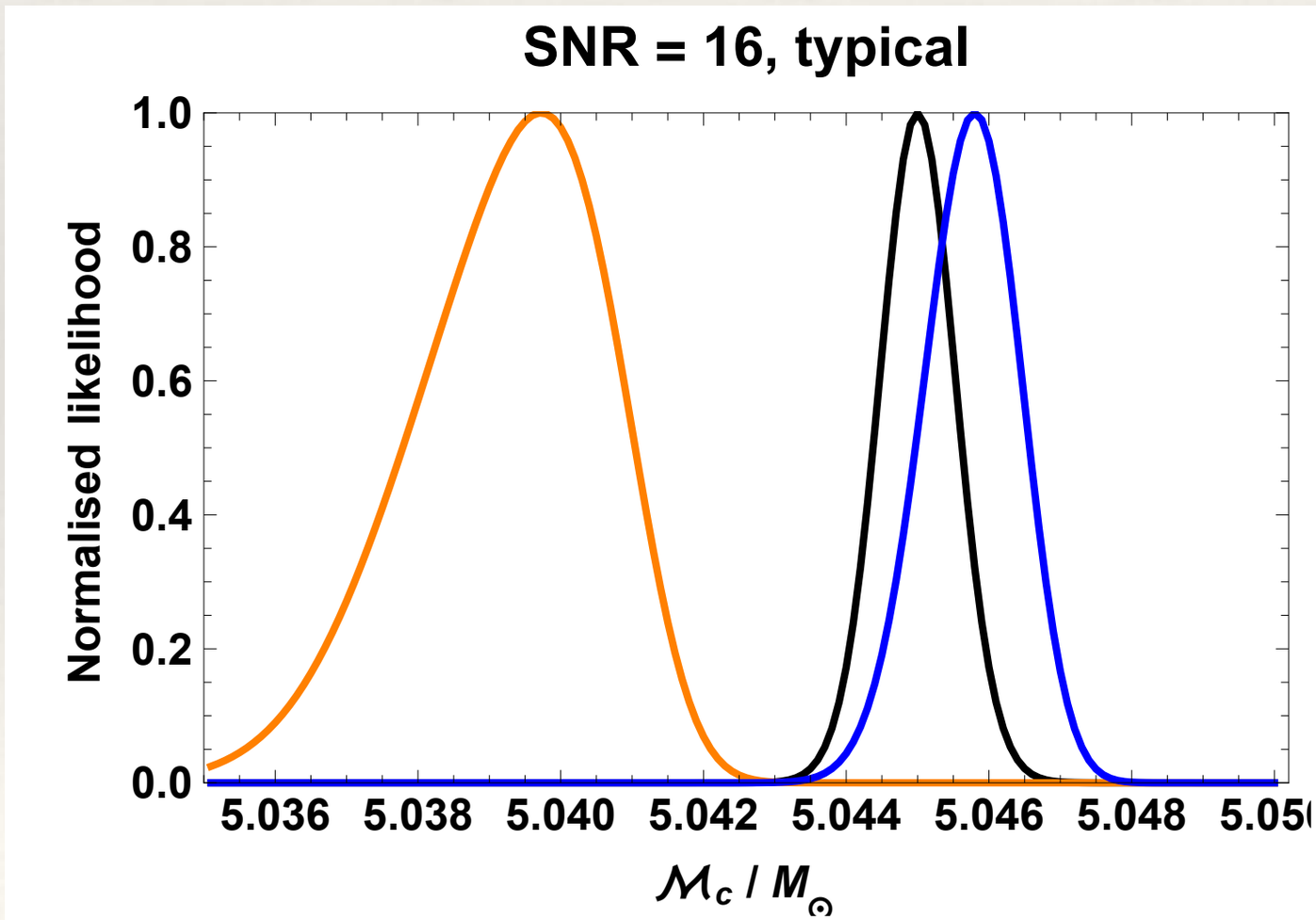
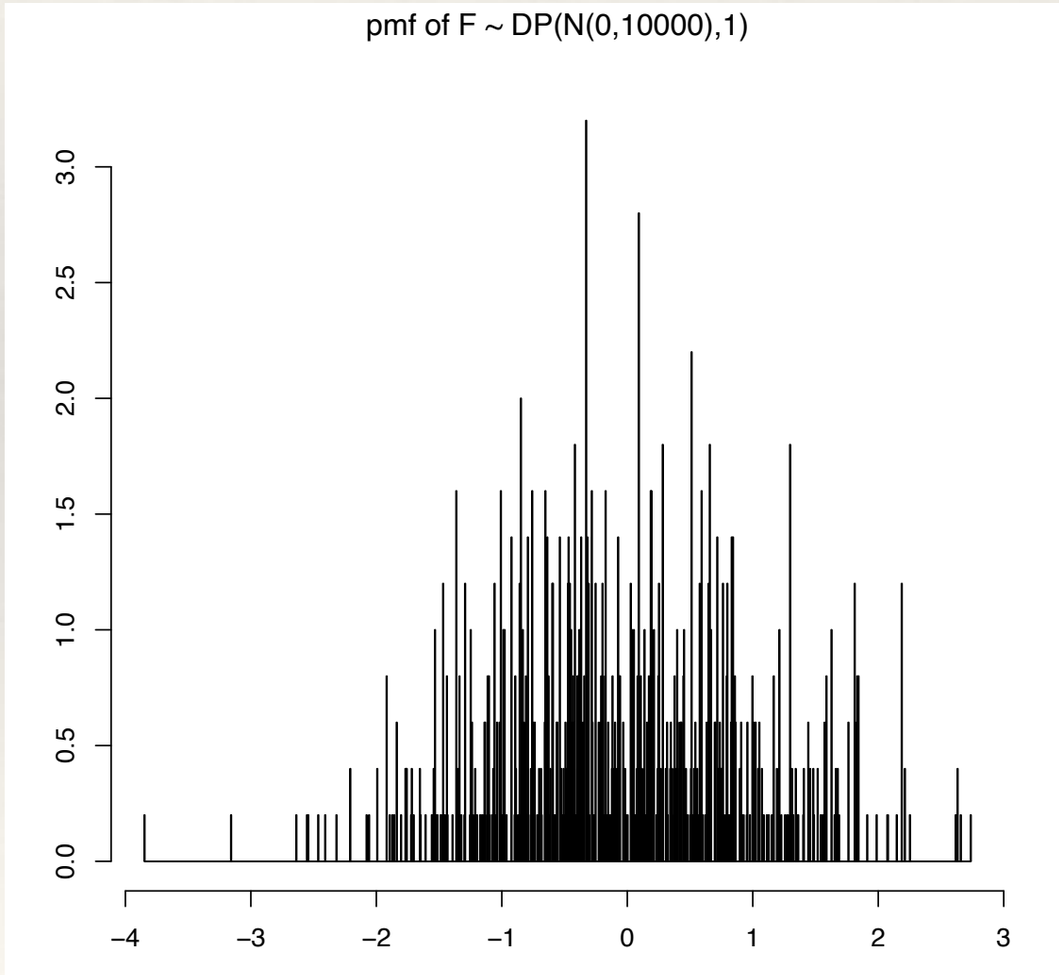
Making sense of data: introduction to statistics for gravitational wave astronomy

Lecture 12: Gaussian and Dirichlet Processes

AEI IMPRS Lecture Course

Jonathan Gair jgair@aei.mpg.de





Stochastic processes

- We encountered stochastic processes when we discussed instrumental noise and again in the team series lectures. Stochastic processes are random procedures that generate a continuous sequence of events.

- Stochastic processes can also be used to generate realisations of formally infinitely-dimensional probability distributions.

- We will discuss two particular types of stochastic process.

    - **Gaussian processes** are infinite dimensional generalisations of Gaussian distributions, a realisation of which is a correlated random field.

    - **Dirichlet processes** are infinite dimensional generalisations of the Dirichlet distribution, a realisation of which is a probability distribution.

What are Gaussian processes?

- A Gaussian process (GP) is an infinite dimensional generalisation of a multi-variate Gaussian distribution.

- Any finite subset of samples are distributed as a multi-variate Gaussian.

- For example, if $y(\boldsymbol{x}) \sim GP(m(\boldsymbol{x}), k(\boldsymbol{x}_i, \boldsymbol{x}_j))$, then the values $\{y_i = y(\boldsymbol{x}_i)\}$ at points $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ follow

$$p(\mathbf{y}) \propto \exp\left[-\frac{1}{2}\sum_{ij}(x_i - m_i)\,K_{ij}^{-1}\,(x_j - m_j)\right]$$

- Here $m_i = m(\boldsymbol{x}_i)$ and $\boldsymbol{K}_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. The functions $m(\boldsymbol{x})$ and $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ are the mean and covariance function of the GP.

Gaussian process regression

- Can use GPs for regression. Represent the true values of the unknown function as a (often zero mean) GP.

$$y(\mathbf{x}) \sim \mathrm{GP}\left(m(\mathbf{x}), k(\mathbf{x}_i, \mathbf{x}_j)\right)$$

- Assume we have a *training set* of points where the function is known (possibly with error).

$$\left\{d_i = y(\mathbf{x}_i) + \epsilon_i, \qquad i = 1, \ldots, n, \qquad \epsilon_i \sim N\left(0, \sigma_i^2\right)\right\}$$

- The value of the function at a new point or set of points is found by conditioning on the observed values.

