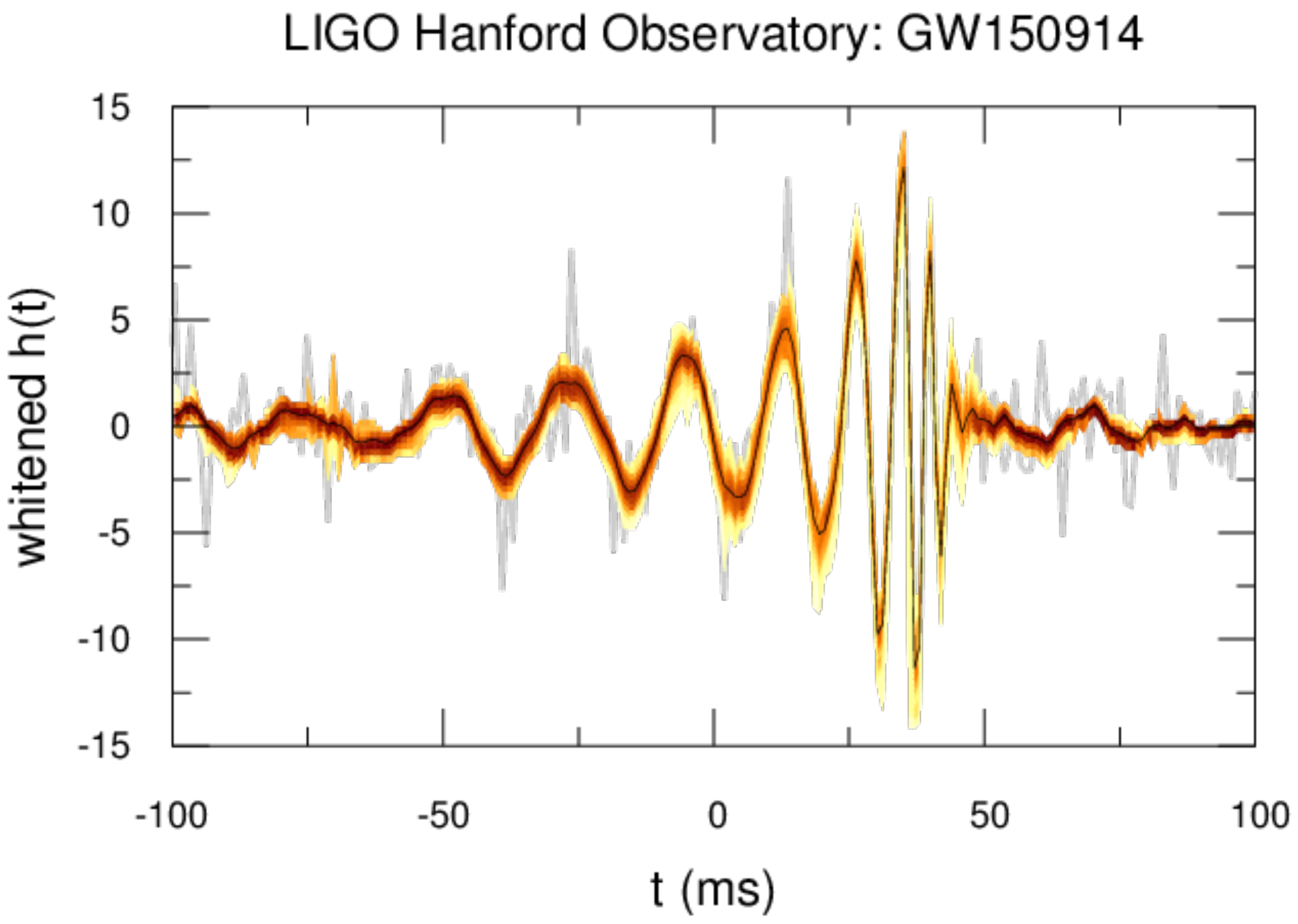
Stochastic signals $\quad p(\mathbf{h}) = \dfrac{1}{\sqrt{\det(2\pi \mathbf{S_h})}} e^{-\frac{1}{2}(\mathbf{h}^\dagger \mathbf{S_h}^{-1} \mathbf{h})}, \quad p(\mathbf{S_h})$

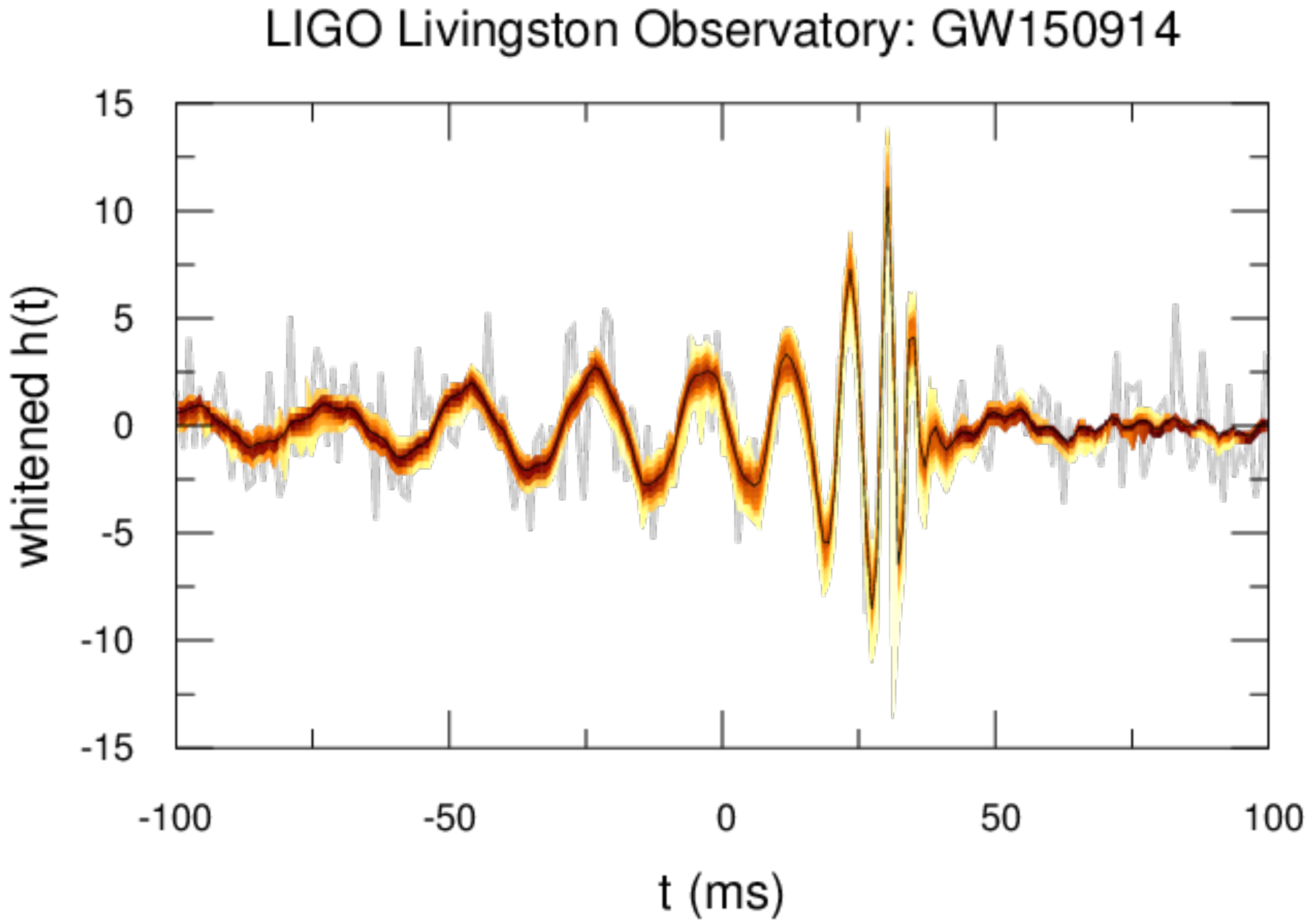
Posterior Distribution for the Waveforms

BayesWave

$p(\mathbf{h}) = \delta(\mathbf{h} - \sum$  $)p($  $)$







Models with Waveform Templates

Template based models have the strongest priors and hence yield the most sensitive searches

$$p(\mathbf{h}) = \delta(\mathbf{h} - \mathbf{h}(\vec{\lambda})), \quad p(\vec{\lambda})$$

Marginal likelihood (hierarchical Bayes)

$$p(\mathbf{d}|\vec{\lambda}) = \int p(\mathbf{d}|\mathbf{h})\delta(\mathbf{h} - \mathbf{h}(\vec{\lambda}))\, d\mathbf{h}$$

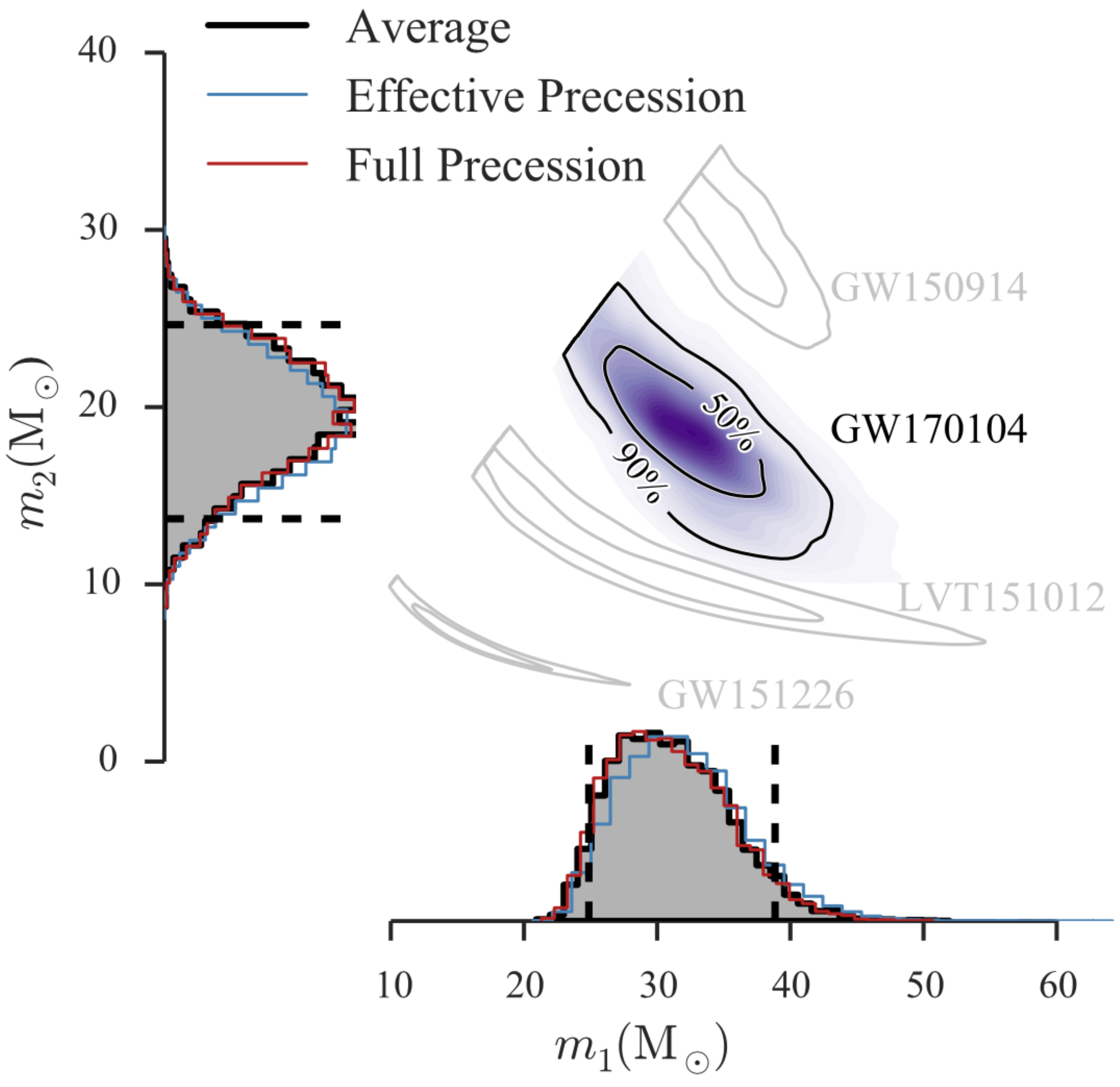
The marginal likelihood and the (hyper) prior on the model parameters then defines the posterior on the model parameters

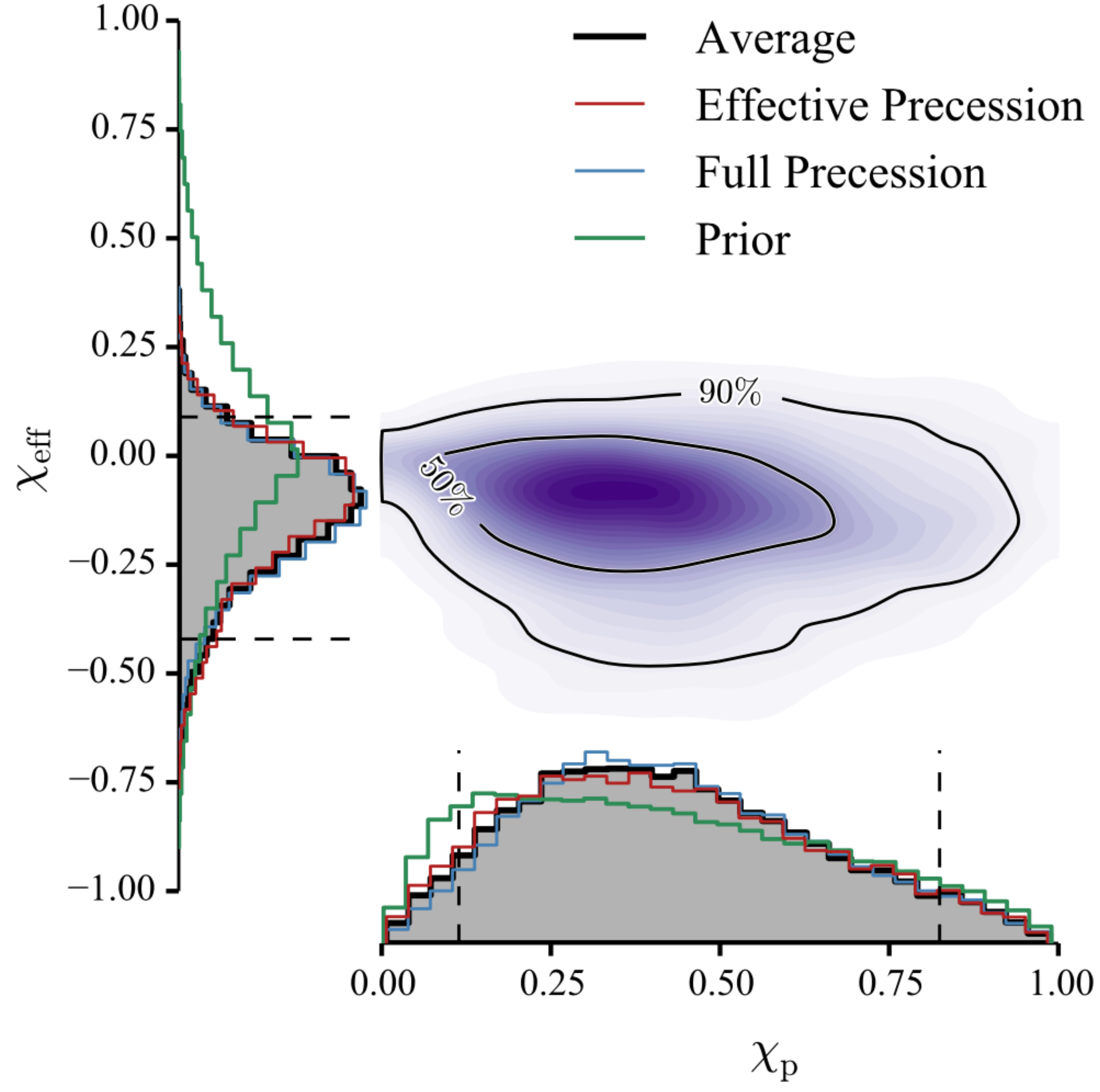
$$p(\vec{\lambda}|\mathbf{d}) = \frac{\overbrace{p(\mathbf{d}|\vec{\lambda})}^{\text{Likelihood}}\overbrace{p(\vec{\lambda})}^{\text{Prior}}}{\underbrace{p(\mathbf{d})}_{\text{Evidence}}}$$

Techniques such as MCMC and Nested Sampling can be used to map out the full posterior distribution, allowing us to compute mean, median and mode and credible intervals.