Gaussian process regression

- At a new set of points, $\{\mathbf{z_i}\}$, we denote the vector of new values by $\mathbf{y}$ with $y_i=y(\mathbf{z_i})$. We have

$$p(\mathbf{y}) \propto \exp\left[-\frac{1}{2}\left(\mathbf{y}-\mu\right)^T \Sigma^{-1}\left(\mathbf{y}-\mu\right)\right]$$

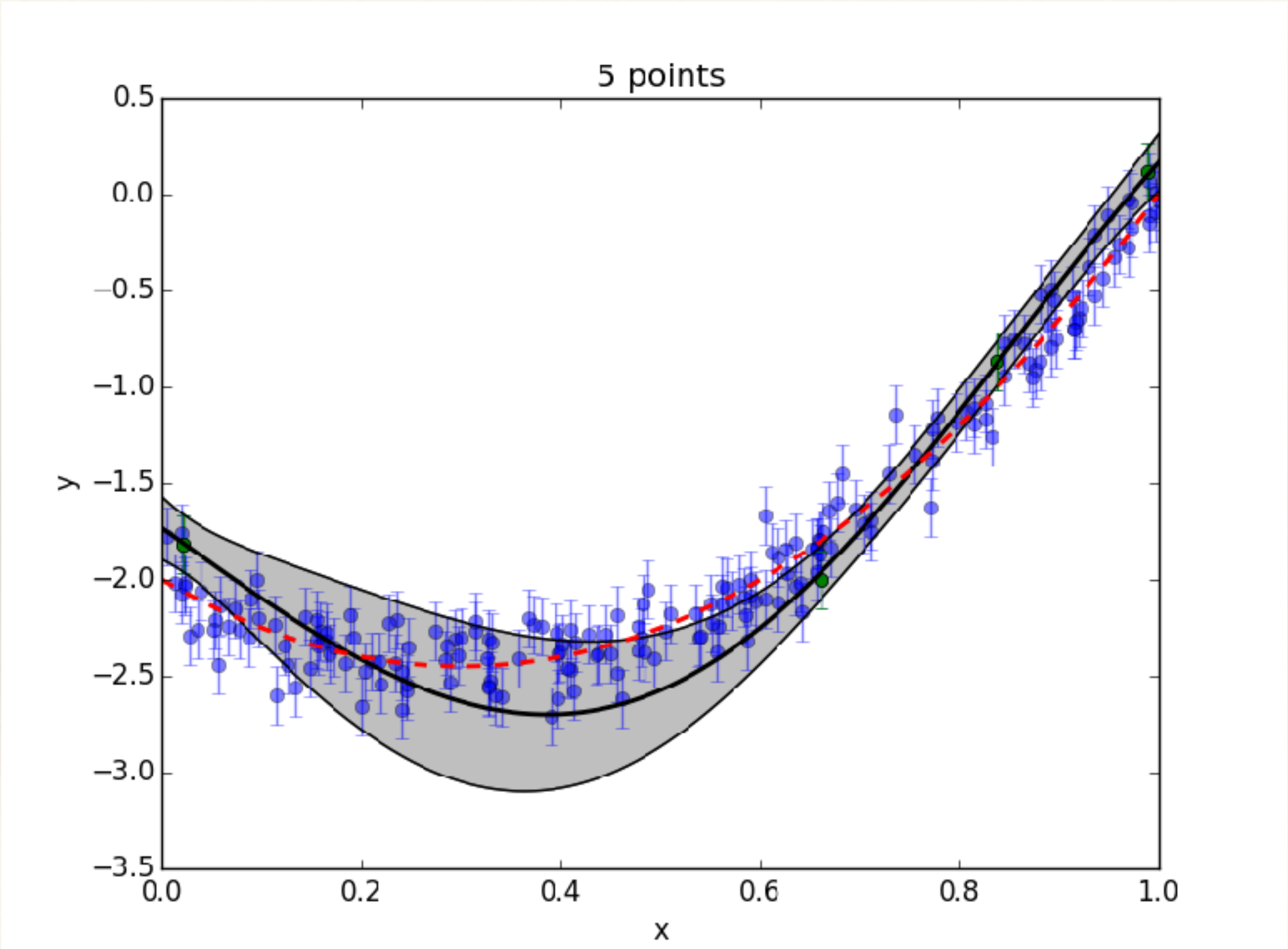
- where the mean and covariance matrix of this Gaussian distribution are given by

$$\mu_i = m(\mathbf{z_i}) + \sum_{jk} k(\mathbf{z_i}, \mathbf{x_j}) \left[\mathbf{K}^{-1}\right]_{jk} (d_k - m(\mathbf{x_k}))$$

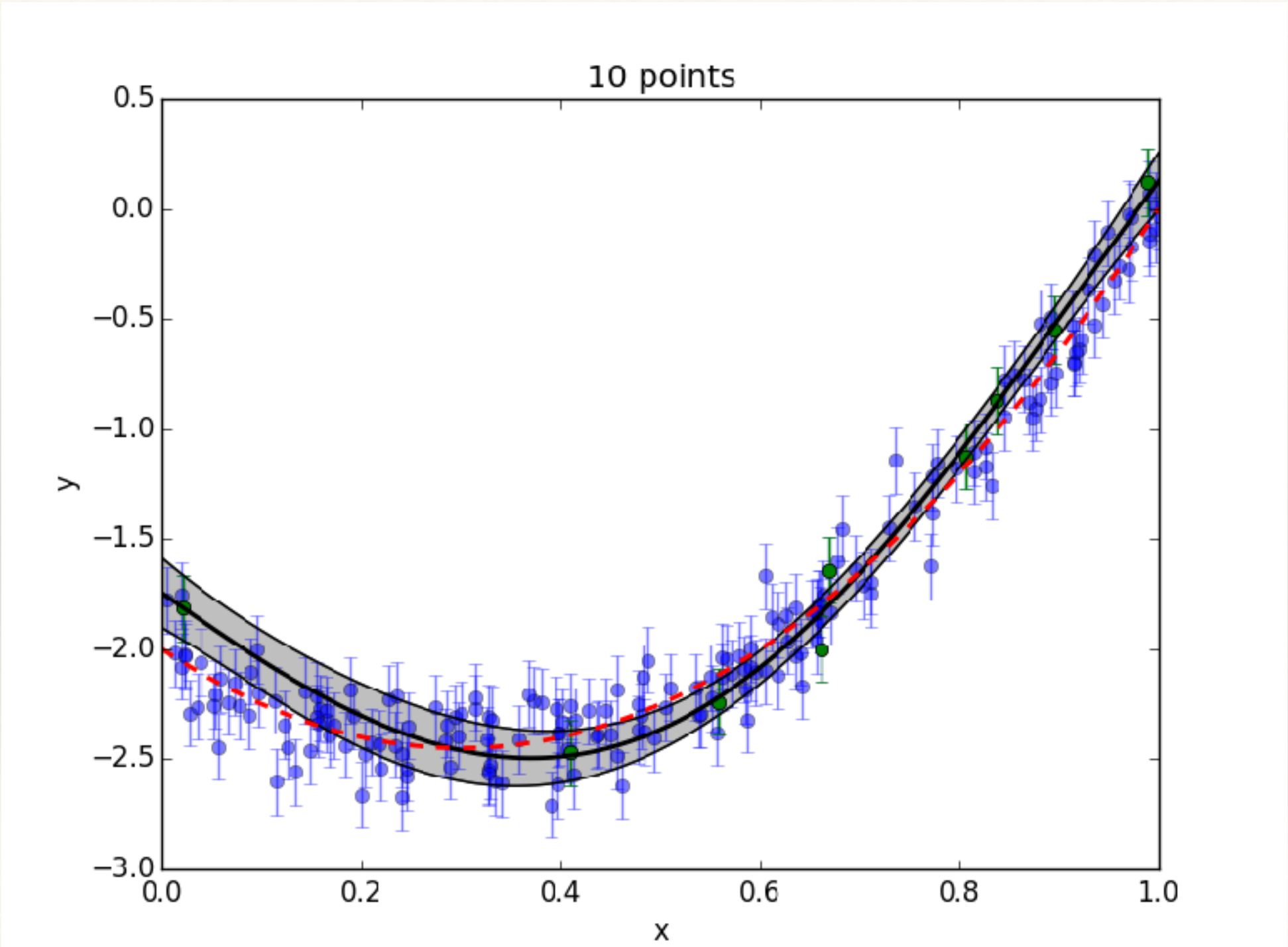
$$\Sigma_{ij} = k(\mathbf{z_i}, \mathbf{z_j}) - \sum_{kl} k(\mathbf{z_i}, \mathbf{x_k}) \left[\mathbf{K}^{-1}\right]_{kl} k(\mathbf{x_l}, \mathbf{z_j})$$

$$\mathbf{K}_{ij} = k(\mathbf{x_i}, \mathbf{x_j}) + \sigma_i^2 \delta_{ij}$$

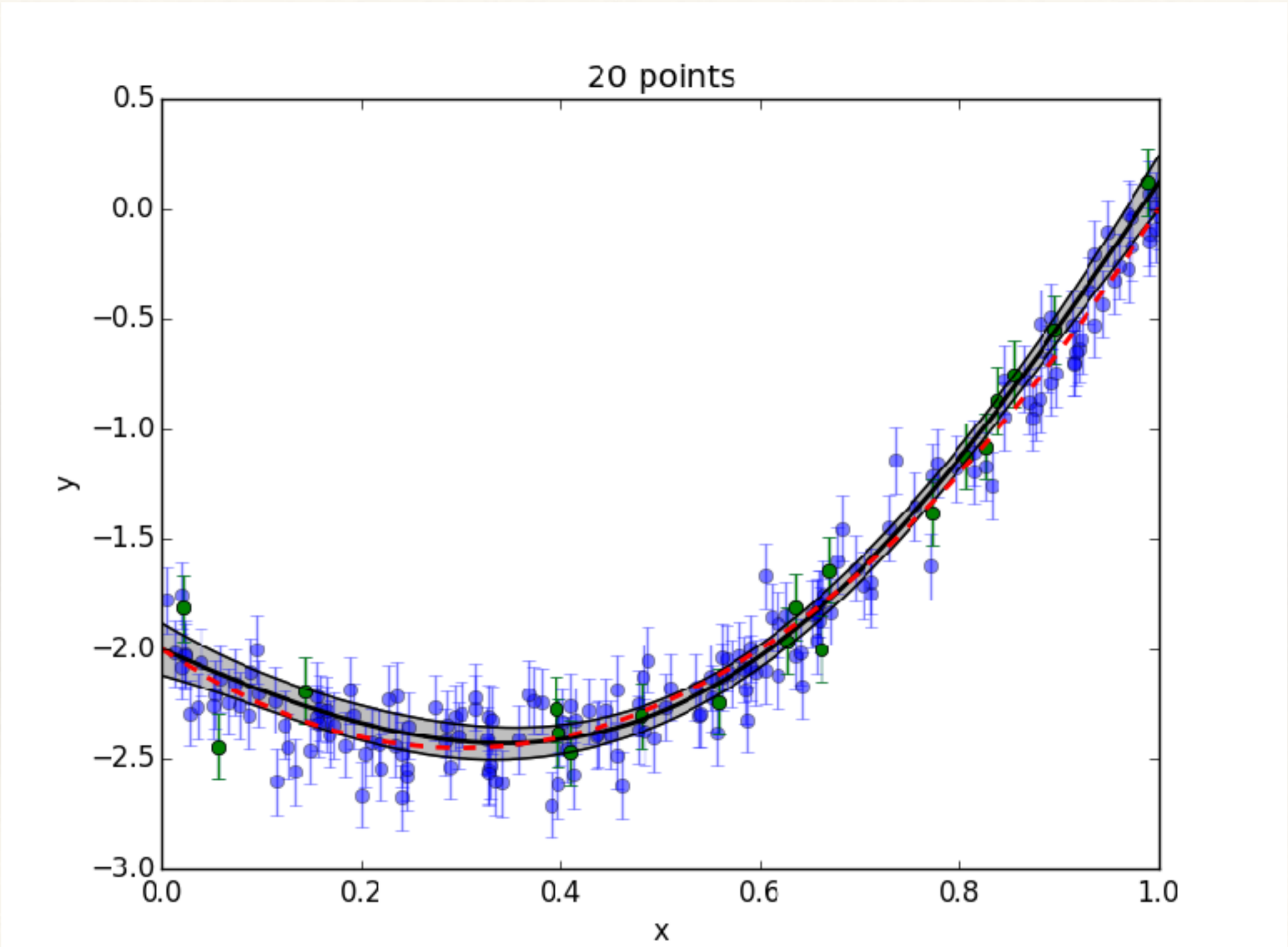
GPR example - quadratic regression



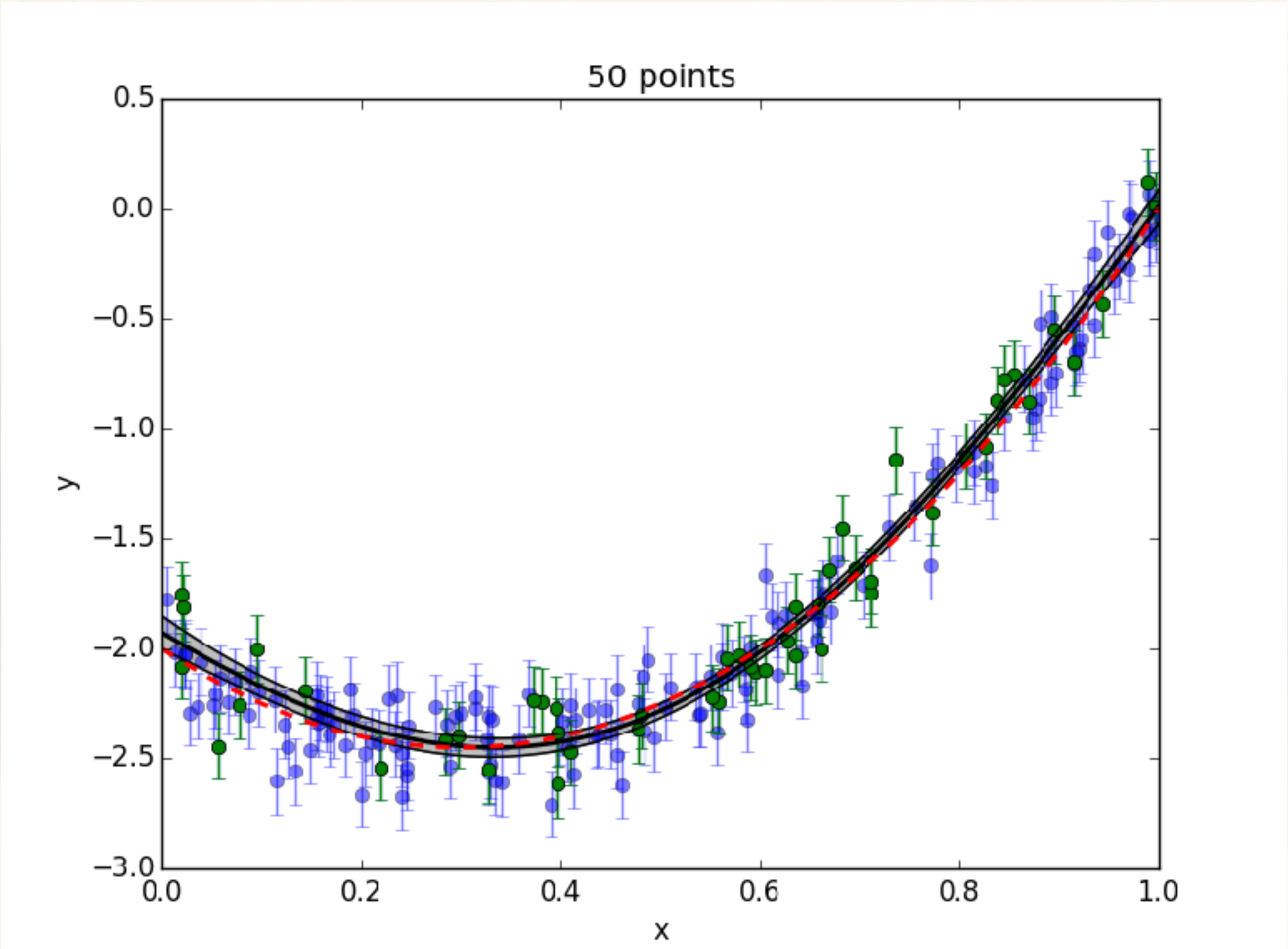
GPR example - quadratic regression



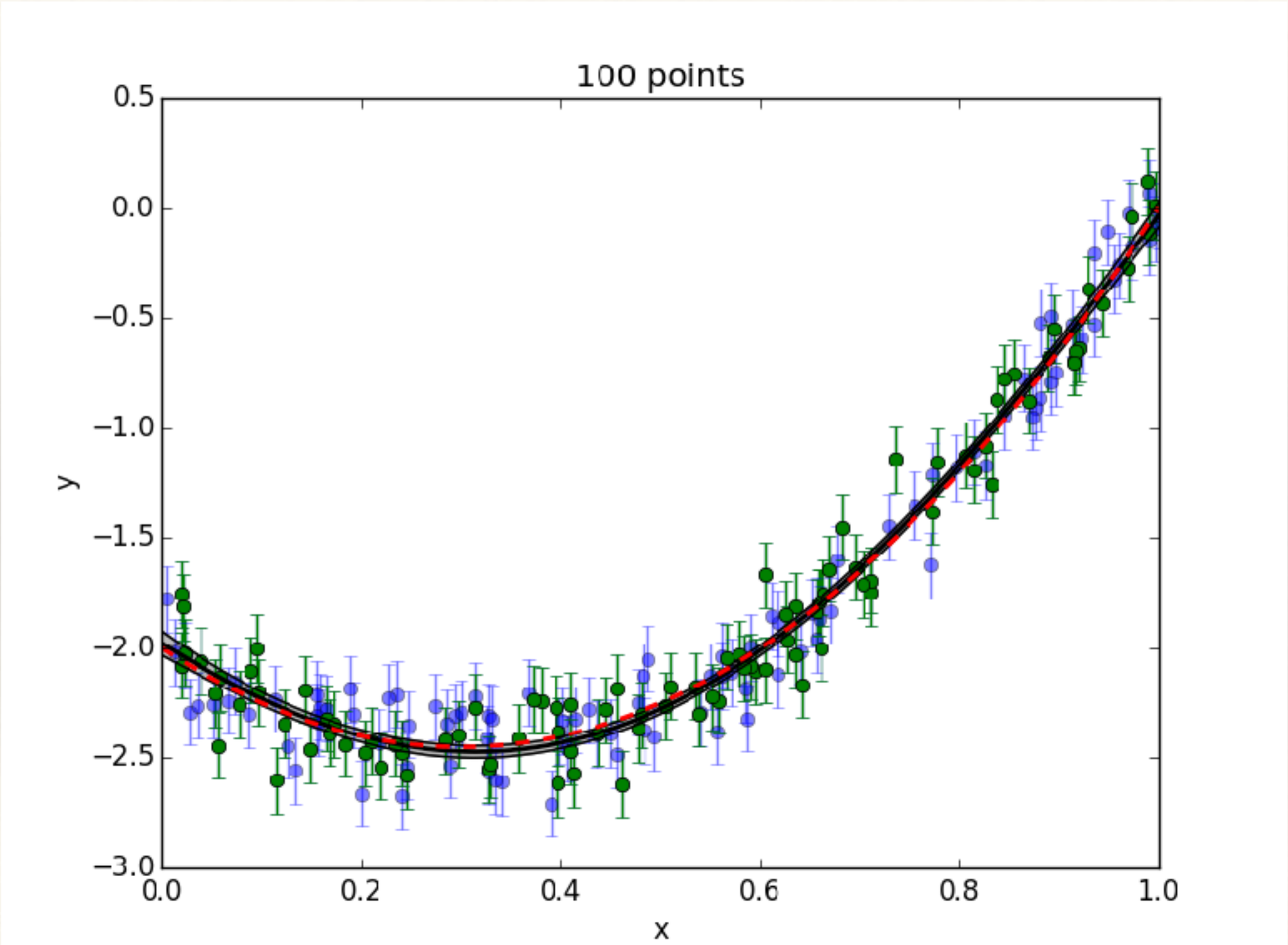
GPR example - quadratic regression



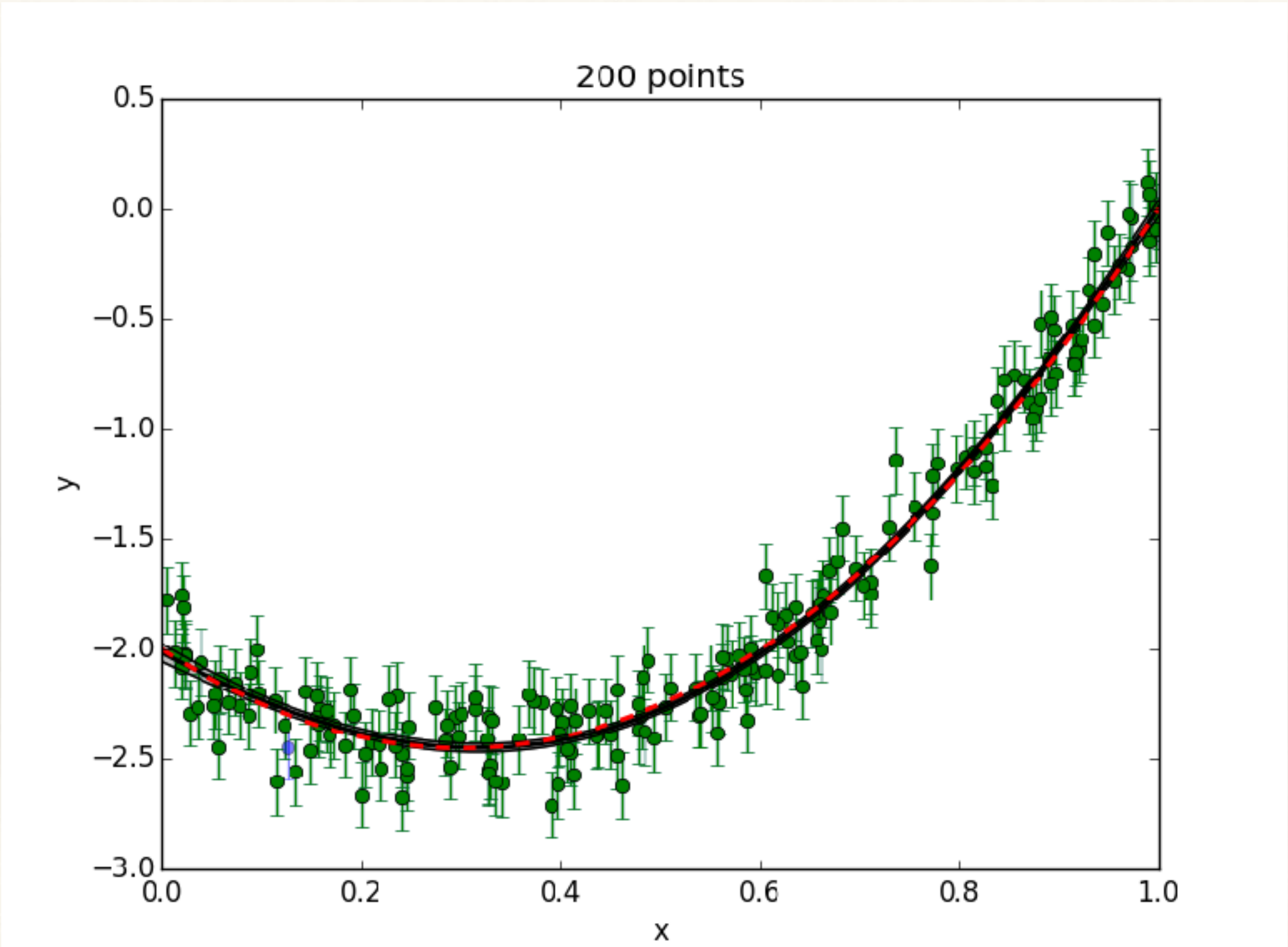