Template based analysis - Parameter Posteriors

$$p(\vec{\lambda}|\mathbf{d})$$





Stochastic Signals

$$p(\mathbf{h}) = \frac{1}{\sqrt{\det(2\pi\mathbf{S_h})}}\, e^{-\frac{1}{2}(\mathbf{h}^\dagger \mathbf{S_h}^{-1}\mathbf{h})}\; p(\mathbf{S_h})$$

We are not interested in the value of each GW signal sample $\tilde{h}(f)$. Want to infer the power spectrum $S_h(f)$

Marginalize over h:

$$p(S_h|\mathbf{d}) = \int \frac{p(\mathbf{d}|\mathbf{h})p(\mathbf{h})}{p(\mathbf{d})}\, d\mathbf{h} = \frac{p(\mathbf{d}|S_h)p(S_h)}{p(\mathbf{d})}$$

Stochastic Signals

The integration over h is easy as it just involves Gaussians [Cornish & Romano, PRD 2013]

$$p(\mathbf{d}|S_h) \propto e^{-\frac{1}{2}(\mathbf{d}|\mathbf{d})_S}$$

Where $$(\mathbf{a}|\mathbf{b})_S = 2\sum_{I,J}\int_0^\infty \left(\tilde{a}_I(f)\tilde{b}^*_J(f) + \tilde{a}^*_I(f)\tilde{b}_J(f)\right) S^{-1}_{IJ}(f)\, df$$

and $$S_{IJ}(f) = S_{n,I}(f)\,\delta_{IJ} + S_h(f)\,\gamma_{IJ}(f)$$

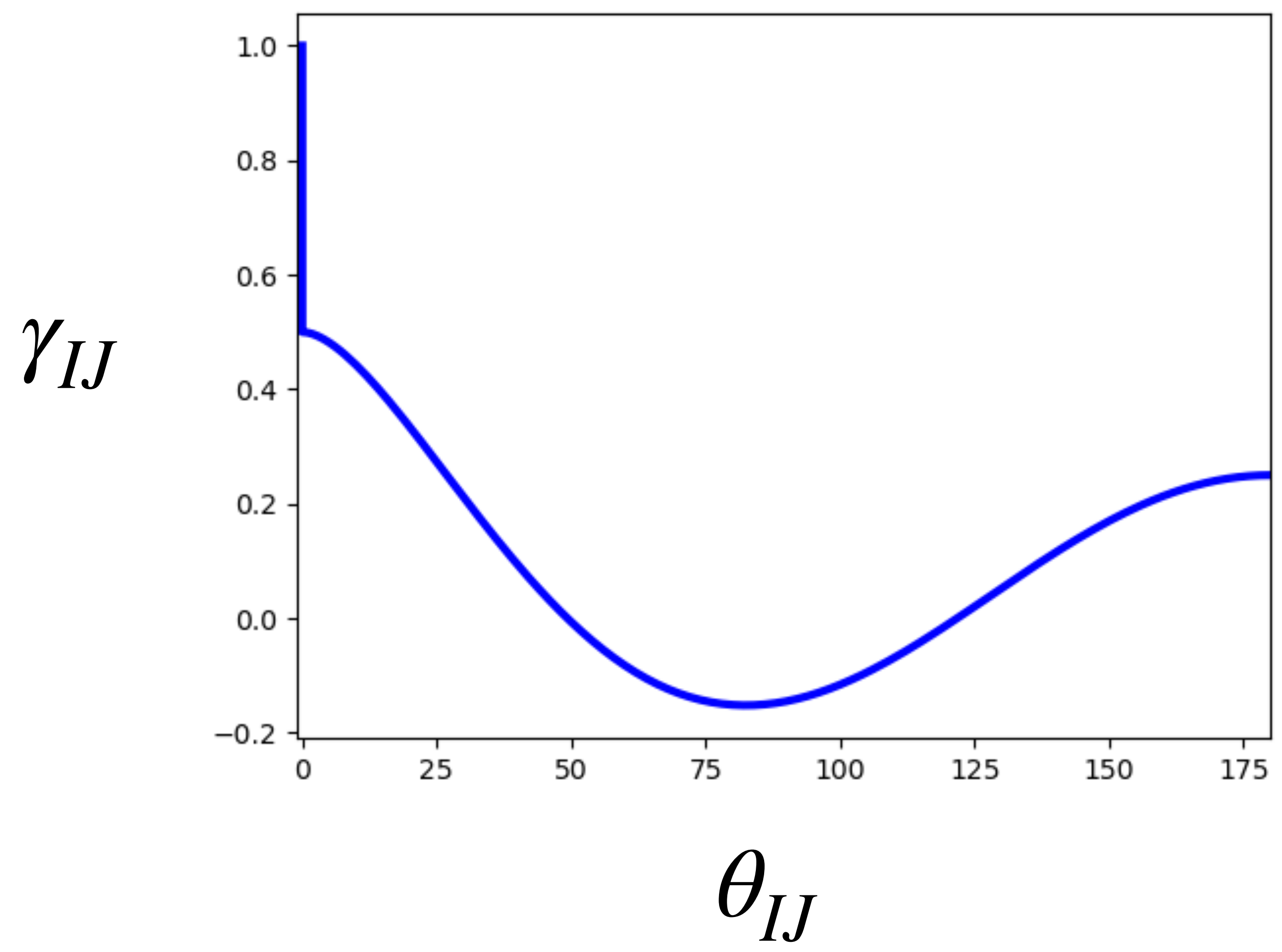
Stochastic Signals

$$S_{IJ}(f) = S_{n,I}(f)\,\delta_{IJ} + S_h(f)\,\gamma_{IJ}(f)$$

The quantity $\gamma_{IJ}(f)$ is a geometrical factor that encodes the response of the detectors. In the long wavelength limit it is called the overlap reduction function

$$\gamma_{IJ}(f) = \frac{1}{4\pi}\int (F_I^+(\hat{n})F_J^+(\hat{n}) + F_I^+(\hat{n})F_J^+(\hat{n}))e^{2\pi i f(\vec{x}_I - \vec{x}_J)\cdot\hat{n}}d\Omega_{\hat{n}}$$

In the short wavelength limit it is called the Hellings-Downs correlation curve



Frequentist Optimal Statistic

Maximizing the log likelihood ratio leads to the "optimal statistic"

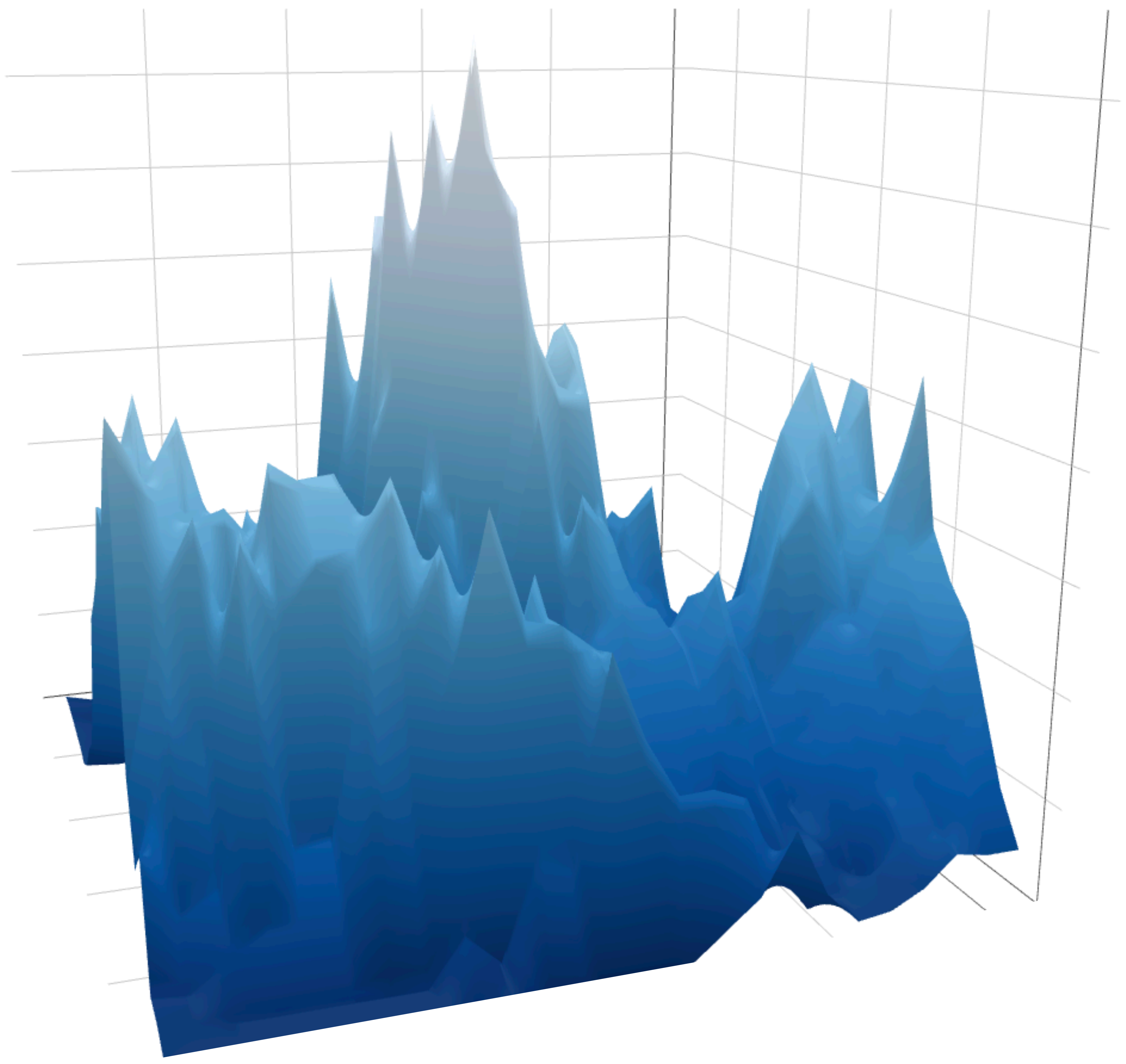
$$S = \sum_{I \neq J} \int \frac{\left(r_I(f)\, r_J^\star(f) + r_I^\star(f)\, r_J(f)\right) \gamma_{IJ}(f)\, S_h(f)}{S_I(f) S_J(f)}\, df$$

The expected SNR of the optimal statistic is

$$E[\rho] = \left[2T \sum_{I \neq J} \int \frac{\gamma_{IJ}\, S_h^2(f)}{S_I(f)\, S_J(f)}\, df \right]^{1/2}$$

[Allen & Romano PRD 1999, Anholm et al 0809.0701, Chamberlin et al 1410.8256]

The Bayesian way - all we need to do is map out the signal model and noise mode; posterior distribution in some high dimensional parameter space. How hard could that be?

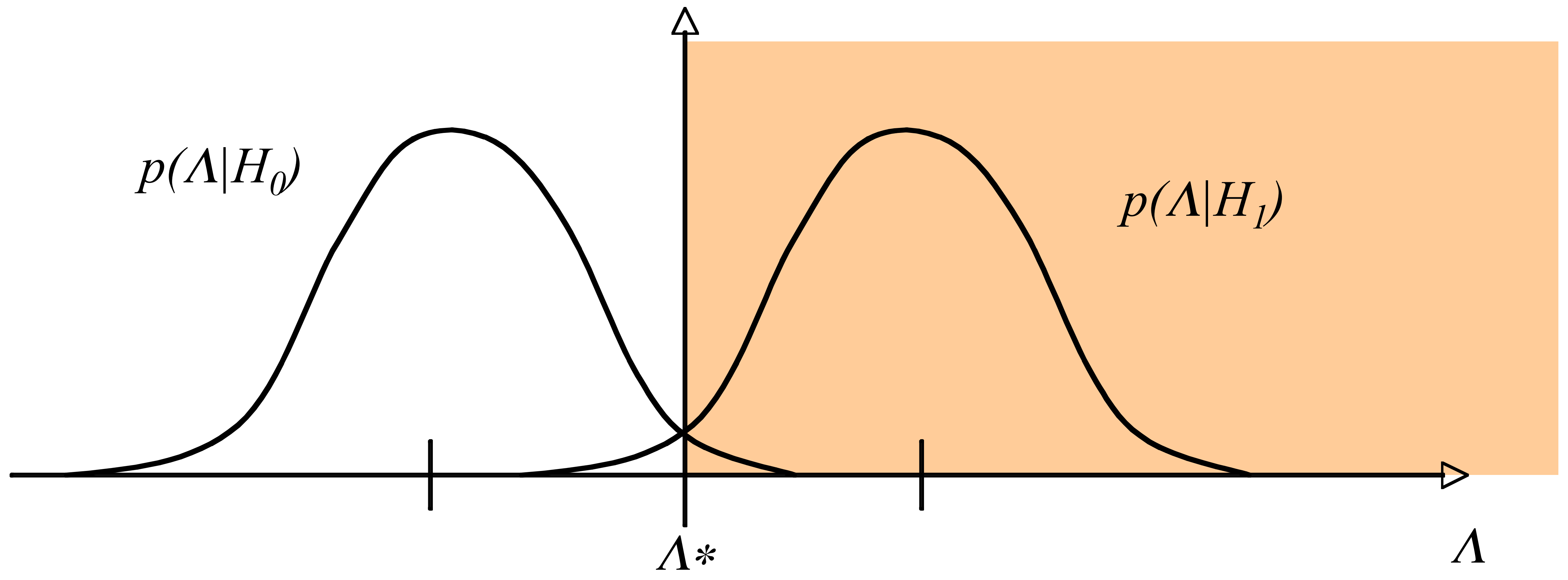


- The likelihood can be multi-modal with narrow peaks
- Noise models can involve vast numbers of parameters
- Waveform generation and/or the likelihood evaluation can be very computationally intensive
- LIGO/Virgo analyses have to sift through vast amounts of data for rare signals
- LISA analyses will have much smaller data sets to contend with, but the data will be contain millions of overlapping sources
- PTA data is unevenly sampled with complicated noise properties

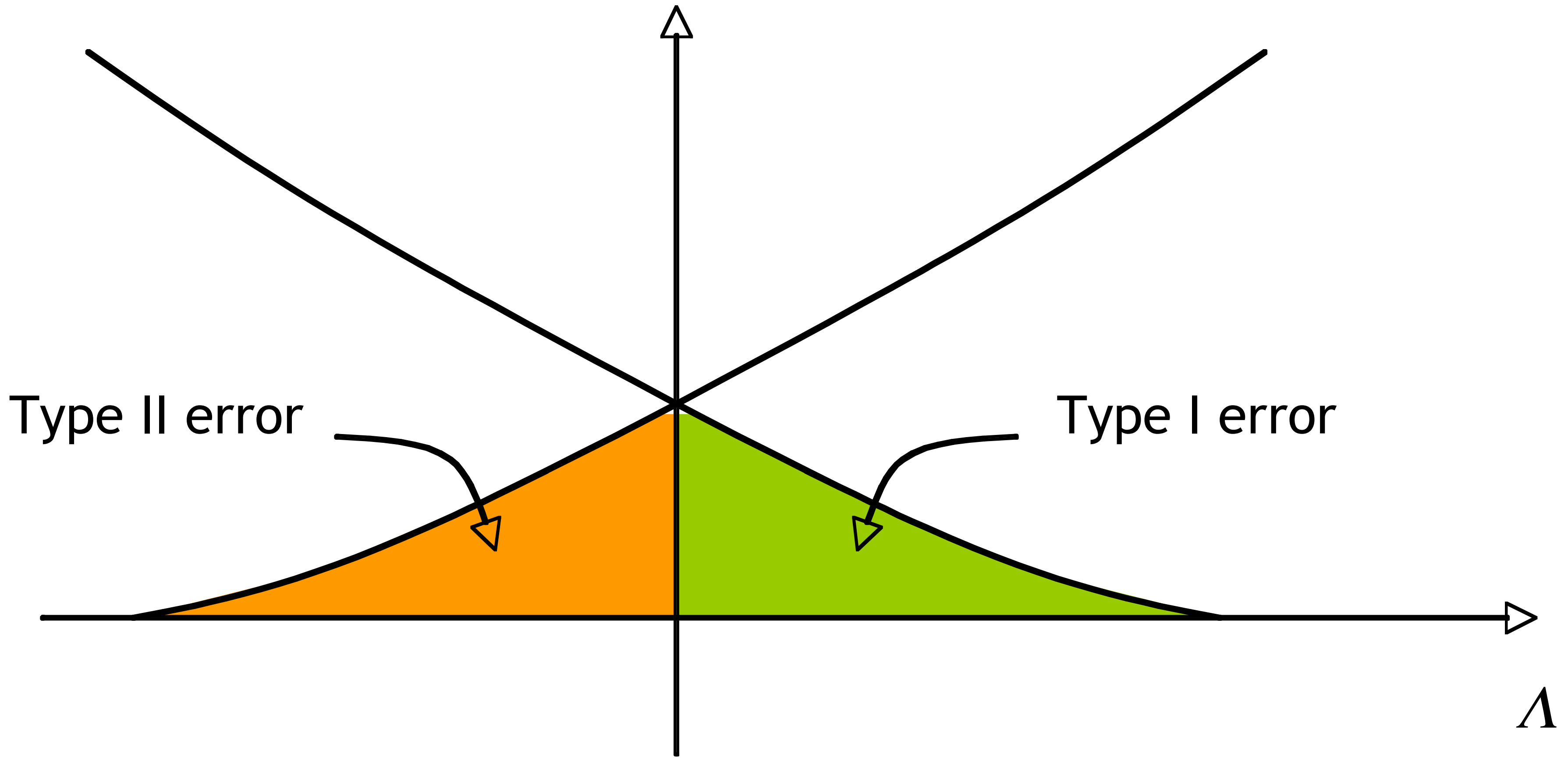
Many analyses start with a search phase followed by Bayesian parameter estimation

Searching for signals (a highly simplified treatment)

Λ – Detection Statistic $\qquad H_0$ – Noise Hypothesis $\qquad H_1$ – Noise + Signal Hypothesis