GPR example - quadratic regression



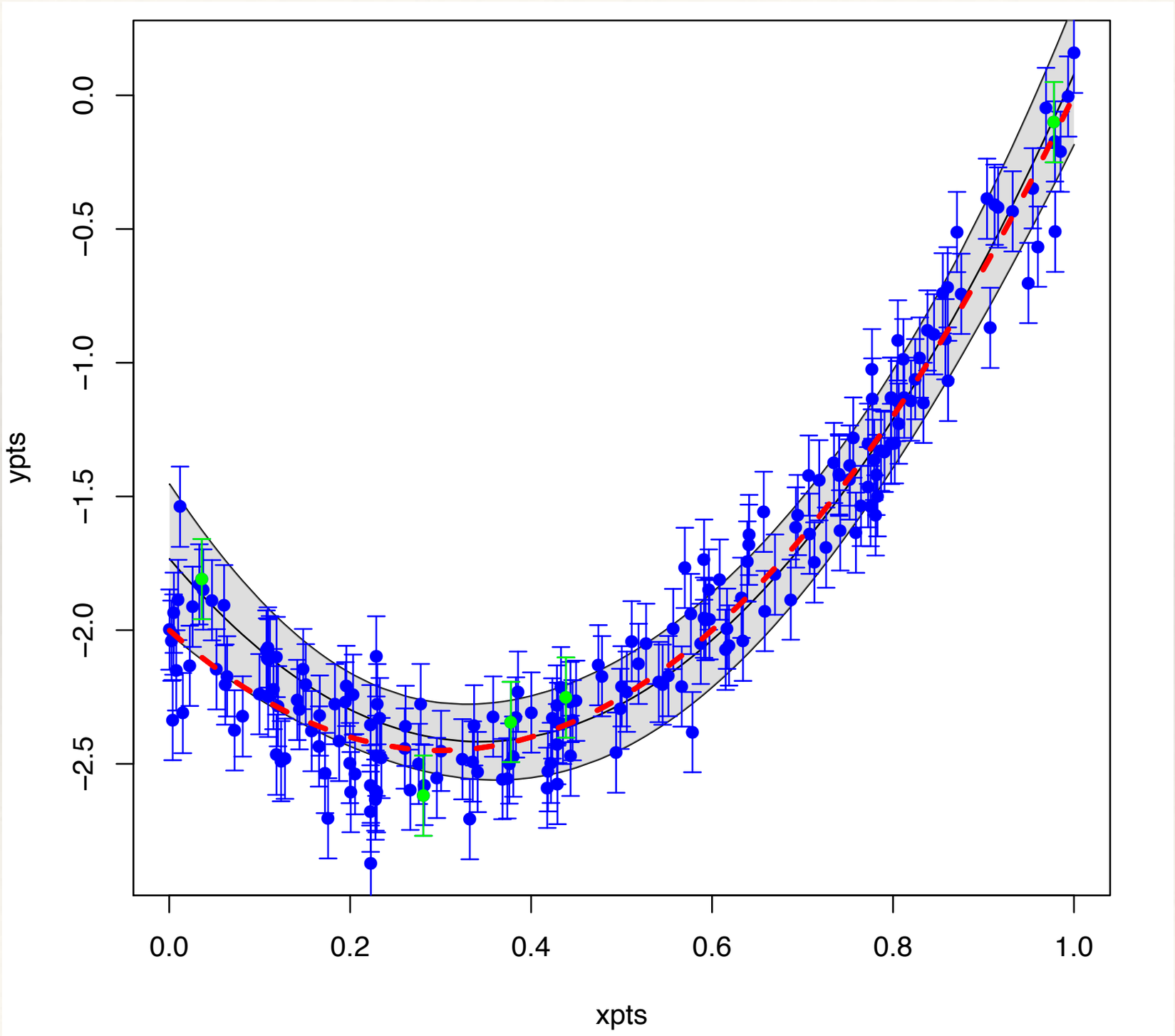
GPR example - quadratic regression



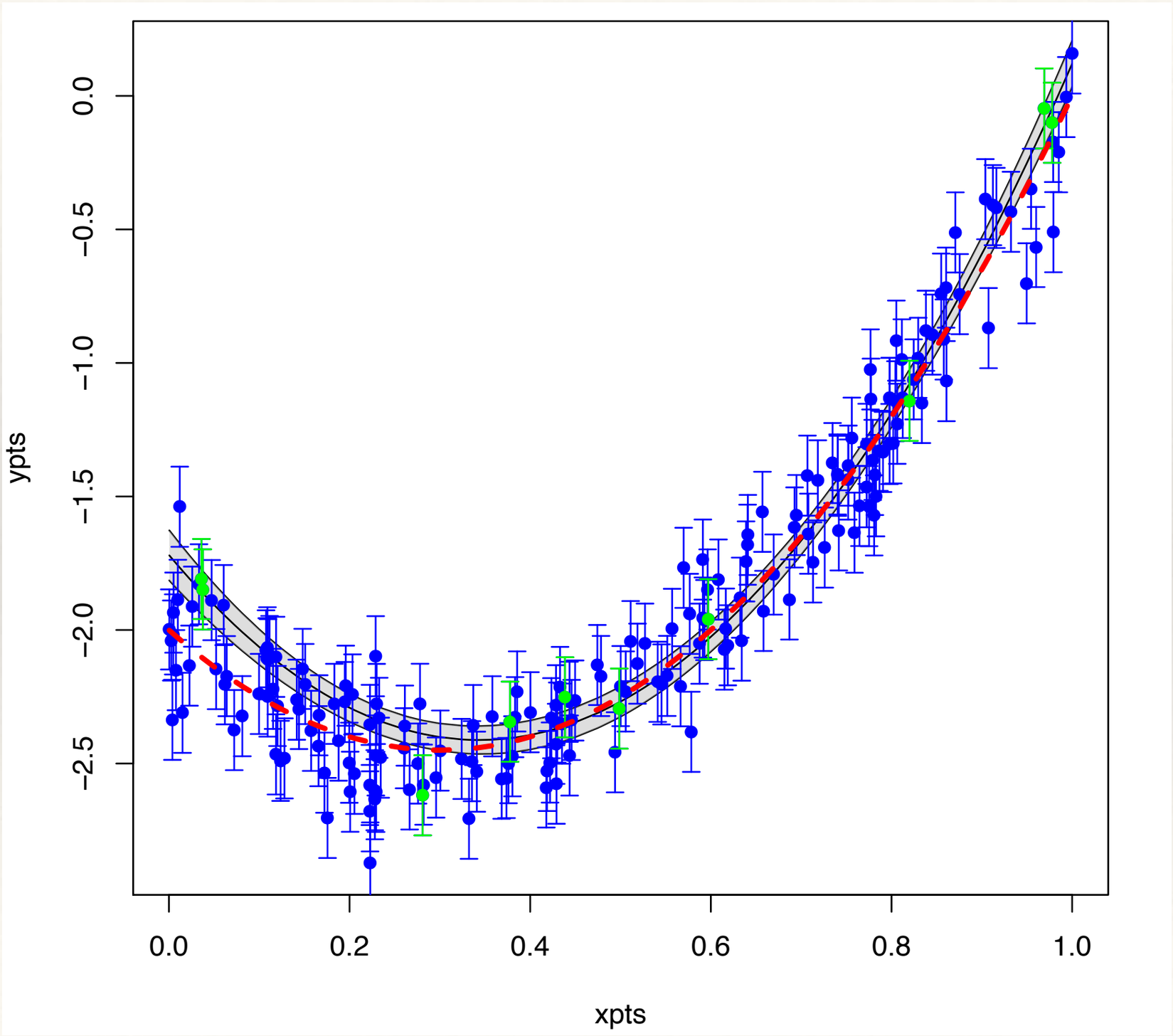
GPR example - quadratic regression



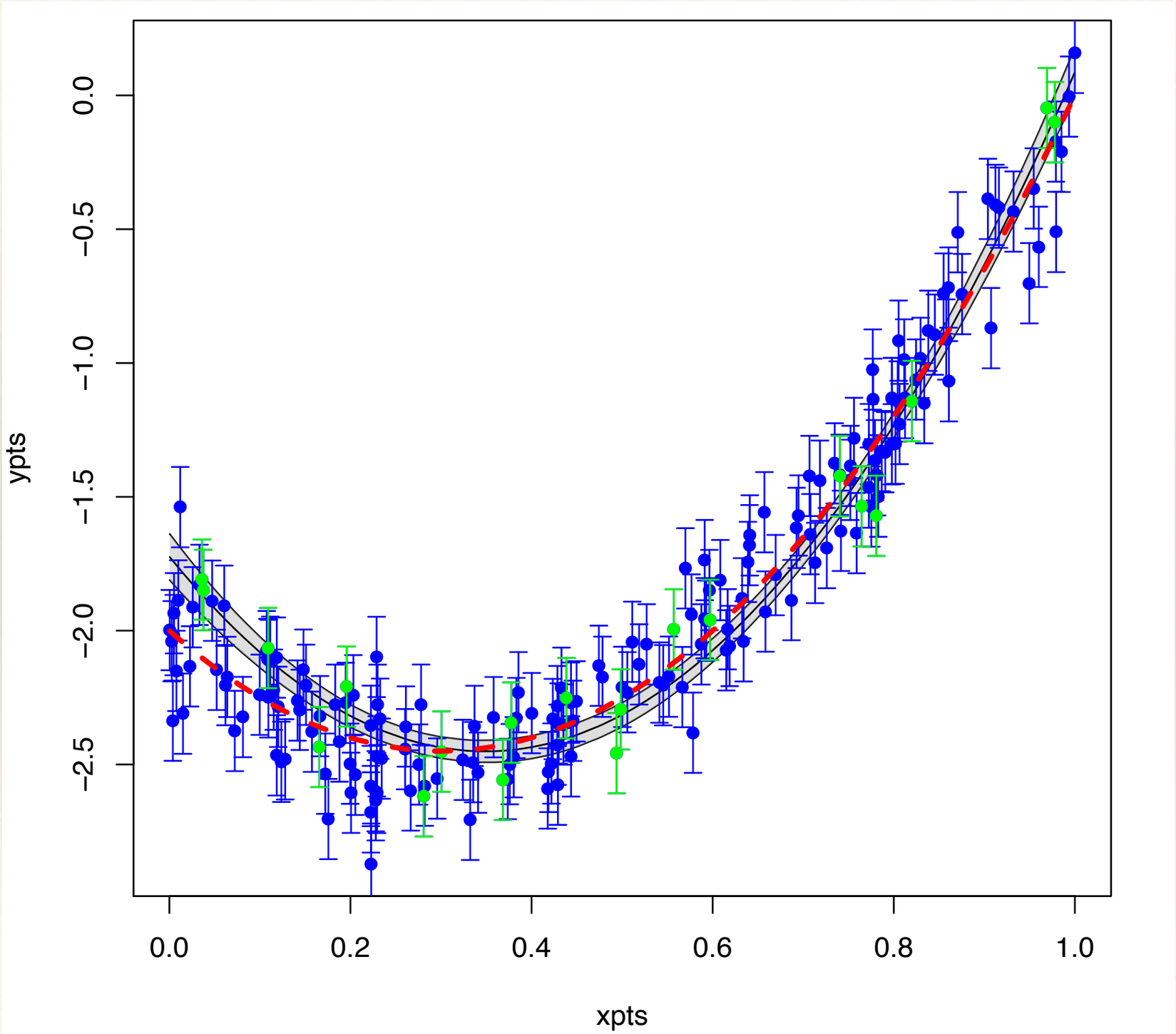
Comparison - quadratic regression



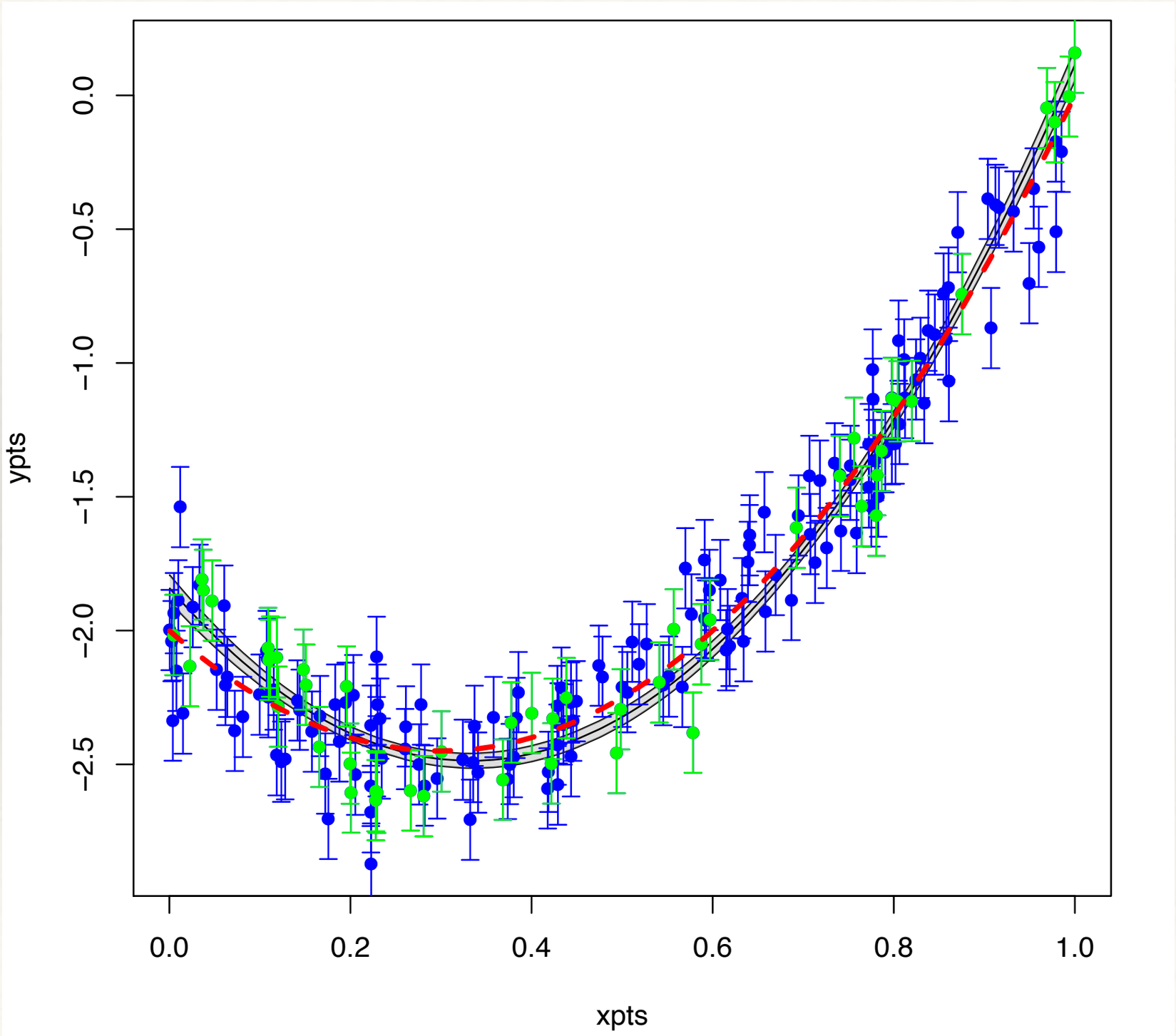
Comparison - quadratic regression



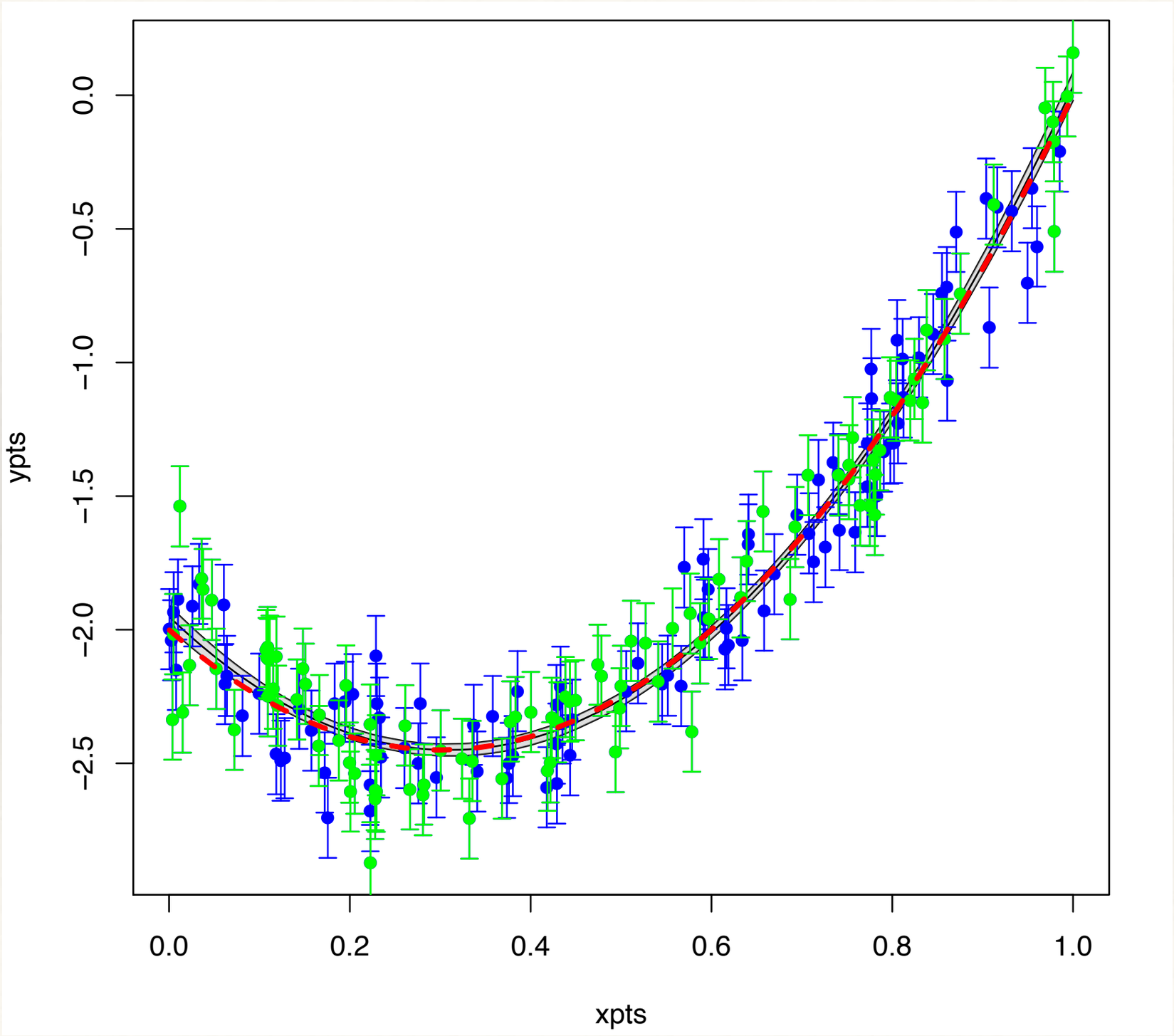