Set threshold Λ_* such that $\Lambda > \Lambda_*$ favors hypothesis H_1



Type I error - False Alarm

Type II error - False Dismissal

The likelihood ratio statistic

Likelihood for Stationary Gaussian Noise

$$p(\mathbf{d}|\vec{\lambda}) \sim e^{-(\mathbf{d}-\mathbf{h}(\vec{\lambda})|\mathbf{d}-\mathbf{h}(\vec{\lambda}))/2}$$

Noise weighted inner product

$$(\mathbf{a}|\mathbf{b}) = 2\int_0^\infty \frac{\tilde{a}(f)\tilde{b}^*(f) + \tilde{a}^*(f)\tilde{b}(f)}{S_n(f)}\,df$$

Suppose that we have two hypotheses:

H_1 : A signal with parameters $\vec{\lambda}$ is present

H_0 : No signal is present

Likelihood ratio: $$\Lambda(\vec{\lambda}) = \frac{p(\mathbf{d}|\mathbf{h}(\vec{\lambda}), H_1)}{p(\mathbf{d}, H_0)}$$

For Gaussian noise: $$\Lambda(\vec{\lambda}) = e^{-(\mathbf{d}|\mathbf{h})+\frac{1}{2}(\mathbf{h}|\mathbf{h})}$$

The ρ - statistic

For a fixed false alarm rate, the false dismissal rate
is minimized by the likelihood ratio statistic (Neyman-Pearson)

$$\Lambda(\vec{\lambda}) = \frac{p(\mathbf{d}|\mathbf{h}(\vec{\lambda}), H_1)}{p(\mathbf{d}, H_0)}$$

The likelihood ratio is maximized over the signal parameters $\overrightarrow{\lambda}$

The likelihood ratio can be maximized wrt to the overall amplitude to yield the ρ - statistic:

Writing $h(\overrightarrow{\lambda}) = \rho(\overrightarrow{\lambda})\,\hat{h}(\overrightarrow{\lambda}) \quad \Rightarrow \quad \Lambda(\vec{\lambda}) = e^{\,\rho(\mathbf{d}|\hat{h}) - \frac{1}{2}\rho^2}$

Maximizing: $\dfrac{\partial \Lambda(\vec{\lambda})}{\partial \rho} = 0$

$$\Rightarrow \quad \rho(\vec{\lambda}) = (\mathbf{d}|\hat{h}(\vec{\lambda}))$$

The ρ-statistic and SNR

The signal-to-noise ratio (SNR) is defined:

$$\begin{aligned}
\mathrm{SNR} &= \frac{\text{Expected value when signal present}}{\text{RMS value when signal absent}} \\
&= \frac{E[\rho]}{\sqrt{E[\rho_0^2] - E[\rho_0]^2}} \\
&= (h|\hat{h}) \\
&= \sqrt{(h|h)}
\end{aligned}$$

In practice, the detector noise is not perfectly Gaussian, and variants of the ρ - statistic are now used, notably the "new SNR" statistic, introduced by B. Allen Phys.Rev. D71 (2005) 062001

$$\boxed{\mathrm{SNR}^2 = 4\int_0^\infty \frac{|\tilde{h}(f)|^2}{S_n(f)}\,df}$$

Frequentist Detection Threshold

For stationary, Gaussian noise the ρ - statistic is Gaussian distributed.

For the null hypothesis we have $$p_0(\rho) = \frac{1}{\sqrt{2\pi}}\, e^{-\rho^2/2}$$

For the detection hypothesis we have $$p_1(\rho) = \frac{1}{\sqrt{2\pi}}\, e^{-(\rho^2-\mathrm{SNR}^2)/2}$$

Setting a threshold of ρ_* gives the false alarm and false dismissal probabilities

$$P_{\rm FA} = \frac{1}{2}\mathrm{erfc}(\rho_*/\sqrt{2})$$

$$P_{\rm FD} = \frac{1}{2}\mathrm{erfc}((\rho_* - \mathrm{SNR})/\sqrt{2})$$

LIGO/Virgo analyses do not use SNR thresholds, but rather use False Alarm Rate thresholds

$$\mathrm{FAR} = \frac{P_{\rm FA}}{T_{\rm obs}}$$

e.g. FAR = One in million years and an observation time of one year

$$P_{\rm FA} = 10^{-6} \quad \text{aka} \quad 4.9\,\sigma \qquad\qquad \rho_* = 4.8$$

Grid Based Searches

Goal is to lay out a grid in parameter space that is fine enough to catch any signal with some good fraction of the maximum matched filter SNR

The match measures the fractional loss in SNR in recovering a signal with a template and defines a natural metric on parameter space:

$$M(\vec{x},\vec{y}) = \frac{(h(\vec{x})|h(\vec{y}))}{\sqrt{(h(\vec{x})|h(\vec{x}))(h(\vec{y})|h(\vec{y}))}}$$

Taylor expanding $\quad M(\vec{x},\vec{x}+\Delta\vec{x}) = 1 - g_{ij}\Delta x^i \Delta x^j + \dots$

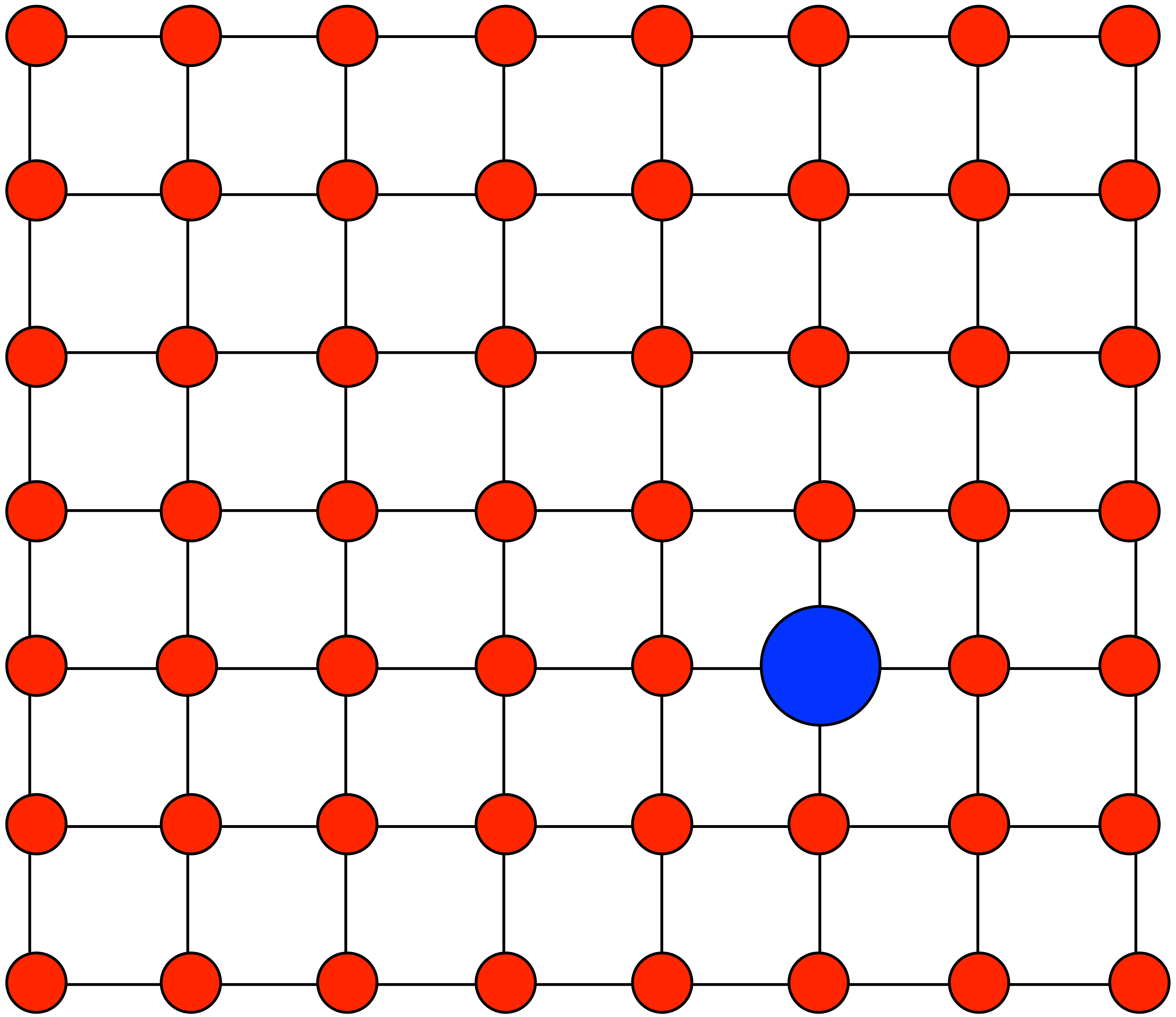
where $\quad g_{ij} = \frac{(h_{,i}|h_{,j})}{(h|h)} - \frac{(h|h_{,i})(h|h_{,j})}{(h|h)^2}$ (Owen Metric)

Number of templates (for a hypercube lattice in D dimensions)

$$N = \frac{V}{\Delta V} = \frac{\int d^D x \sqrt{g}}{(2\sqrt{(1-M_{\rm min})}/D)^D}$$

Cost grows geometrically with D for any lattice

LIGO Style Grid Searches



Typically 2-3 dimensional, 1000's points

Reducing the cost of a search

Phase Offset:

Generate two templates $h(\phi = 0)$ and $h(\phi = \pi/2)$

Then $(d|h)_{\max\,\phi} = \sqrt{(d|h(0))^2 + (d|h(\pi/2))^2}$

Easy to see this in the Fourier domain.

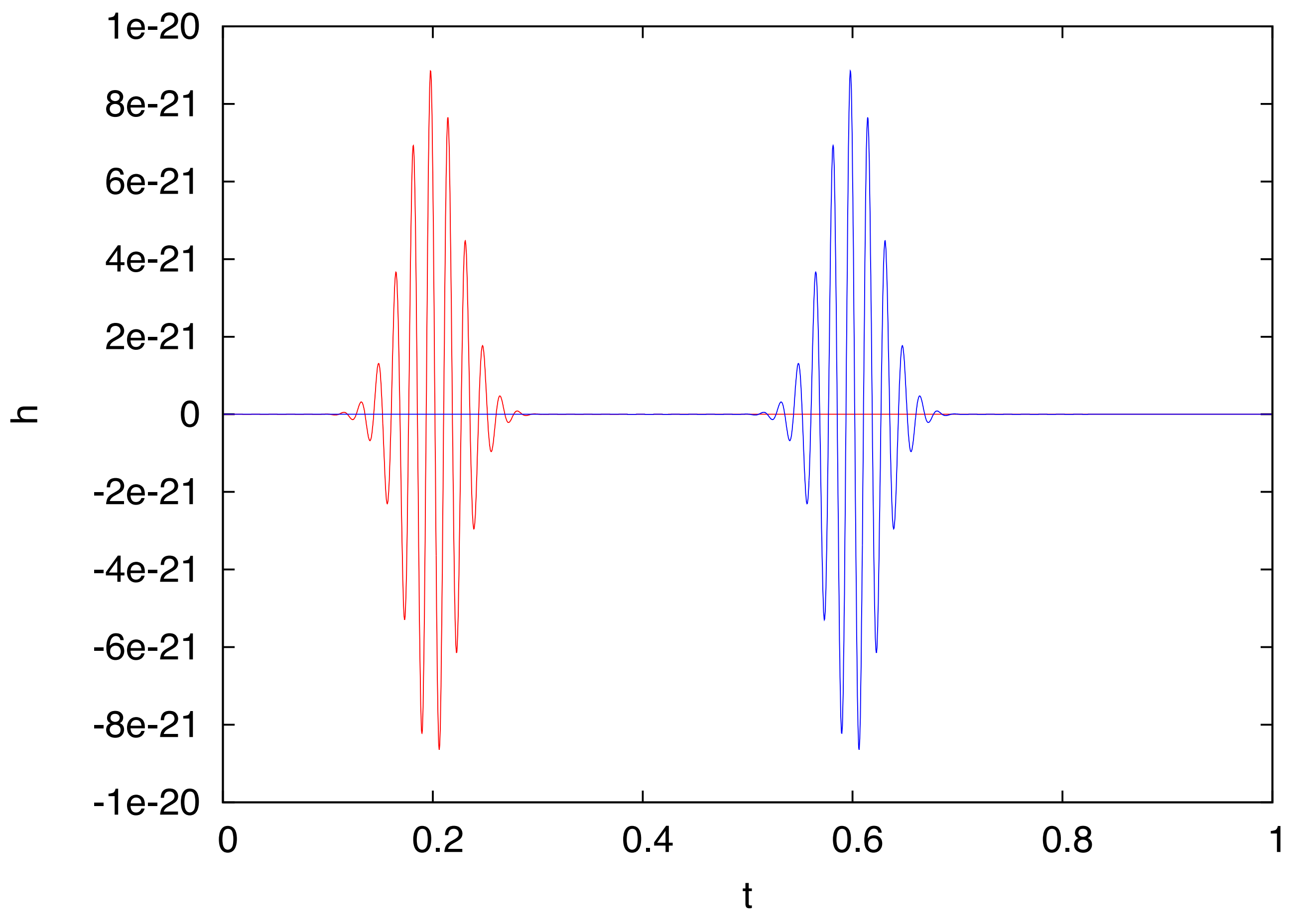
Suppose $\tilde{d} = \tilde{h}_0\, e^{i\phi}$, then

$$(d|h(0)) = (h_0|h_0)\cos\phi$$

$$(d|h(\pi/2)) = (h_0|h_0)\sin\phi$$

Reducing the cost of a search

Time Offset:



Fourier transform treats time as periodic - use this to our advantage

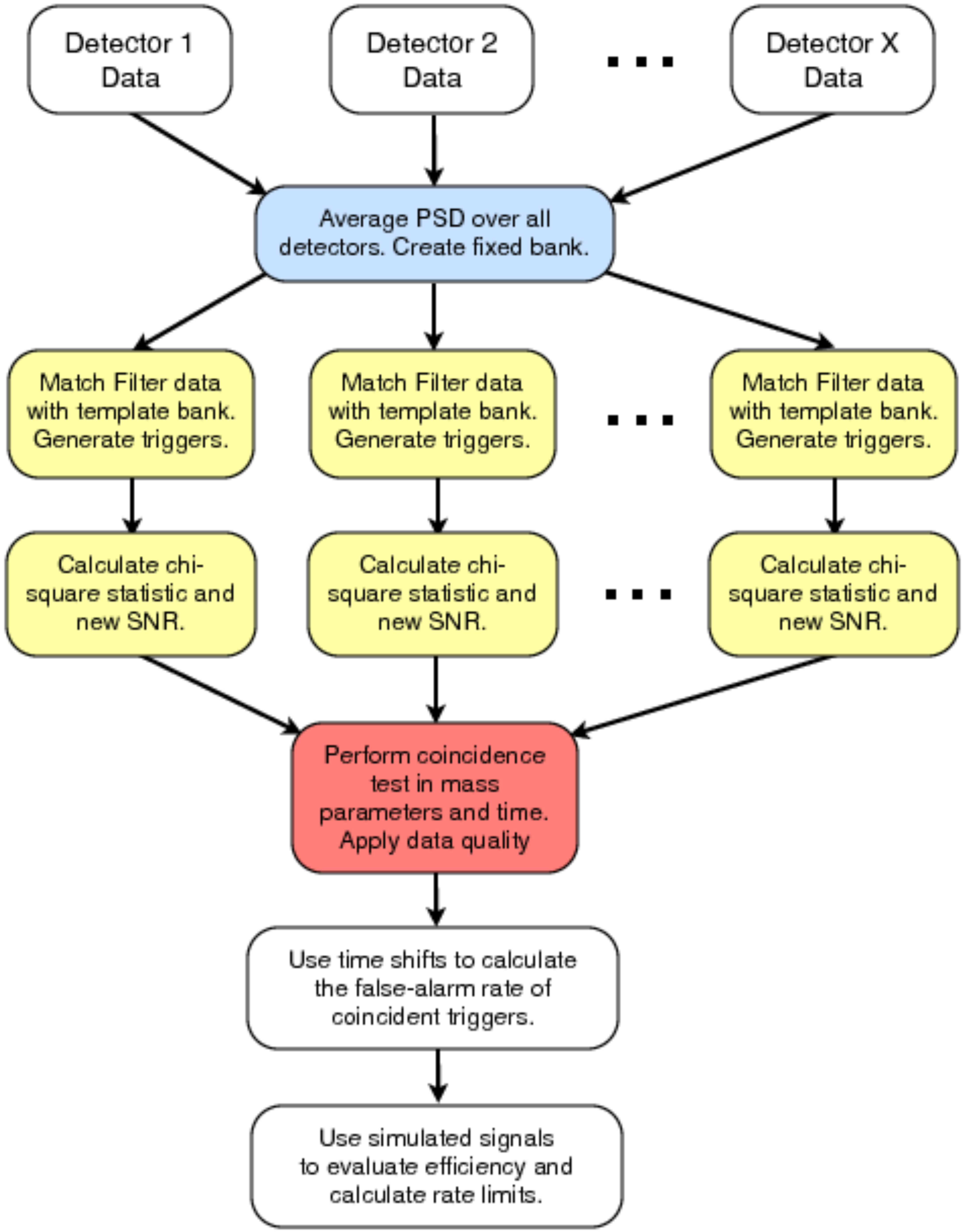
Compute the inverse Fourier transform of the product of the Fourier transforms:

$$(d|h)(\Delta t) = 4 \int \frac{\tilde{d}^*(f)\tilde{h}(f)}{S(f)} \, e^{2\pi i f \Delta t} \, df$$