Comparison - quadratic regression



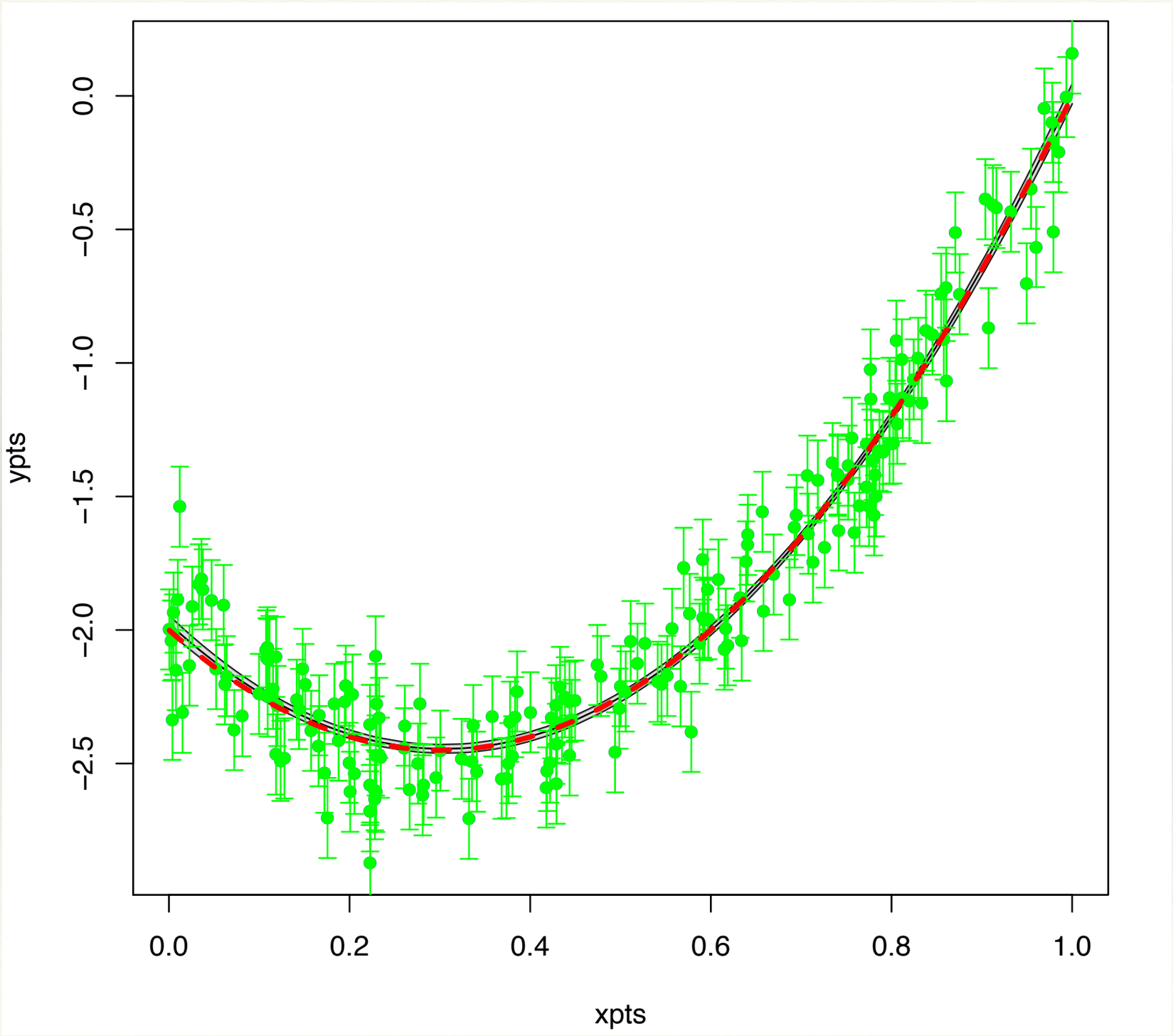
Comparison - quadratic regression



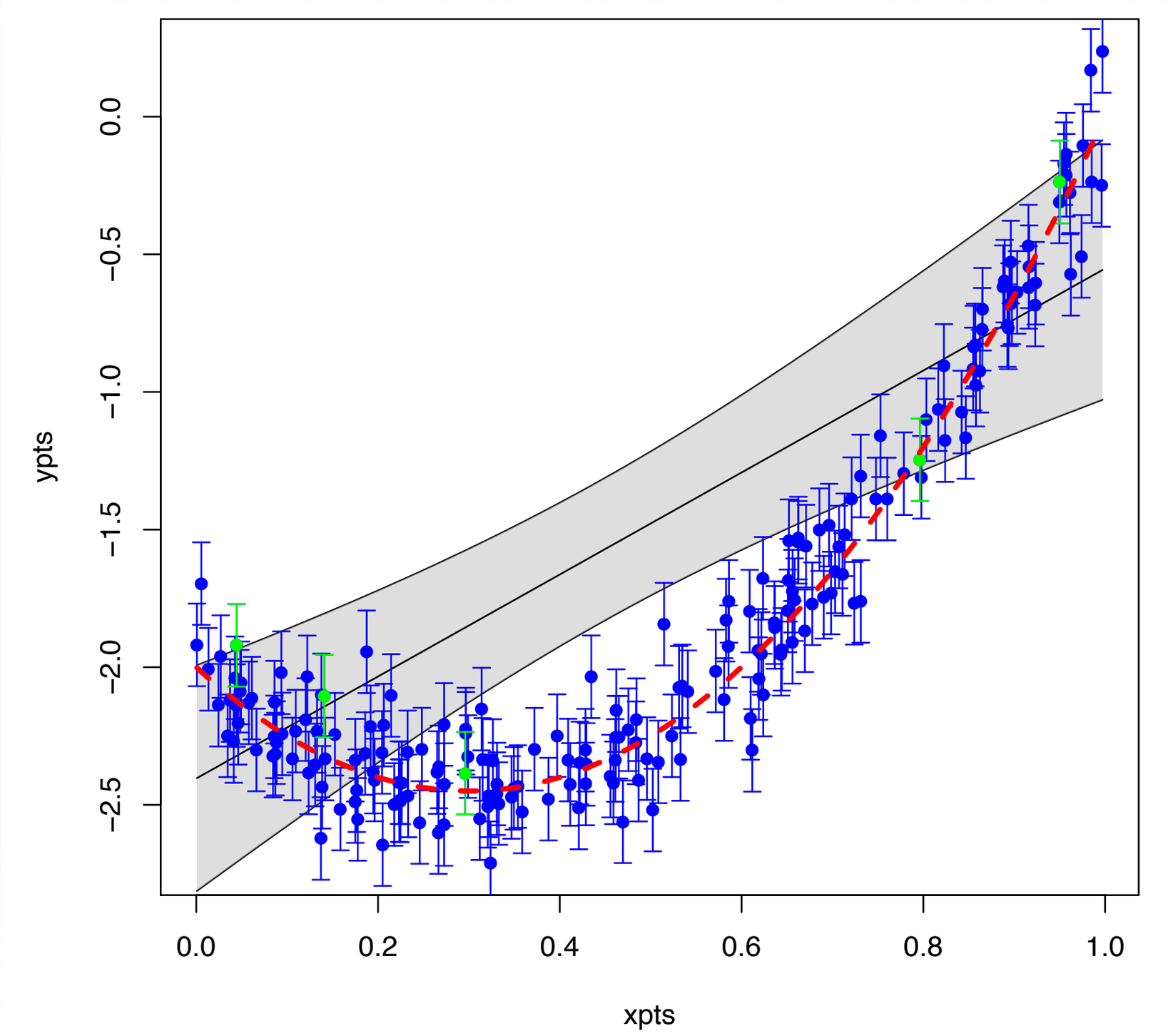
Comparison - quadratic regression



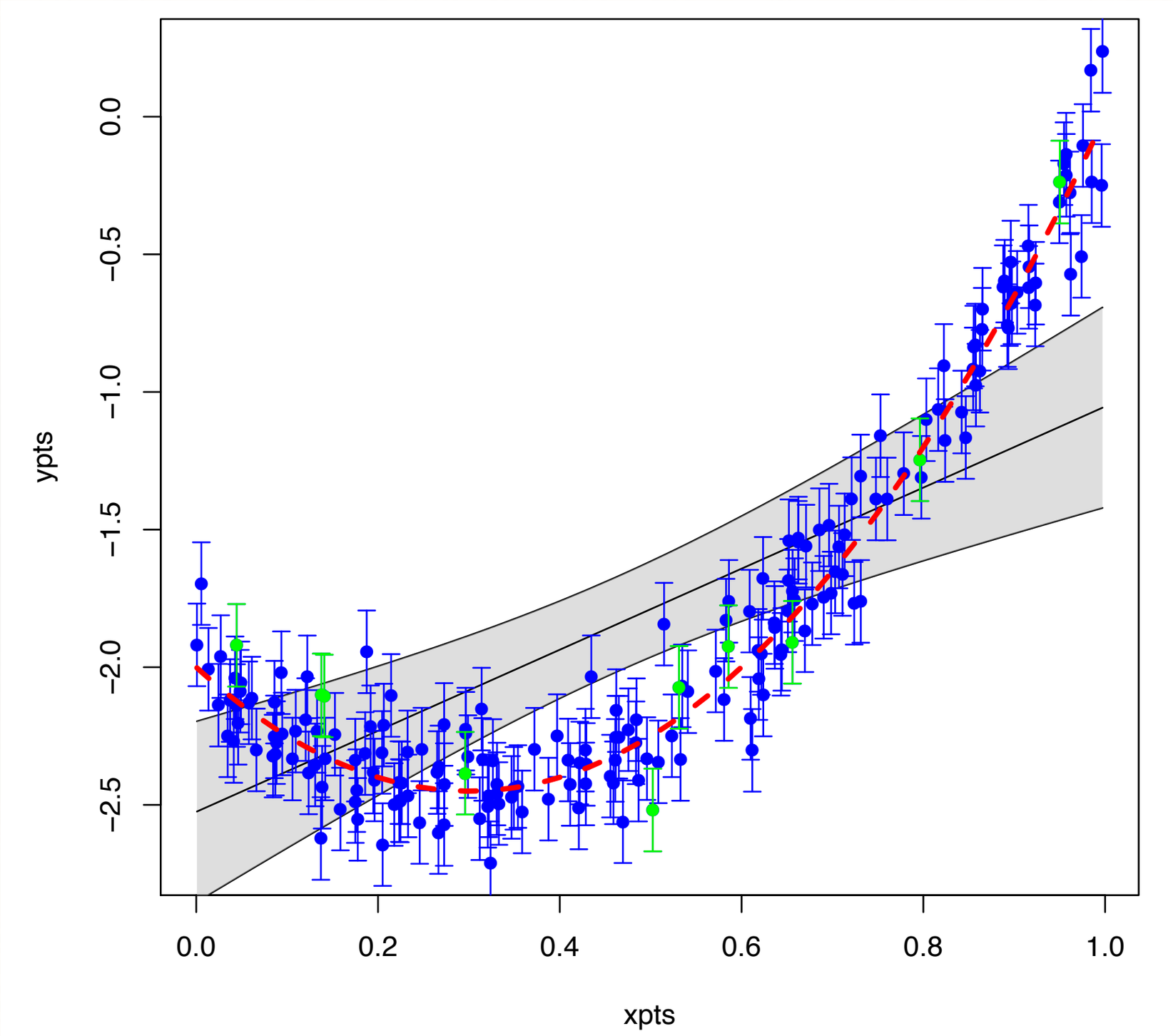
Comparison - quadratic regression



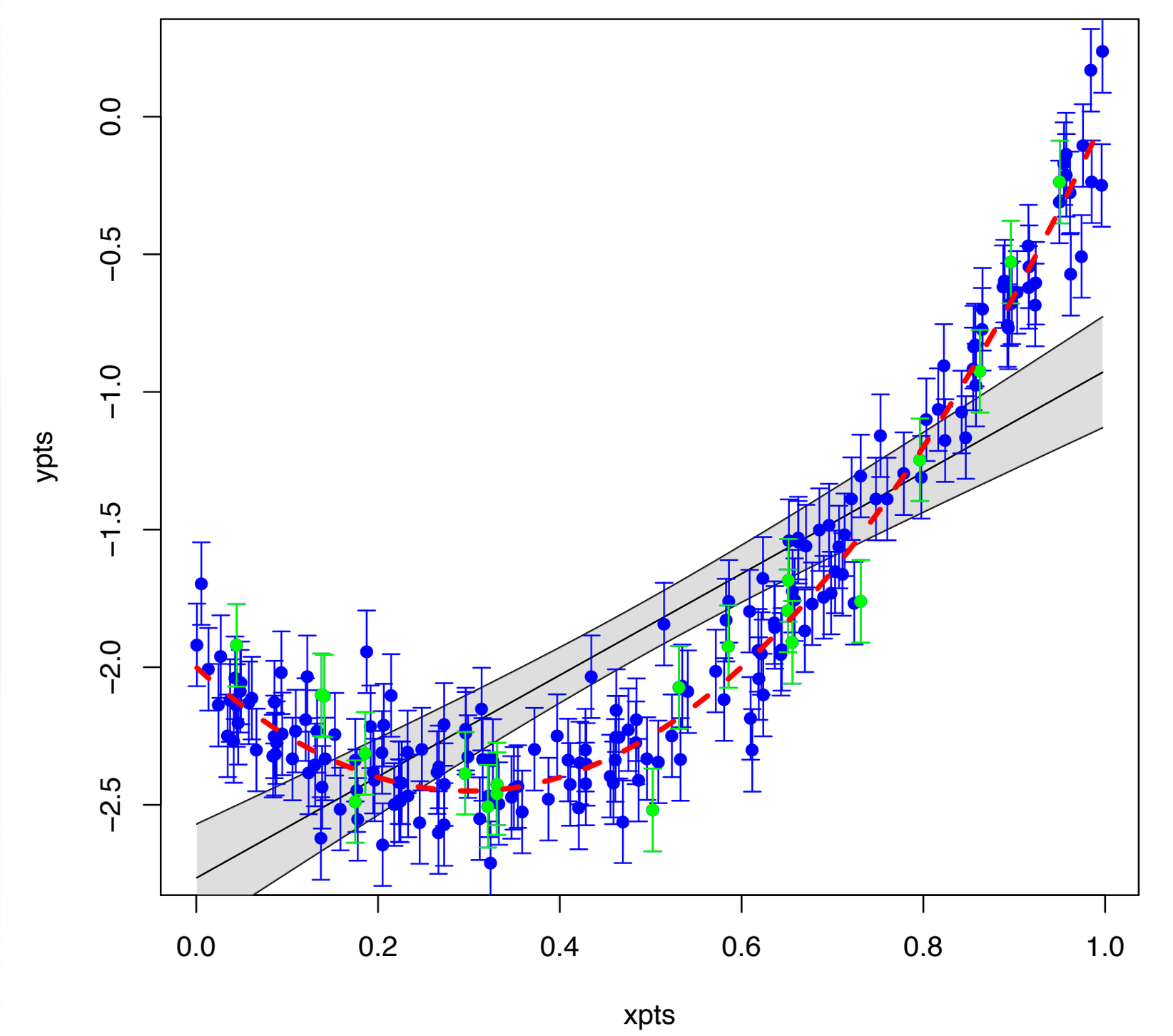
Comparison - linear regression