Then if the template and data differ by a time shift: $d(t) = h(t - t_0)$

$$(d|h)_{\max\ t} = (d|h)(\Delta t = t_0)$$

Workflow for pyCBC search



Template bank constructed

Matched filtering is done per-detector (not coherent)

Detection statistic computed ("new SNR")

Coincidence in time/mass enforced
Data quality vetoes applied

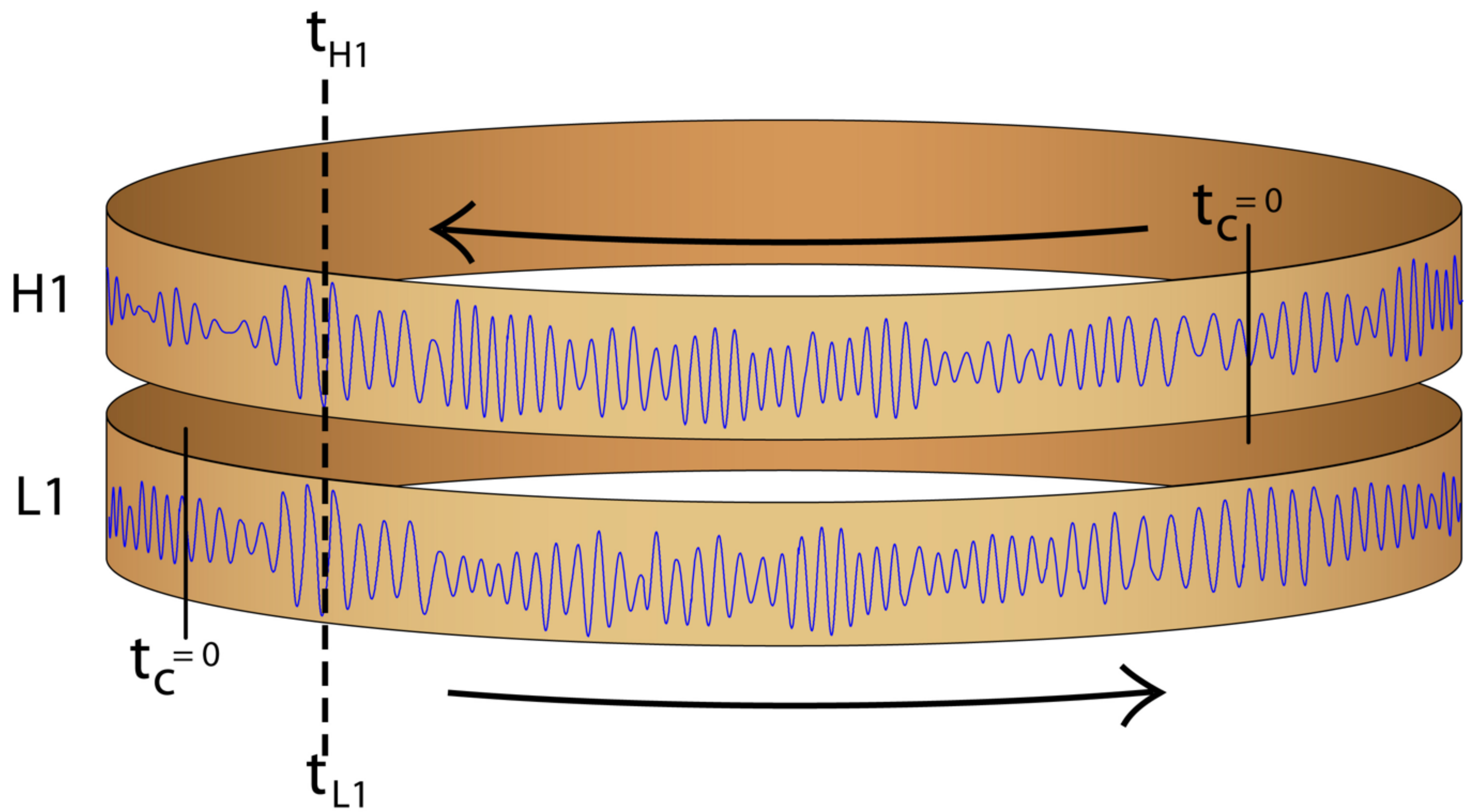
Monte Carlo background to compute FAR vs new SNR

Contending with non-stationary, non-Gaussian noise

Non-stationary: → Adiabatic drifts in the PSD — - work with short data segments

Non-stationary: → Glitches — - vetoes and time-slides

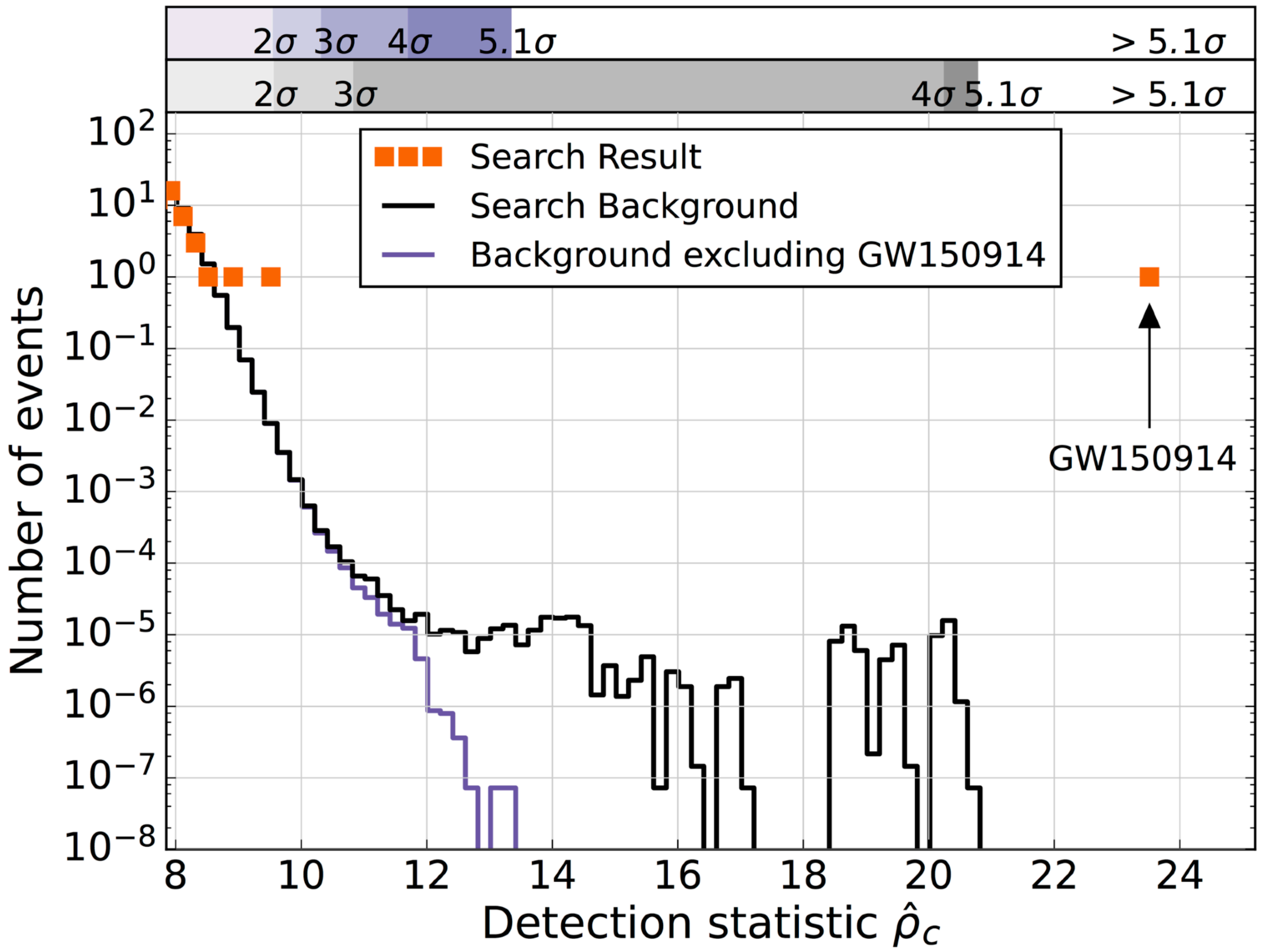
Non-Gaussian: → Glitches



Non-Gaussian Noise Transients

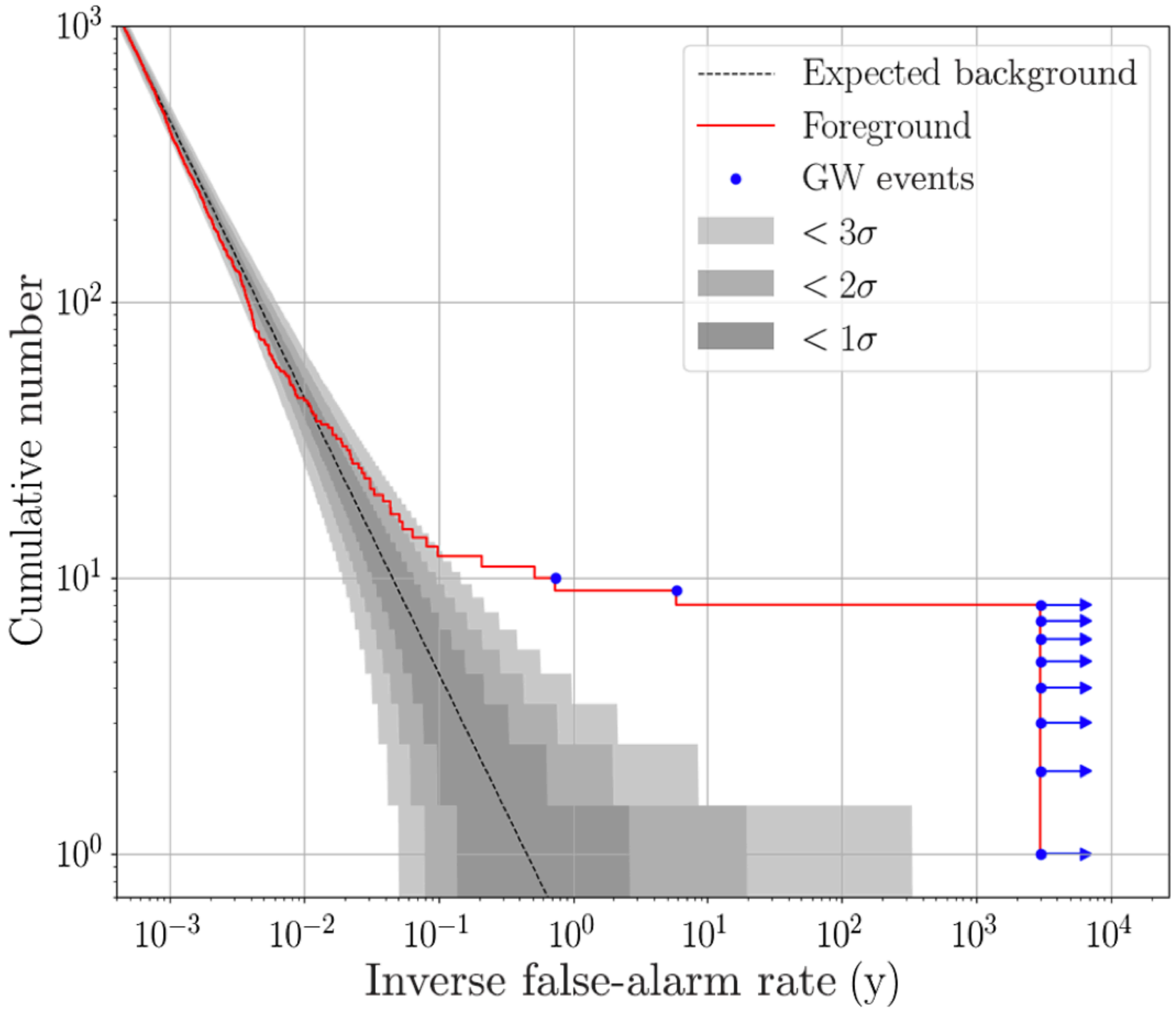
Search results

First detection



"new SNR"

GWTC-1 (O1 & O2)



Bayesian Inference

- Bayesian Probability Theory
- Bayesian Learning
- Model Selection
- Markov Chain Monte Carlo

Bayesian Parameter Estimation

Degree of belief interpretation of probability - the natural expression of the scientific method

Initial Understanding	$\Rightarrow$	New Observations	$\Rightarrow$	Updated Understanding
$p(\vec{x})$		$p(d\|\vec{x})$		$p(\vec{x}\|d)$
Prior	$\Rightarrow$	Likelihood	$\Rightarrow$	Posterior

Bayes' Theorem

$$p(\vec{x}|d) = \frac{p(\vec{x})p(d|\vec{x})}{p(d)}$$

Normalization factor is the marginal likelihood or ***evidence***

$$p(d) = \int p(\vec{x})p(d|\vec{x})\, d\vec{x}$$

Bayesian Probability Theory

The posterior distribution fully characterizes the model.

E.g. expectation values

$$E[x^i] = \int x^i \, p(\vec{x}|d) \, d\vec{x}$$

E.g. single parameter probability distributions

$$p(x^i|d) = \int p(\vec{x}|d) \, dx^1 dx^2 \ldots dx^{i-1} dx^{i+1} \ldots dx^D$$

E.g. quantile regions, such as 90%

$$\begin{aligned} 0.05 &= \int^{x_1} p(x|d) \, dx \\ 0.9 &= \int_{x_1}^{x_2} p(x|d) \, dx \end{aligned}$$

Bayesian Learning

"Today's posterior is tomorrow's prior" - Lindley

The amount we learn from the data can be measured in bits, and can be computed in terms of the Kullback–Leibler divergence

$$D_{KL} = \int p(\vec{x}|d) \, \log_2 \left(\frac{p(\vec{x}|d)}{p(\vec{x})} \right) \, d\vec{x} \quad [\text{bits}]$$

Bayesian Model Selection

Probability of Model M: $$p(M|d) \propto \underbrace{p(M)}_{\text{Prior Probability of M}}\underbrace{p(d|M)}_{\text{Evidence for M}}$$

Odds Ratio:
$$\begin{aligned} O_{ij} &= \frac{p(M_i|d)}{p(M_j|d)} \\ &= \frac{p(M_i)}{p(M_j)}\frac{p(d|M_i)}{p(d|M_j)} \\ &= \text{Prior Odds Ratio} \times \text{Bayes Factor} \end{aligned}$$

More on how we compute the Bayes Factor later...

Bayesian Machinery: Markov Chain Monte Carlo

Bayes' Theorem $$p(\vec{x}|d) = \frac{p(\vec{x})p(d|\vec{x})}{p(d)}$$

Marginal likelihood or evidence $$p(d) = \int p(\vec{x})p(d|\vec{x})\, d\vec{x}$$

We know how to compute the prior and the likelihood. The difficulty lies in computing the evidence.

The MCMC technique, introduced by Metropolis and developed by Hastings, allows us to simulated samples from the posterior distribution directly, without having to compute the evidence.