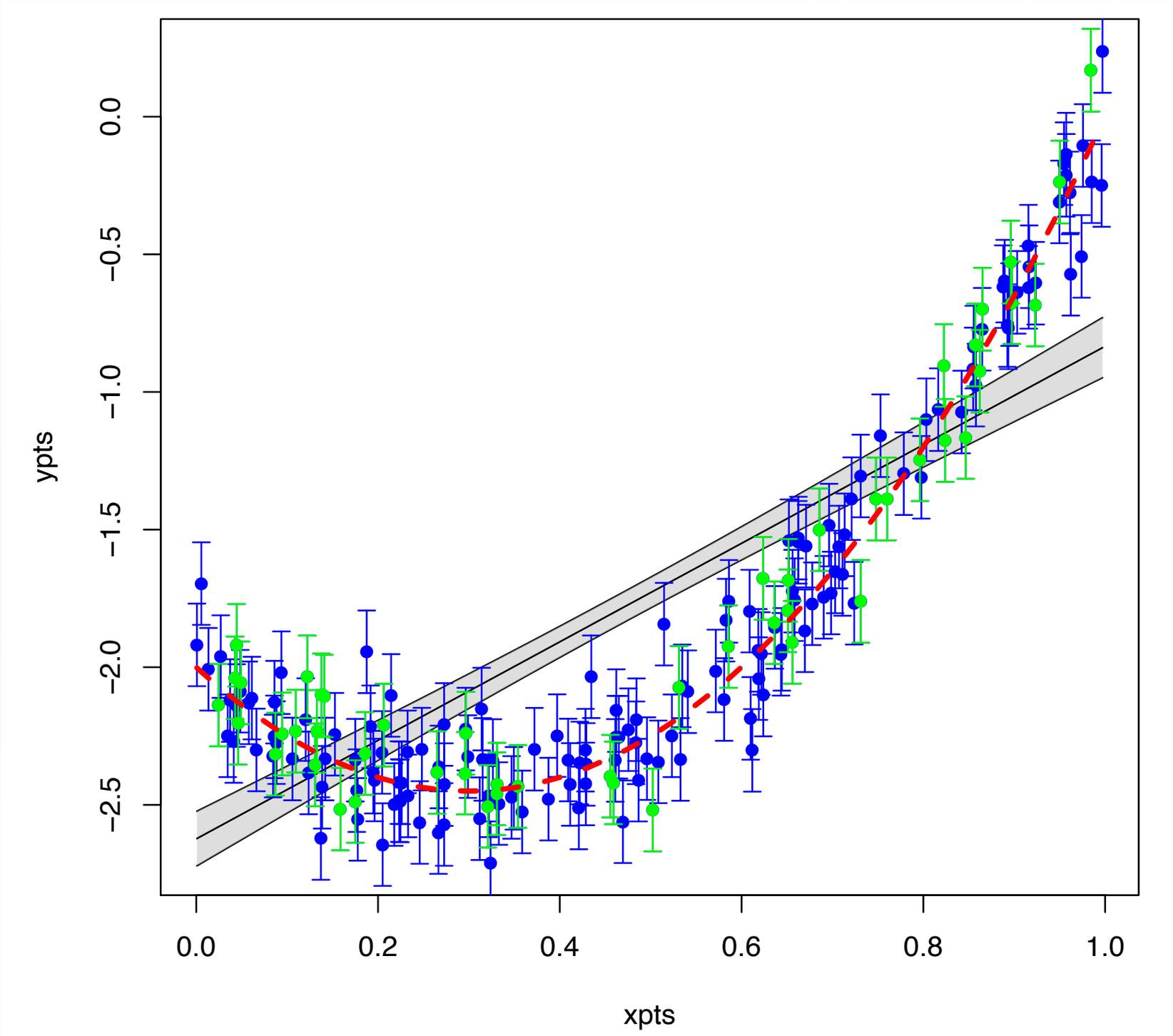
Comparison - linear regression



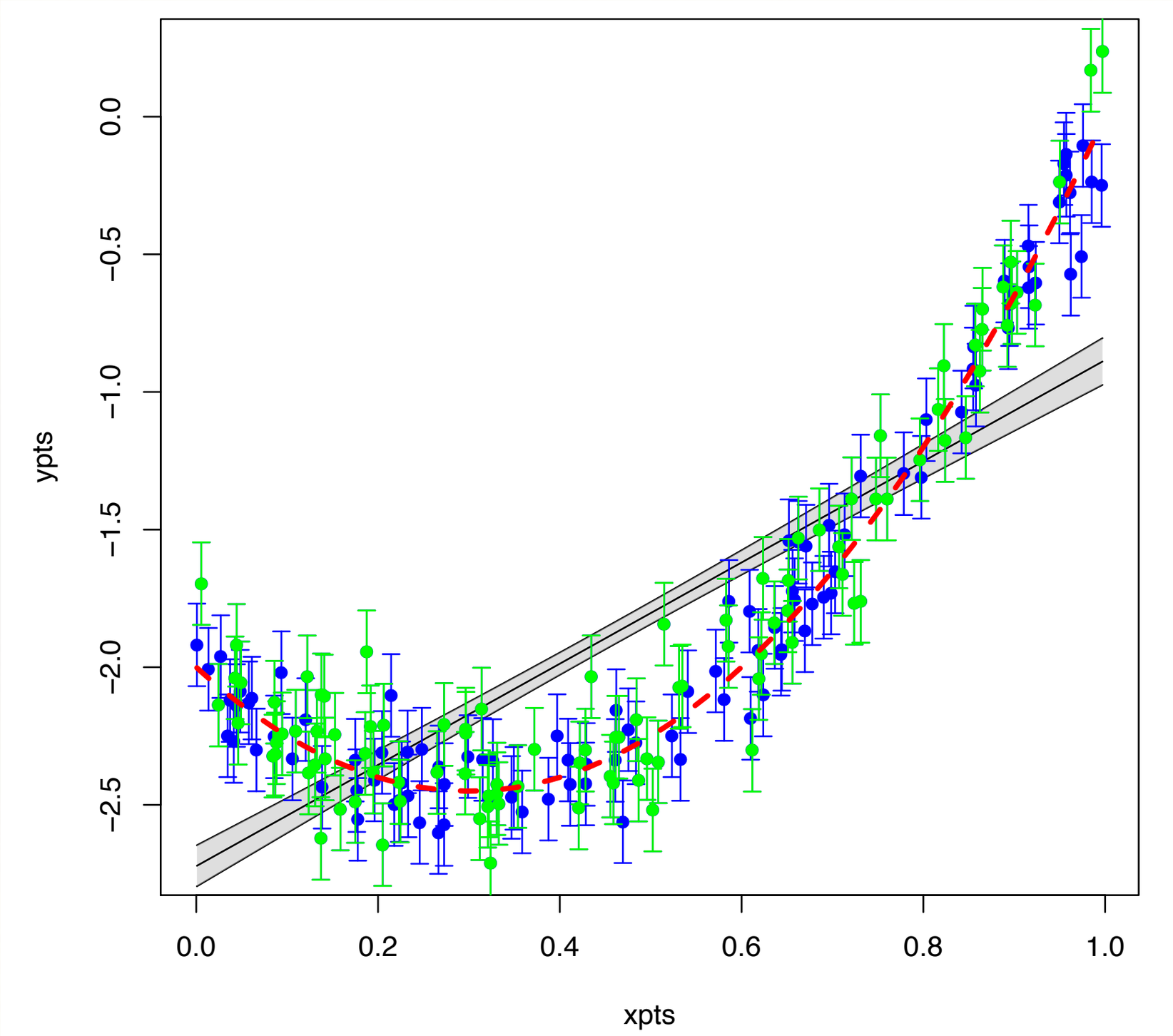
Comparison - linear regression



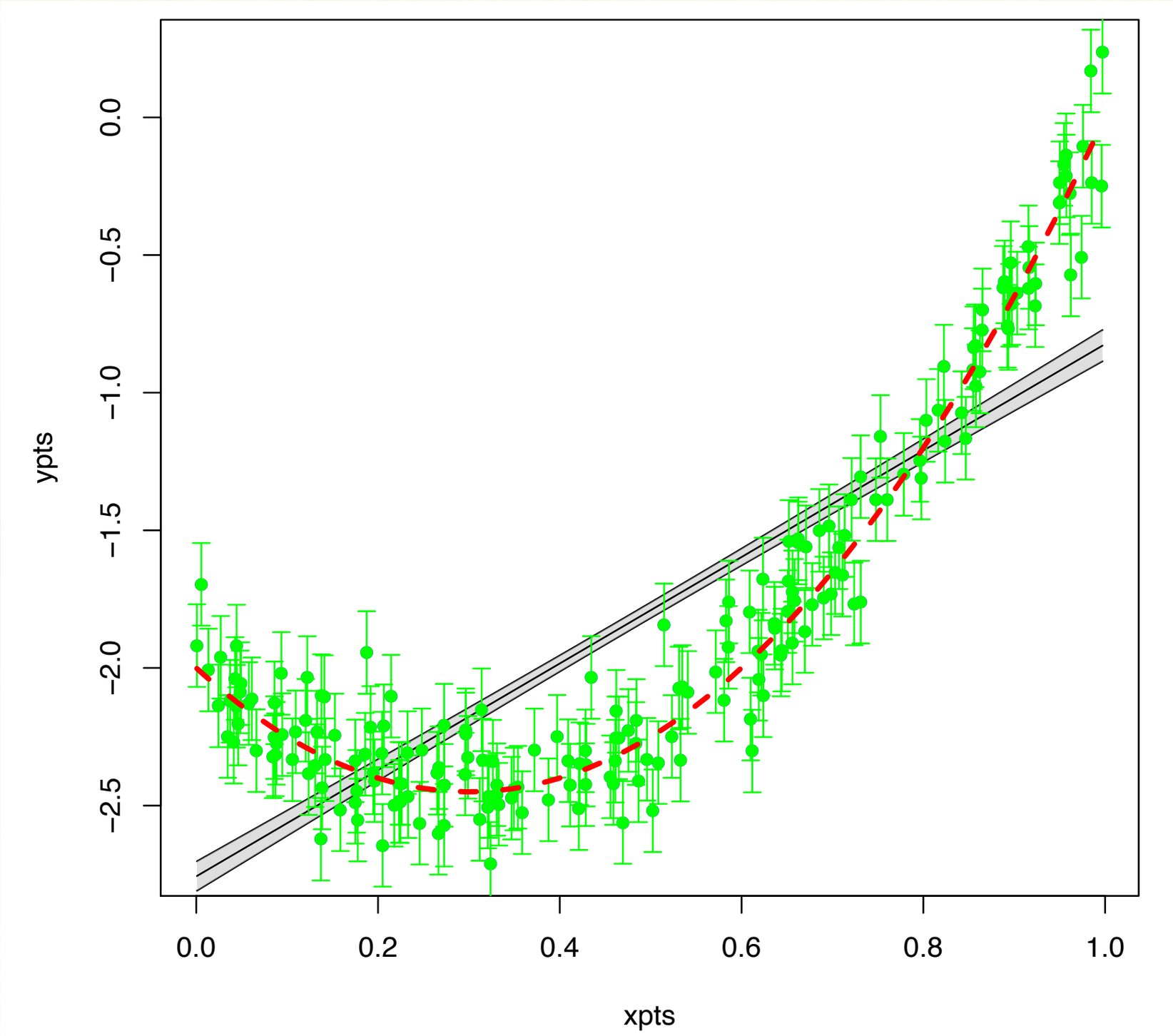
Comparison - linear regression



Comparison - linear regression



Comparison - linear regression



Gaussian process kernels

- The effectiveness of the method depends on the choice of covariance function.