It is possible to compute the evidence using augmented MCMC techniques. Another powerful technique for computing the evidence and the posterior distributions is *Nested Sampling*

Bayesian Inference

Prior

$p(h|M)$

Likelihood

$p(d|h)$

$p(d|M)$

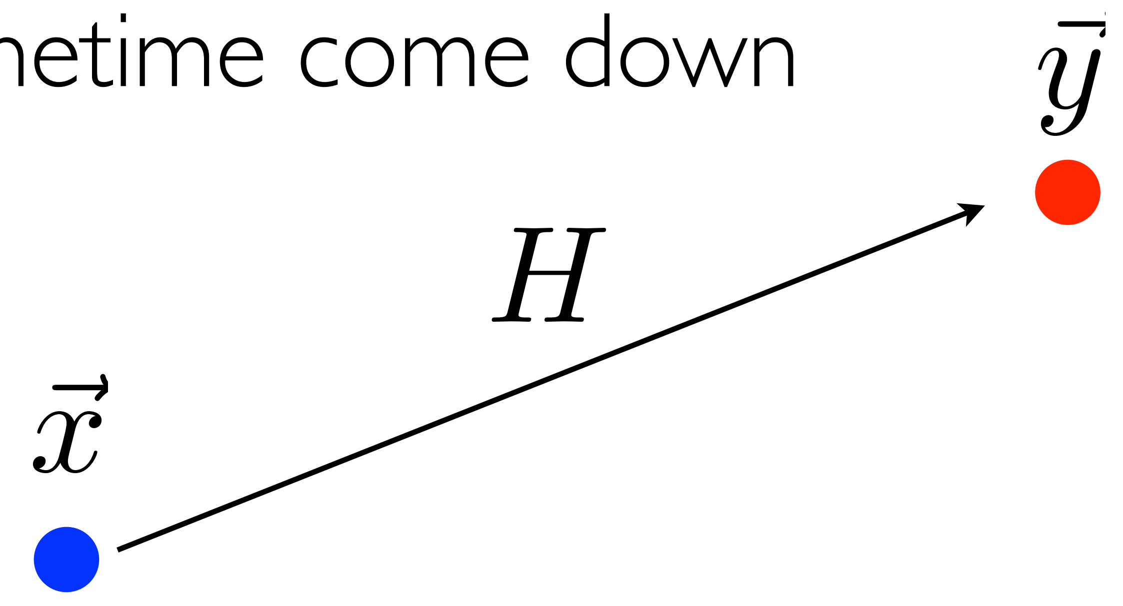
Evidence

$p(h|d,M)$

Posterior

Markov Chain Monte Carlo

Always go up,
Sometime come down



Yields PDF $p(\vec{x}|d)$ for parameters $\vec{x}$ given data d

$$H = \min\left(1, \frac{p(\vec{y})\,p(d|\vec{y})\,q(\vec{x}|\vec{y})}{p(\vec{x})\,p(d|\vec{x})\,q(\vec{y}|\vec{x})}\right)$$

Prior

Likelihood

Proposal

Transition Probability
(Metropolis-Hastings)

Markov Chain Monte Carlo

The choice of jump proposal $q(\vec{y}|\vec{x})$ is key to convergence

Convergence to the target distribution has two facets:

- Burn-in (finding the dominant modes of the posterior)

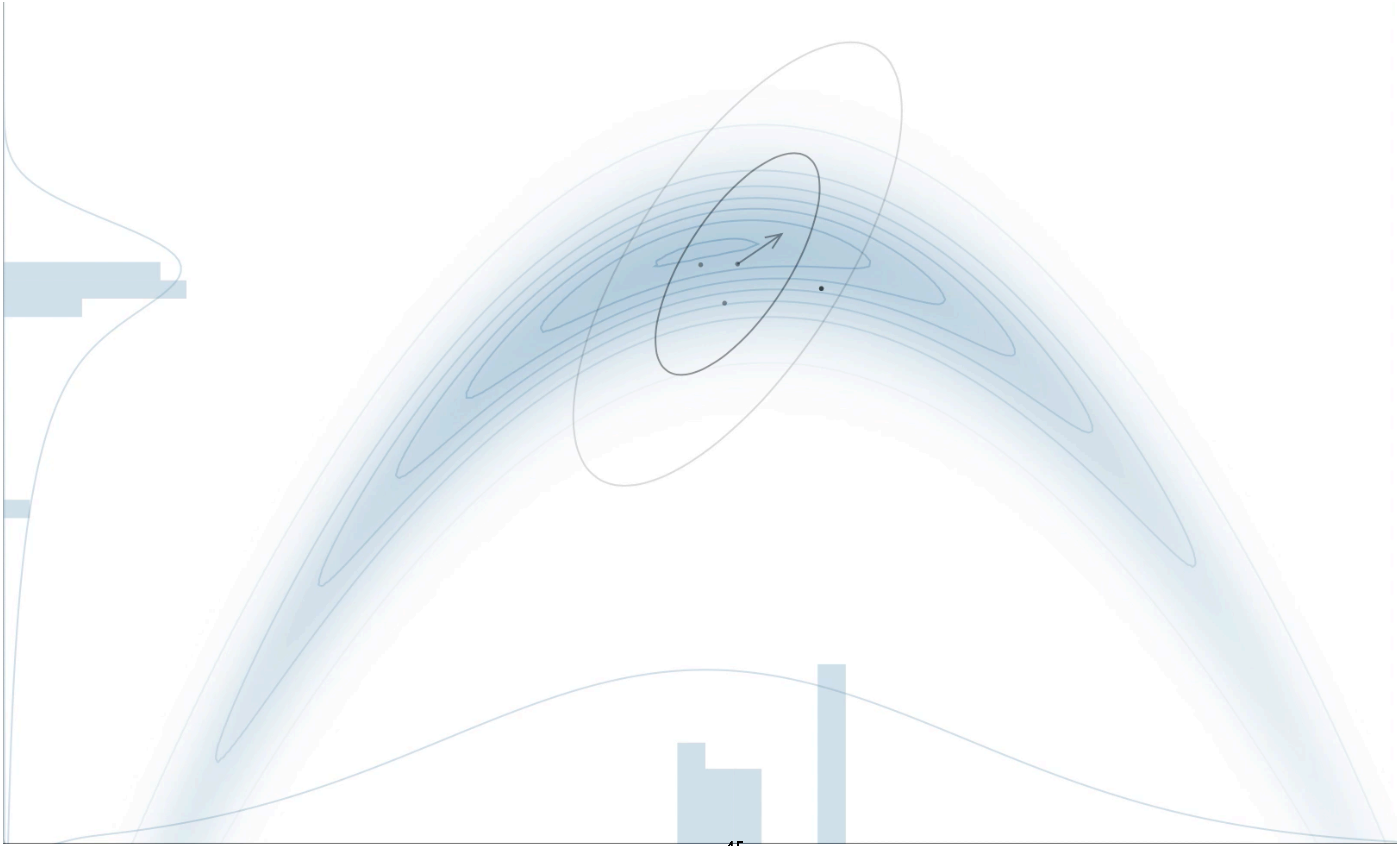
- Mixing (exploring the dominant modes of the posterior)

The perfect proposal distribution is the posterior distribution itself, $q(\vec{y}|\vec{x}) = p(\vec{y}|d)$, since then

$$\begin{aligned} H &= \min\left(1, \frac{p(\vec{y})p(d|\vec{y})p(\vec{x}|d)}{p(\vec{x})p(d|\vec{x})p(\vec{y}|d)}\right) \\ &= 1 \end{aligned}$$

But if we knew the posterior distribution in advance there would be no need for the MCMC procedure! Instead we seek ways to approximate the posterior.

https://chi-feng.github.io/mcmc-demo/app.html?algorithm=RandomWalkMH&target=banana



MCMC à la Montana

Ingredients:

Local posterior approximation

Global likelihood maps

Differential evolution proposals

Parallel tempering

Directions:

Mix all the proposals together. Check consistency by recovering the prior and diagonal PP plots. Results are ready when distributions are stationary.

Proposal Distributions

Local posterior approximation

Quadratic approximation to the posterior using the augmented Fisher Information Matrix

$$q(\vec{y}|\vec{x}) = \frac{1}{\sqrt{\det(2\pi\mathbf{K}^{-1})}} e^{-\frac{1}{2}K_{ij}(x^i - y^i)(x^j - y^j)}$$

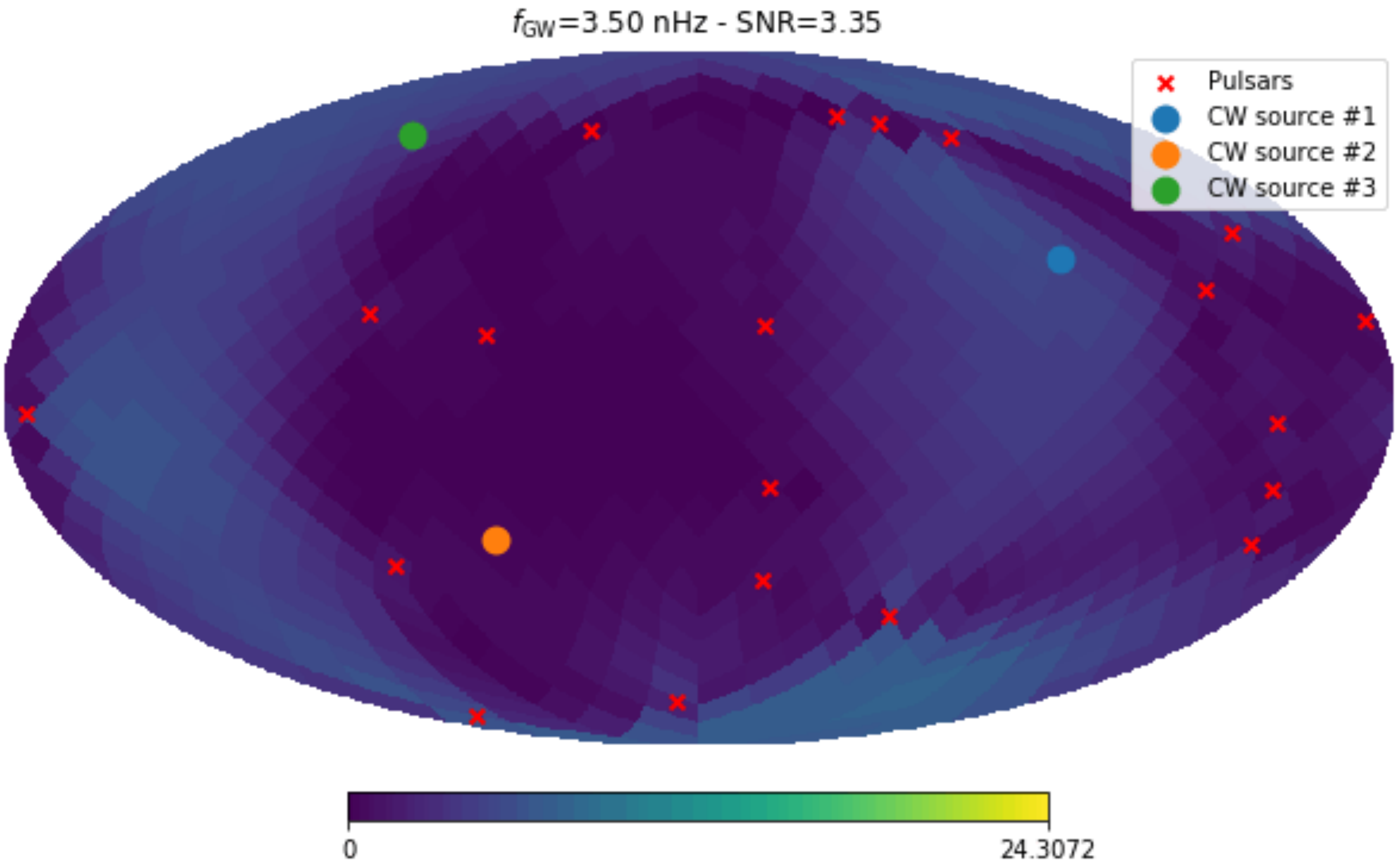
Propose jumps along eigendirections of **K**, scaled by eigenvalues

Global likelihood maps

Use a Non-Markovian Pilot search (max likelihood, hill climbers, simulated annealing, etc) to crudely map the posterior/likelihood and use this as a proposal distribution for a Markovian follow-up [Littenberg & Cornish, PRD 80, 063007, (2009)]

Time-frequency maps, Maximized likelihood maps

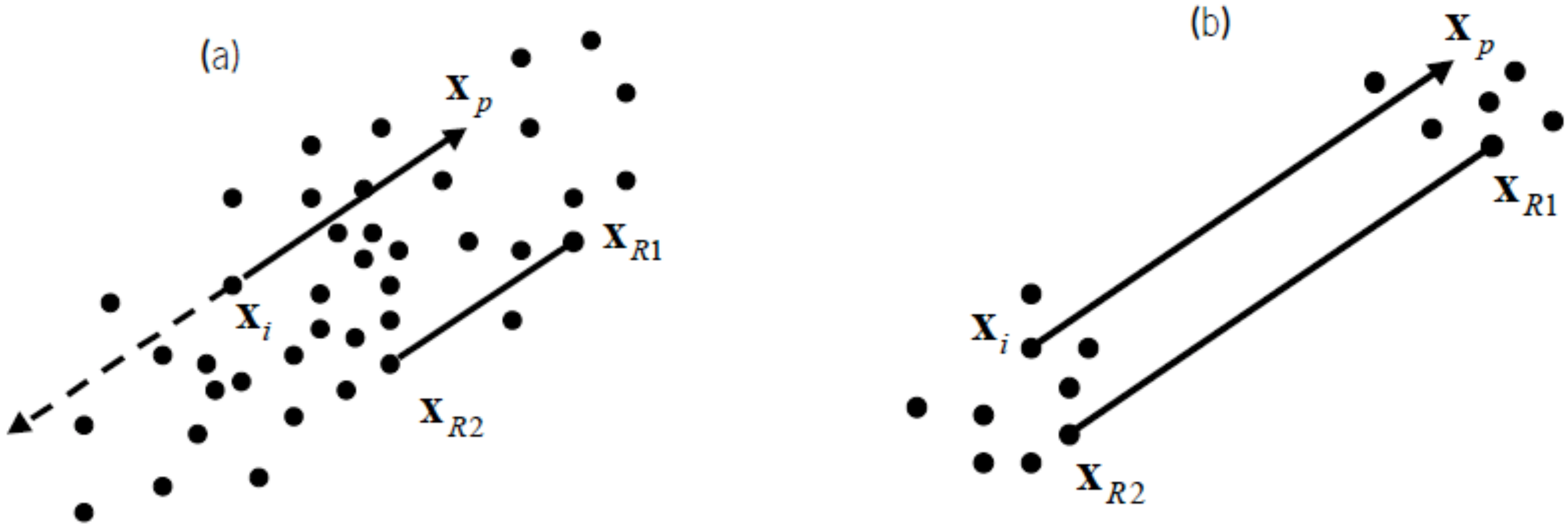
Global F-statistic based proposal



Proposal Distributions

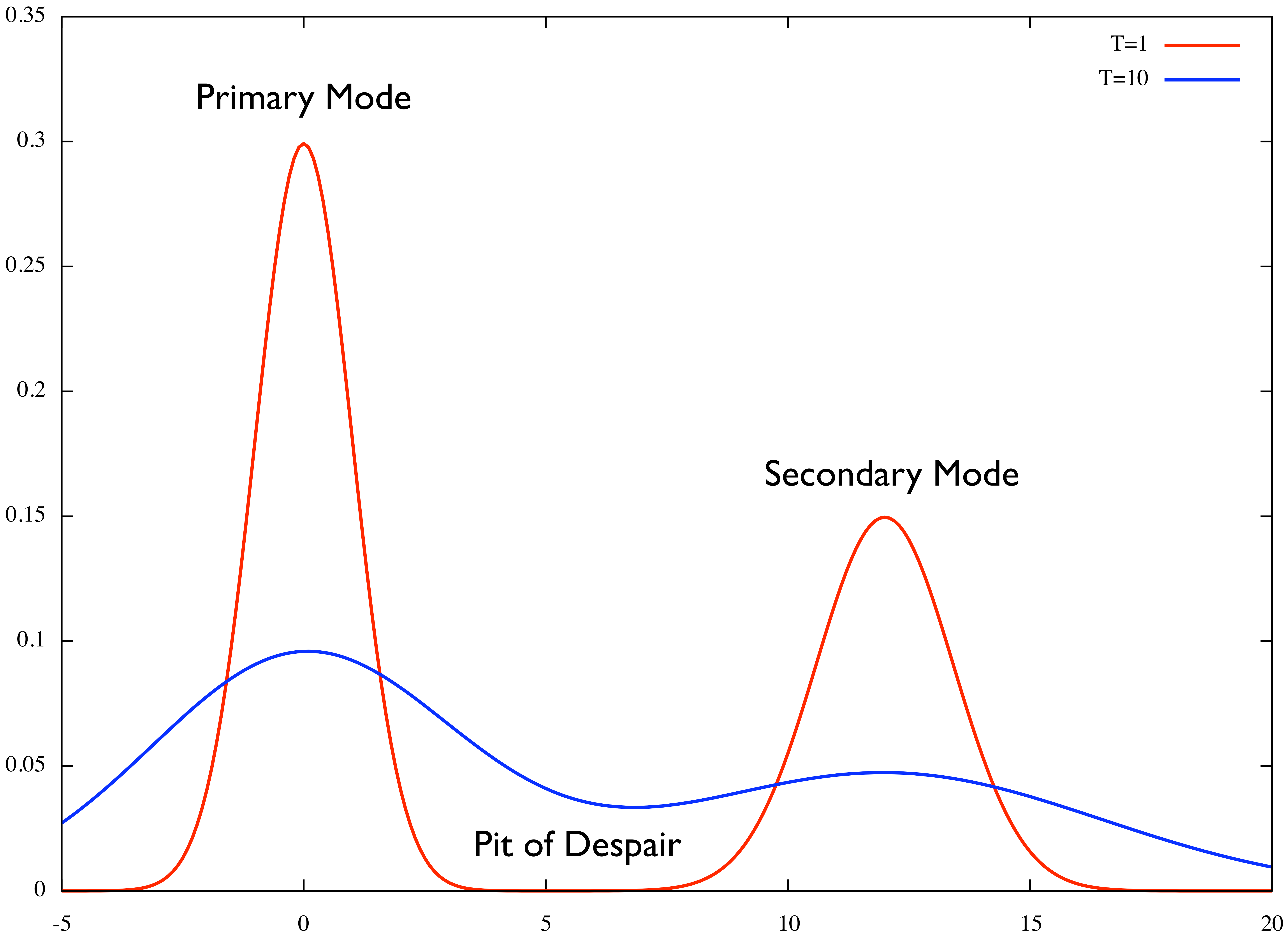
Differential evolution

[Braak (2005)]



Parallel Tempering

[Swendsen & Wang, 1986]



Ordinary MCMC techniques side-step the need to compute the evidence. PT uses multiple, coupled chains to improve mixing, and also allows the evidence to be computed.

Explore tempered posterior

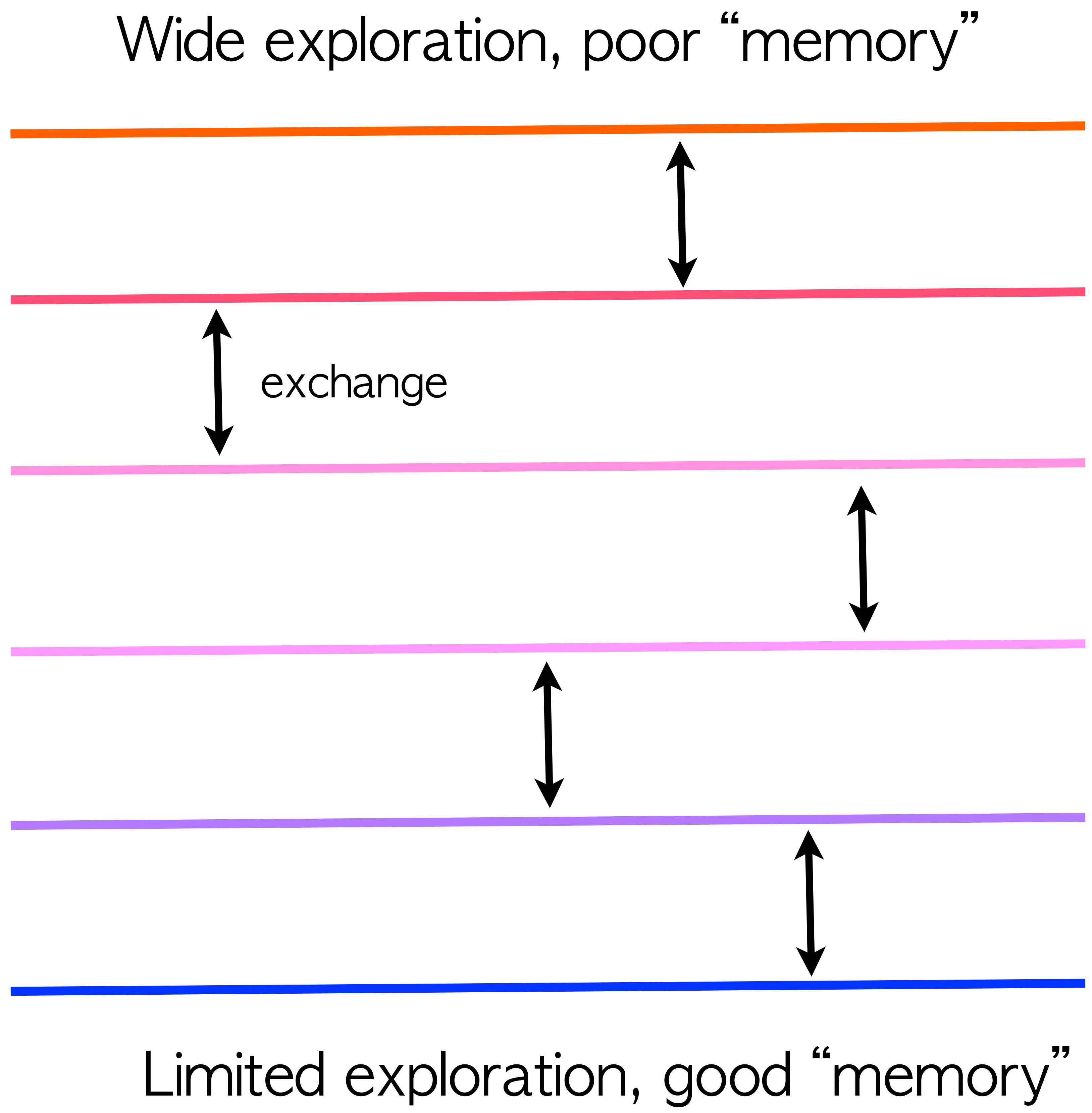
$$\pi(\vec{\lambda}|\mathbf{d})_T = p(\mathbf{d}|\vec{\lambda})^{1/T} p(\vec{\lambda})$$

Compute model evidence

$$\log p(\mathbf{d}) = \int_0^1 \mathbb{E}[\log p(\mathbf{d}|\vec{\lambda})]_\beta \, d\beta$$

(Here $\beta = \frac{1}{T}$)

Parallel Tempering

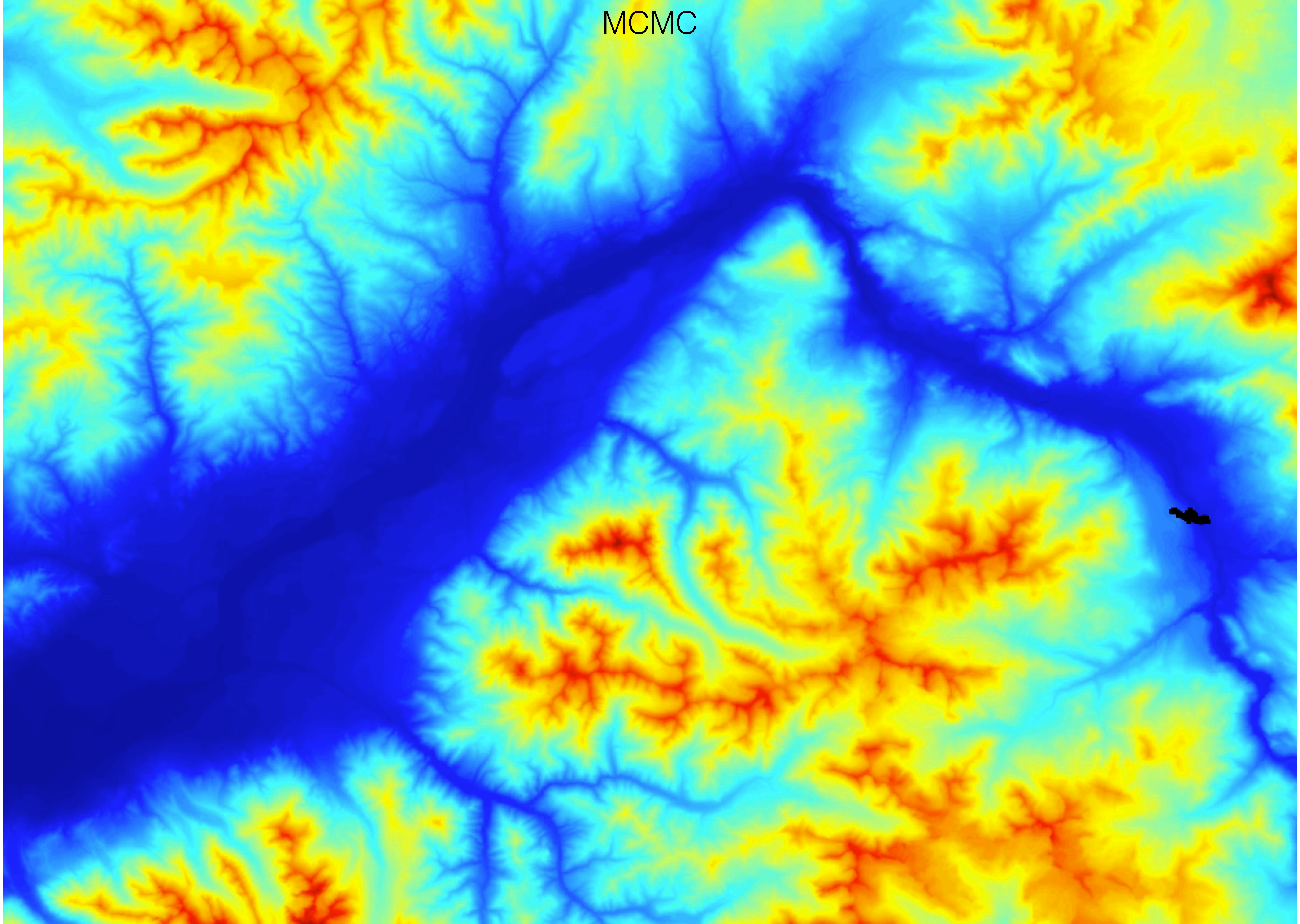


Inter-chain transition probability

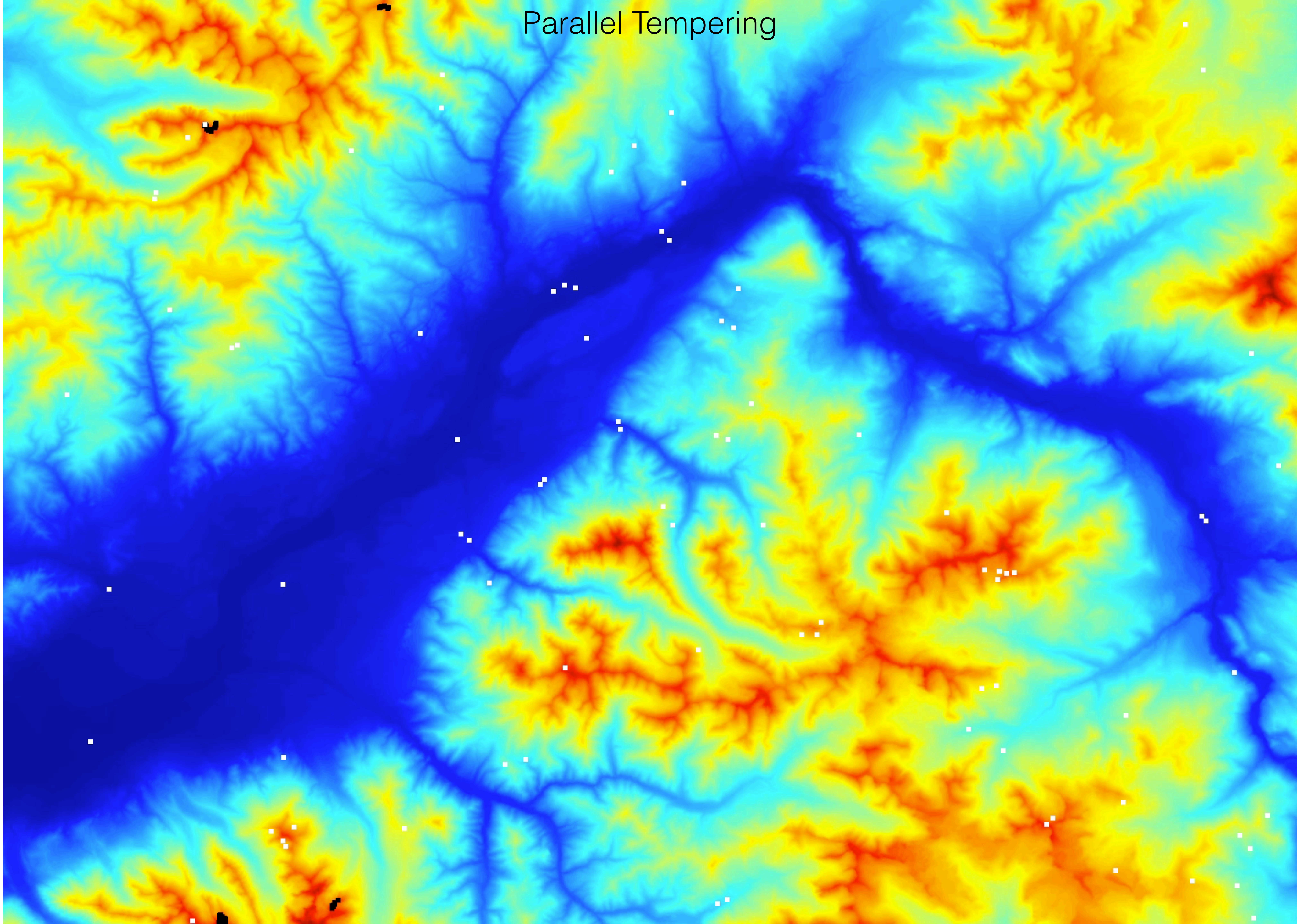
$$H = \min\left(1,\, \frac{\pi(\vec{\lambda}_{j+1}|\mathbf{d})_{T_j}\,\pi(\vec{\lambda}_j|\mathbf{d})_{T_{j+1}}}{\pi(\vec{\lambda}_j|\mathbf{d})_{T_j}\,\pi(\vec{\lambda}_{j+1}|\mathbf{d})_{T_{j+1}}}\right)$$

Good solutions sink down ladder and get stored

MCMC



Parallel Tempering

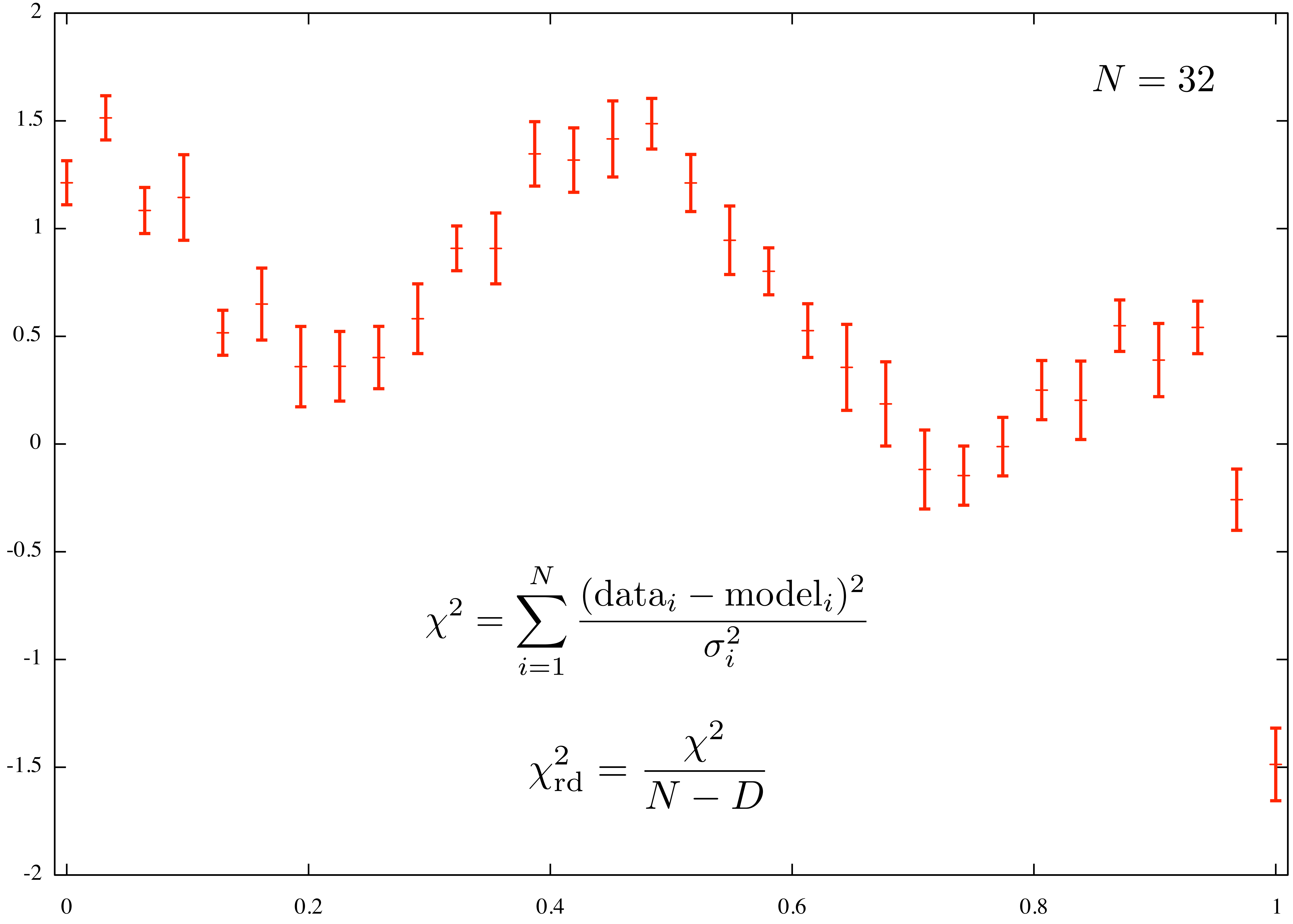


Model everything and let the data sort it out

“Caedite eos. Novit enim Dominus qui sunt eius”

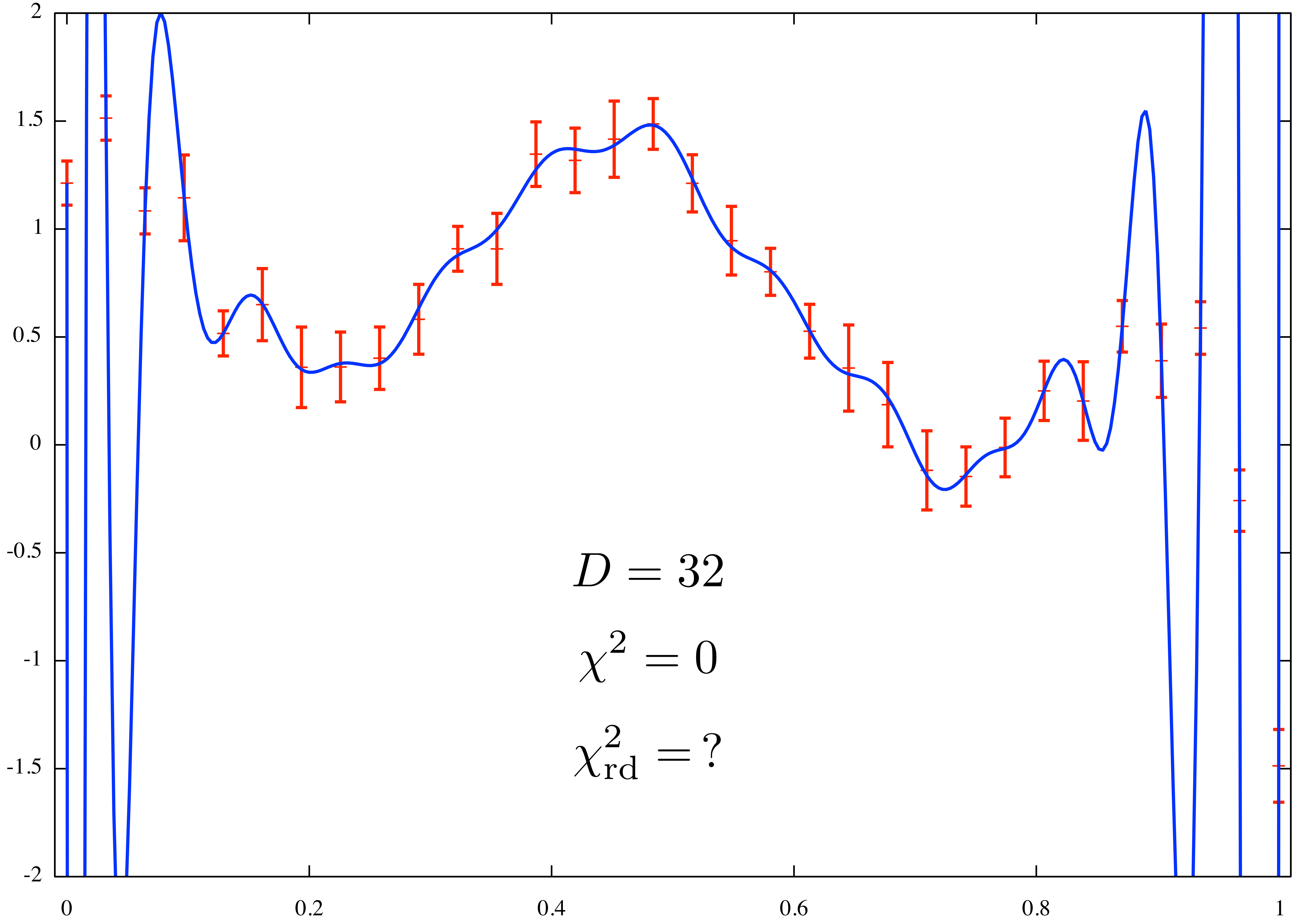
Arnaud Amalric, Papal Legate
(massacre of Béziers)

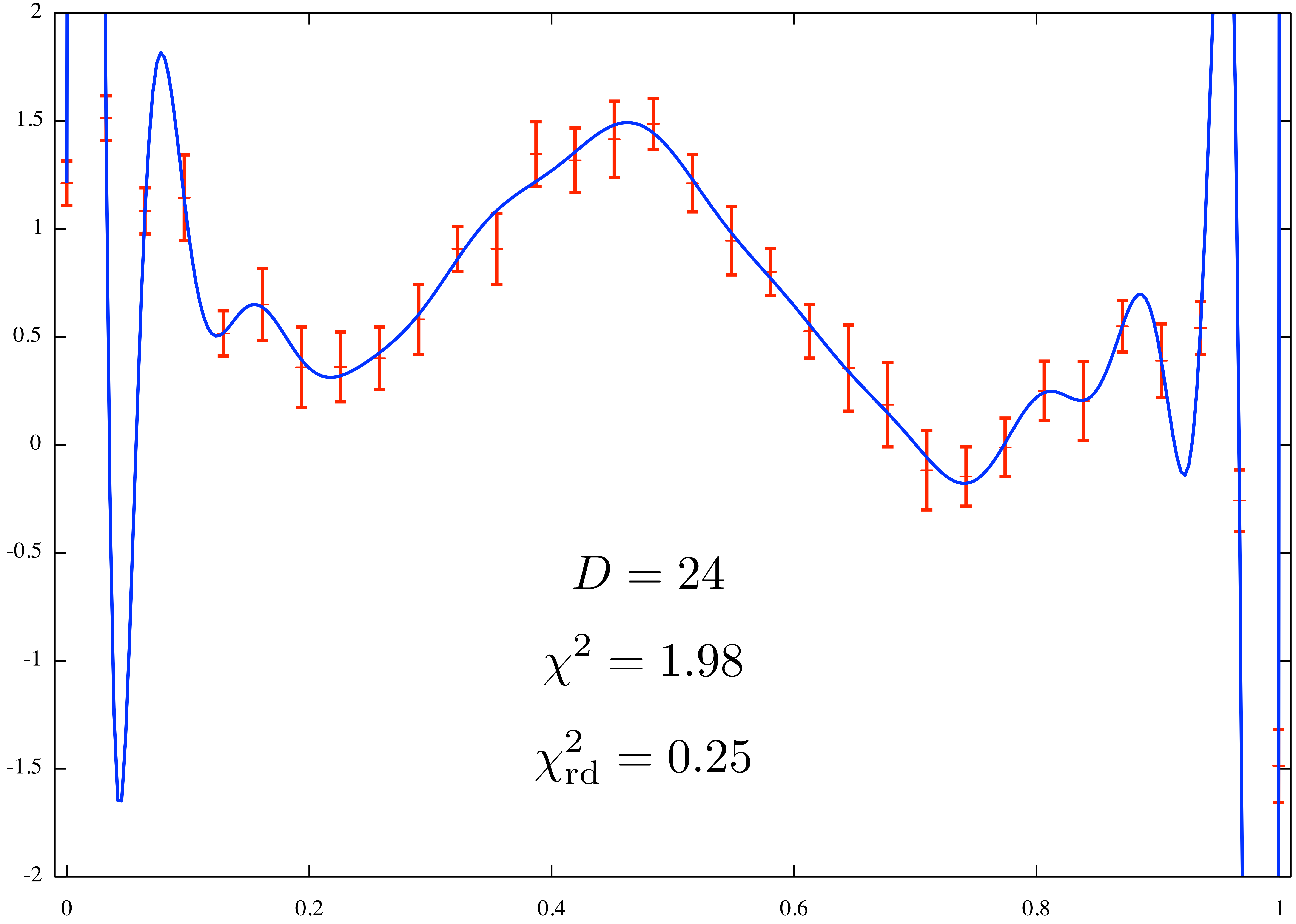
Trans-dimensional modeling



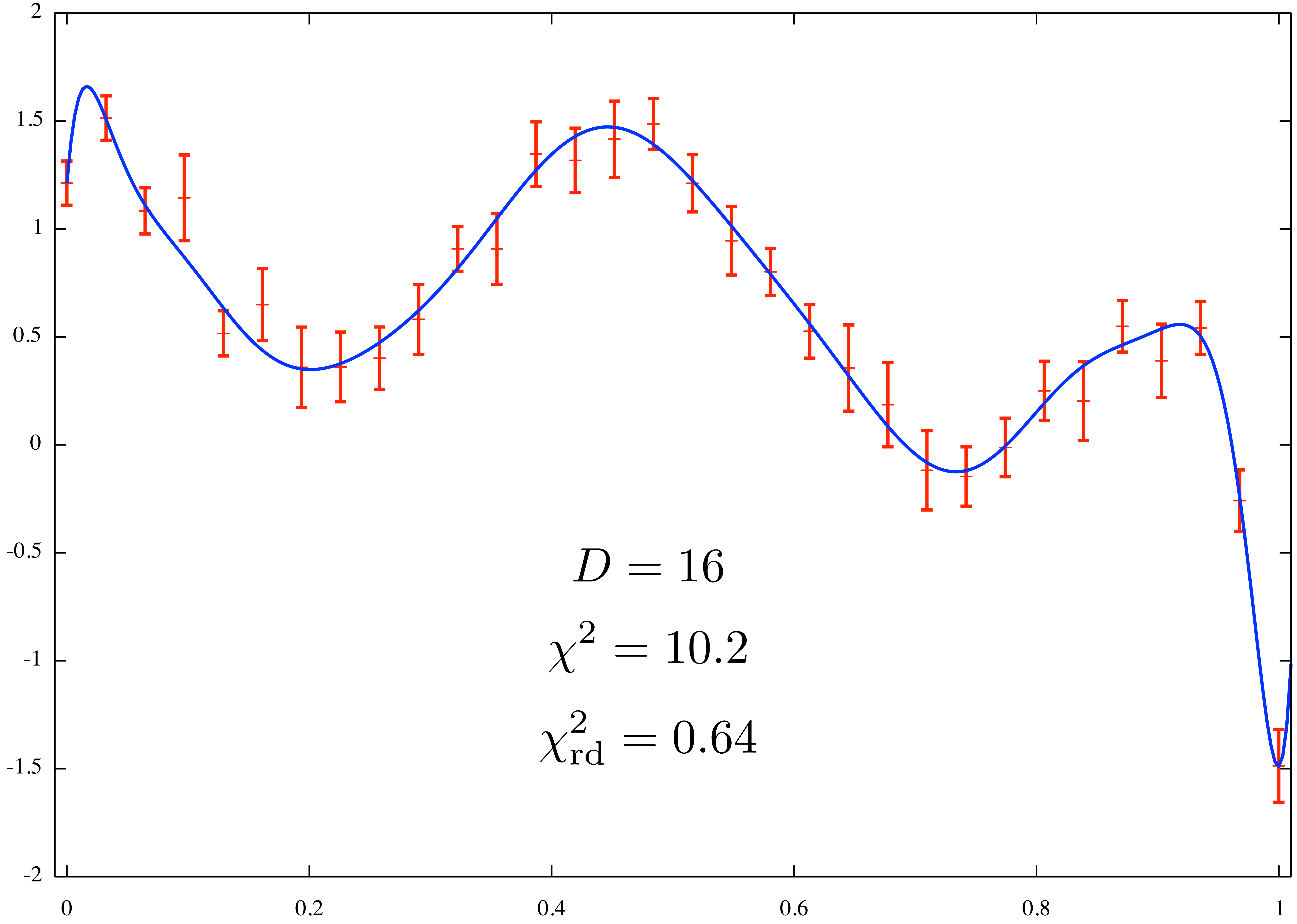
$$\chi^2 = \sum_{i=1}^{N} \frac{(\mathrm{data}_i - \mathrm{model}_i)^2}{\sigma_i^2}$$

$$\chi^2_{\mathrm{rd}} = \frac{\chi^2}{N - D}$$

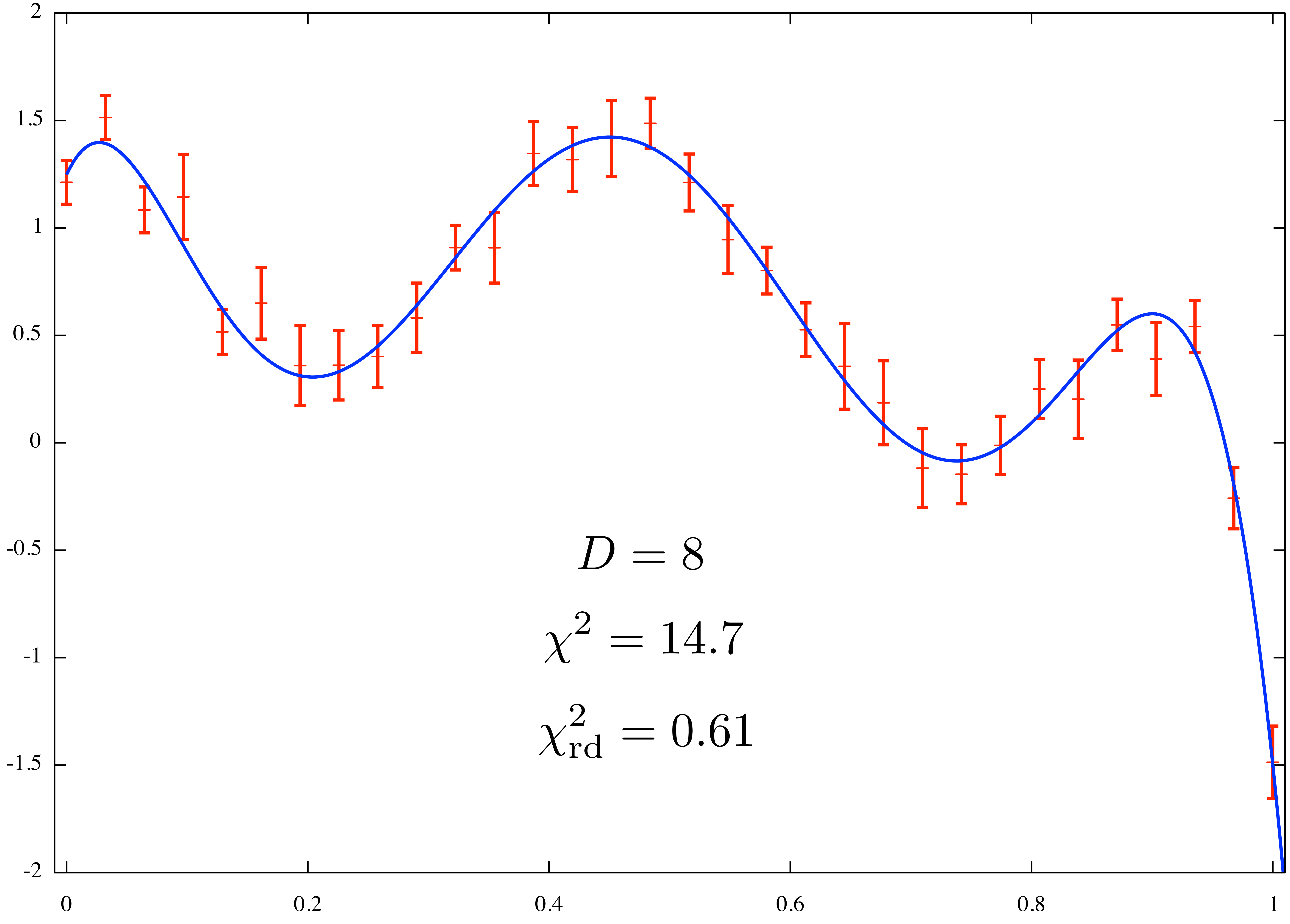

$D = 32$
$\chi^2 = 0$
$\chi^2_{\mathrm{rd}} = ?$
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.2
0.4
0.6
0.8
1



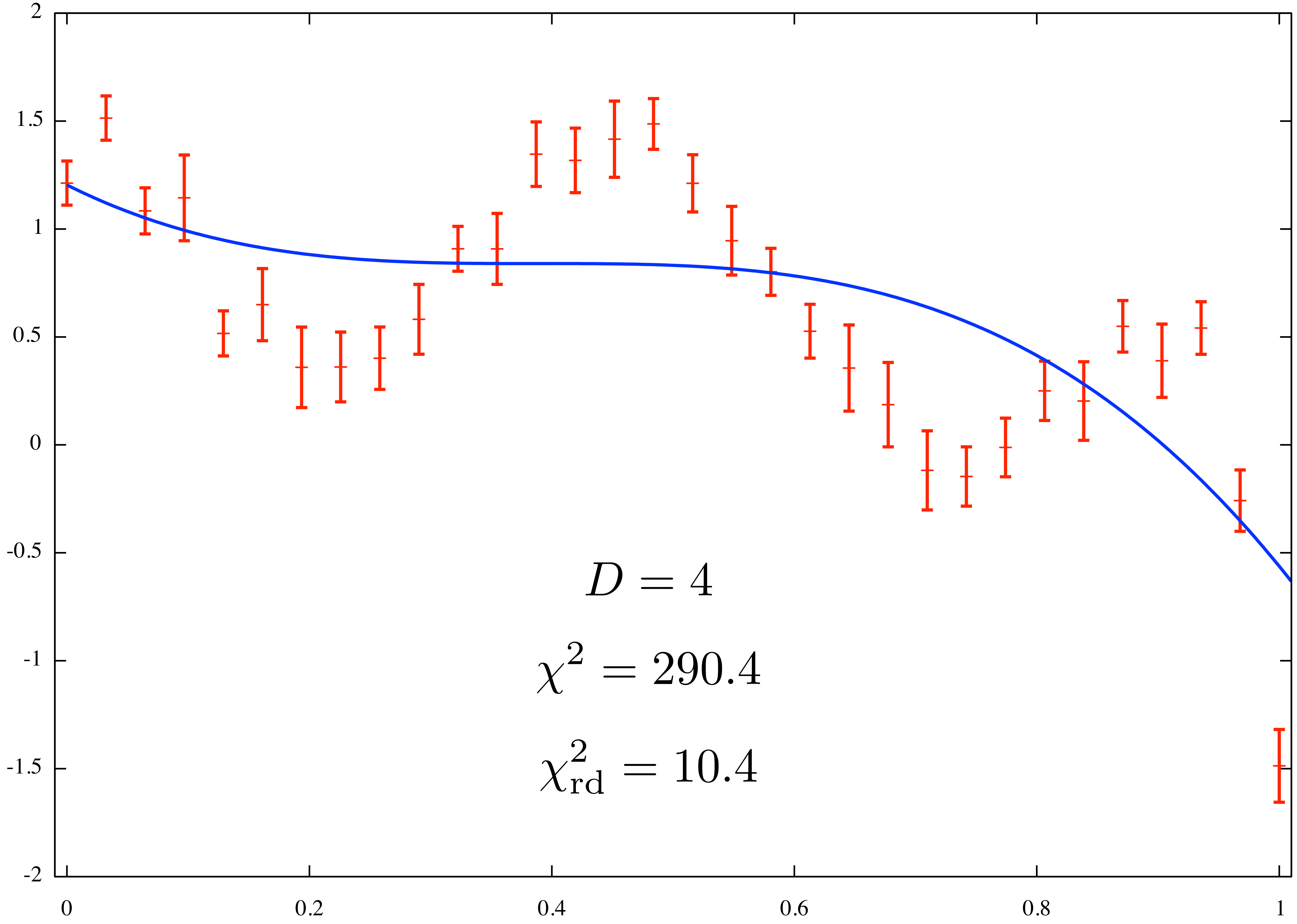
$D = 24$
$\chi^2 = 1.98$
$\chi^2_{\rm rd} = 0.25$
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.2
0.4
0.6
0.8
1



2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.2
0.4
0.6
0.8
1
$D = 16$
$\chi^2 = 10.2$
$\chi^2_{\mathrm{rd}} = 0.64$

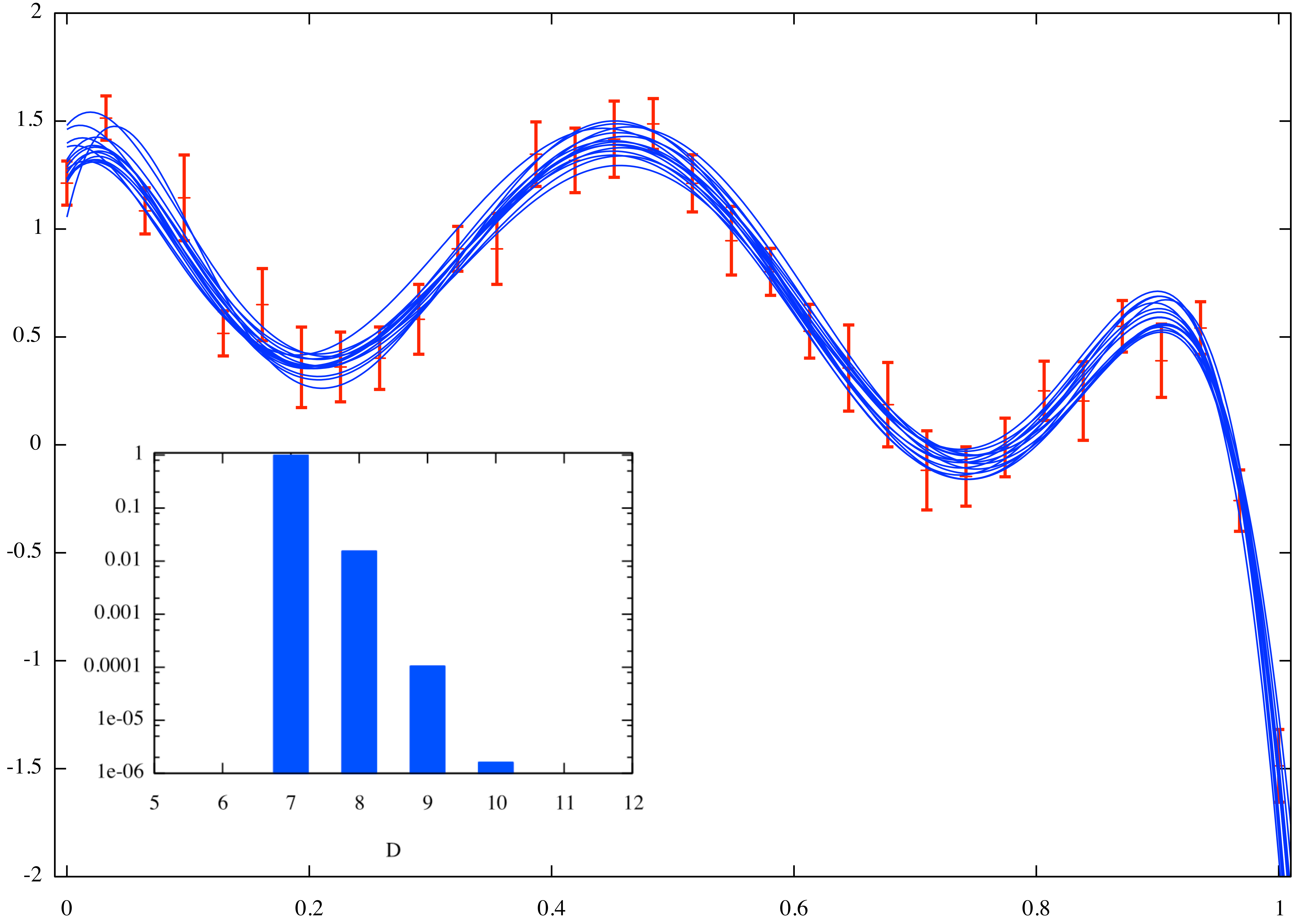


$D = 8$
$\chi^2 = 14.7$
$\chi^2_{\mathrm{rd}} = 0.61$
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.2
0.4
0.6
0.8
1



2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.2
0.4
0.6
0.8
1
$D = 4$
$\chi^2 = 290.4$
$\chi^2_{\mathrm{rd}} = 10.4$

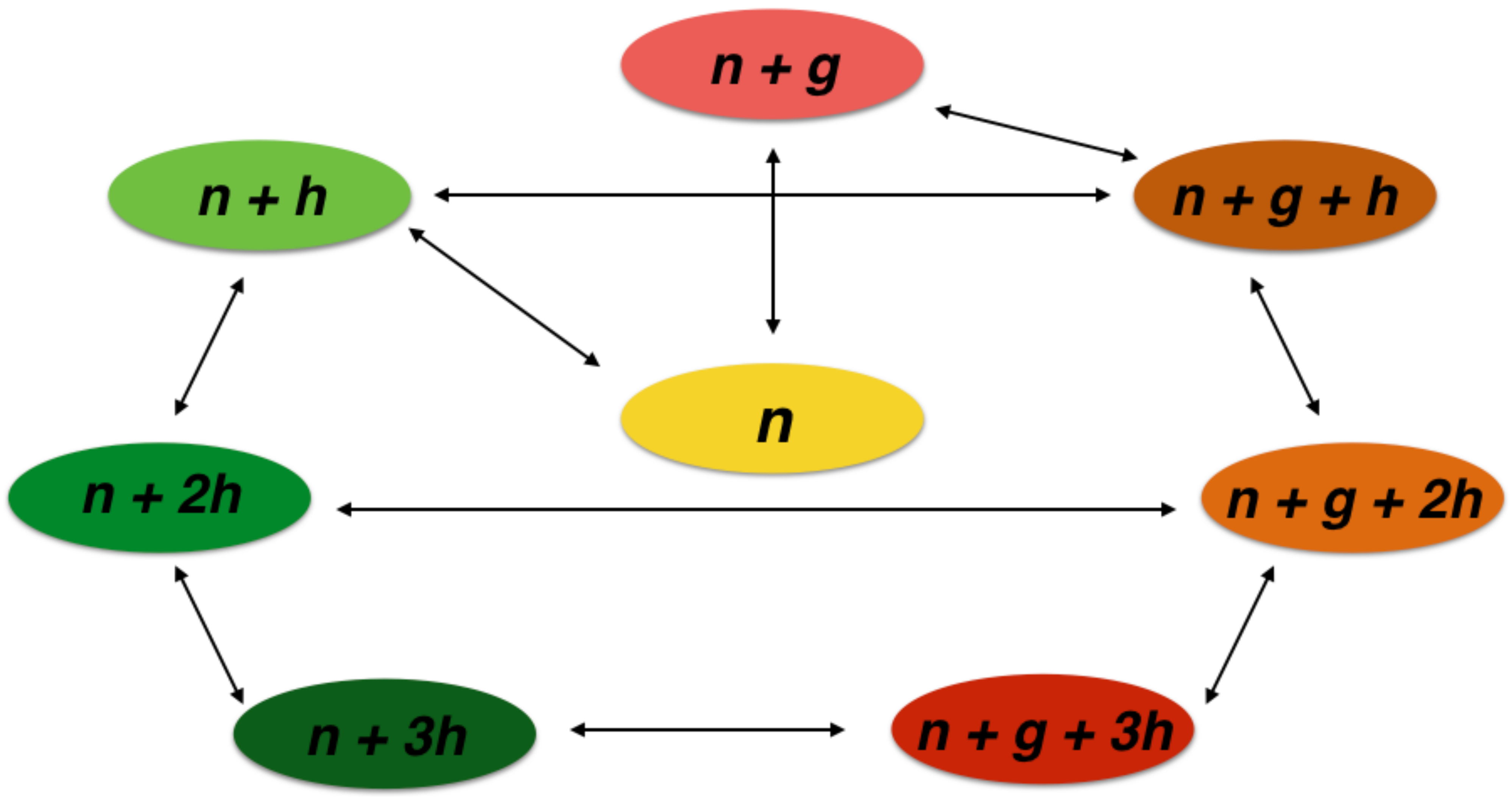
Trans-dimensional Markov Chain Monte Carlo



BayesHopper

Joint search for isolated sources and an unresolved background

➔ Trans-dimensional MCMC



n: Gaussian noise

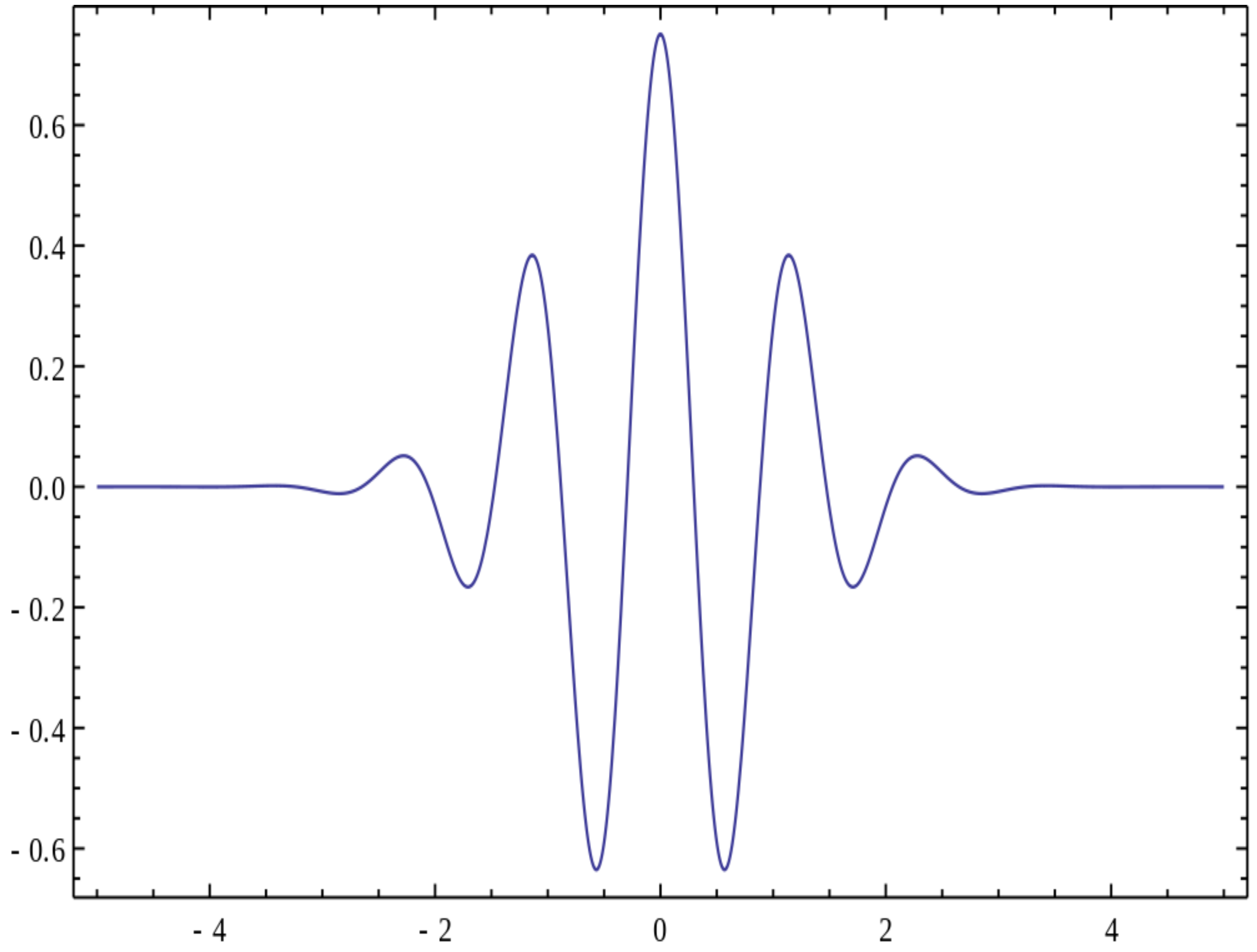
h: individual black hole binary

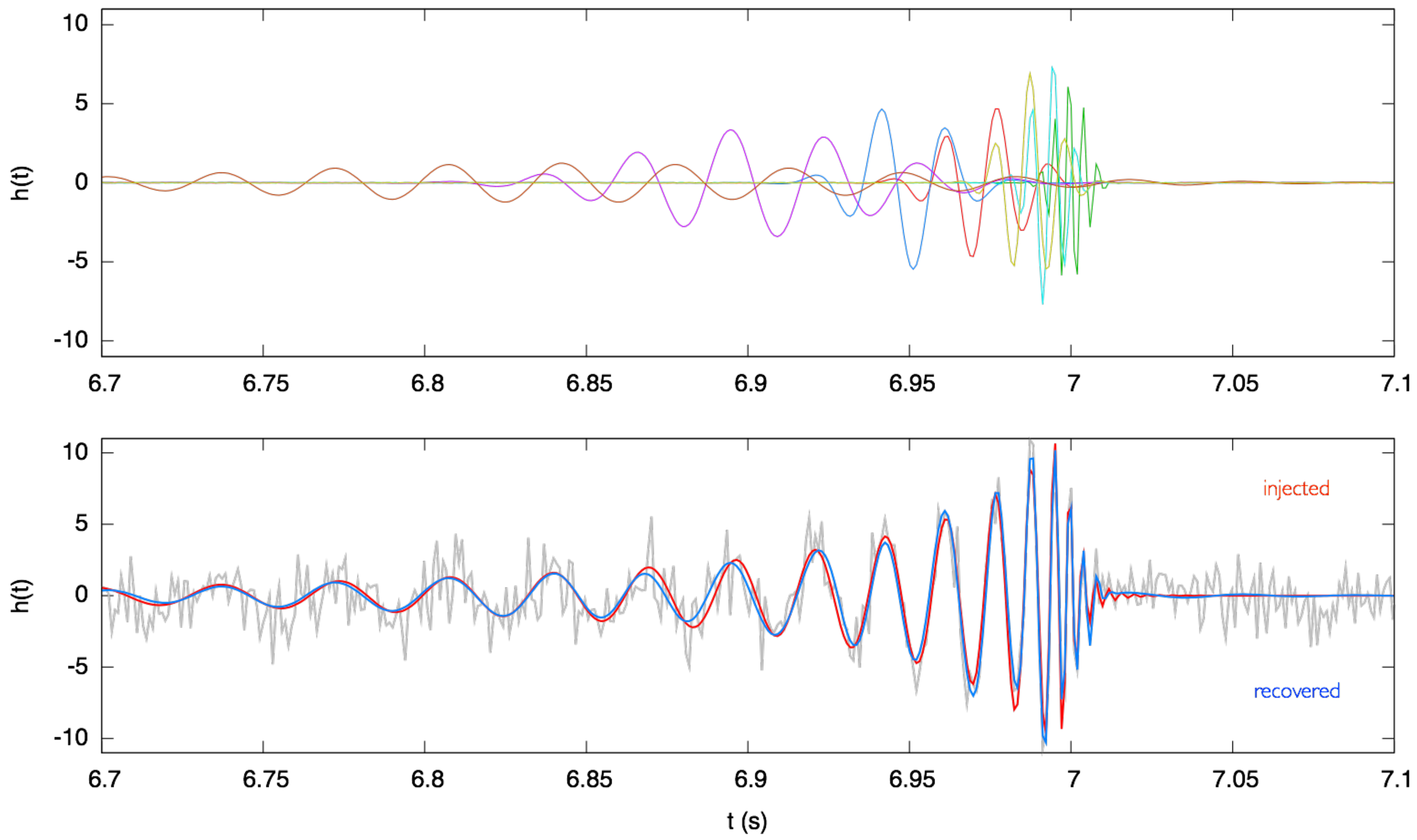
g: stochastic GW background

Paper: Bécsy & Cornish (2019) arXiv:1912.08807

GitHub: https://github.com/bencebecsy/BayesHopper

BayesHopper Burst





We cannot know the number of wavelets needed a priori

➡ Explore models with varying number of parameters using a trans-dimensional MCMC

Earth term only (pulsar terms will be significantly delayed)

Model both GW signals (coherent) and noise transients (incoherent)

➡ competing models are GW and noise transients, not GW and Gaussian noise

Paper: Bécsy & Cornish (2021) arXiv:2011.01942

GitHub: github.com/bencebecsy/BayesHopperBurst

BayesHopper Burst