- Common to use stationary (depends only on $\mathbf{x_1} - \mathbf{x_2}$) and isotropic (depends only on $\tau = |\mathbf{x_1} - \mathbf{x_2}|$) covariance functions.
- Need to define a distance metric on parameter space

$$|\mathbf{x_1} - \mathbf{x_2}| = g_{ab}(\mathbf{x_1} - \mathbf{x_2})^a(\mathbf{x_1} - \mathbf{x_2})^b$$

- and a function of distance, $k(\tau)$.
- Squared exponential

$$k_{\mathrm{SE}}(\tau) = \sigma_f^2 \exp\left(-\frac{1}{2}\tau^2\right)$$

Gaussian process kernels

- Power-law exponential

$$k_{\mathrm{PLE}}(\tau) = \sigma_f^2 \exp\left(-\frac{1}{2}\tau^\eta\right)$$

- Cauchy

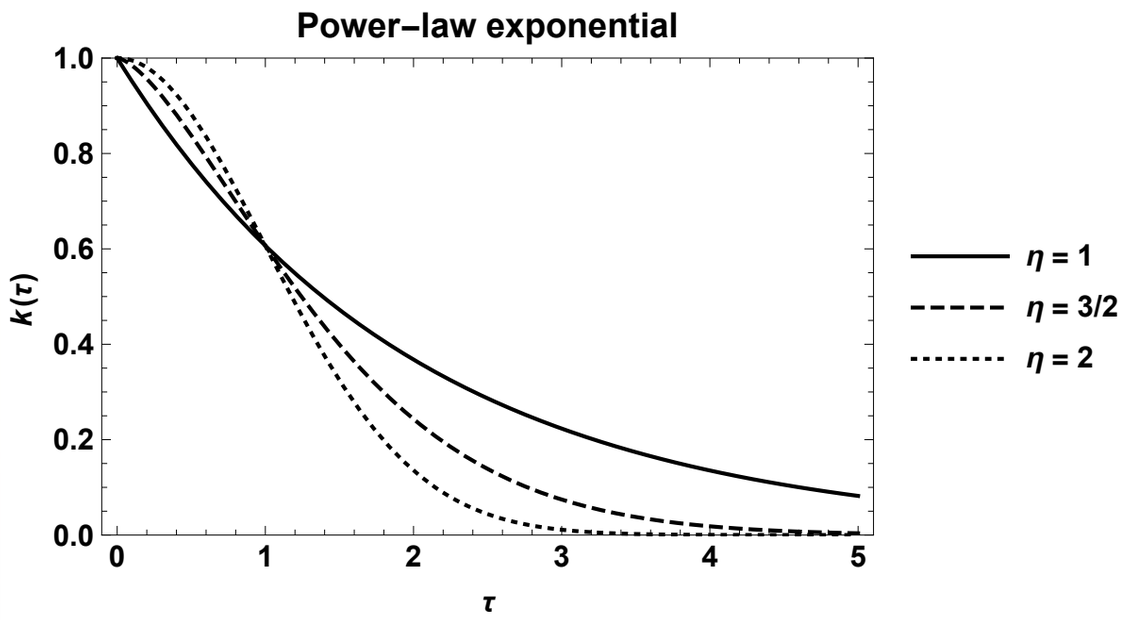
$$k_{\mathrm{Cauchy}}(\tau) = \frac{\sigma_f^2}{(1+\tau^2/2\eta)^\eta}$$

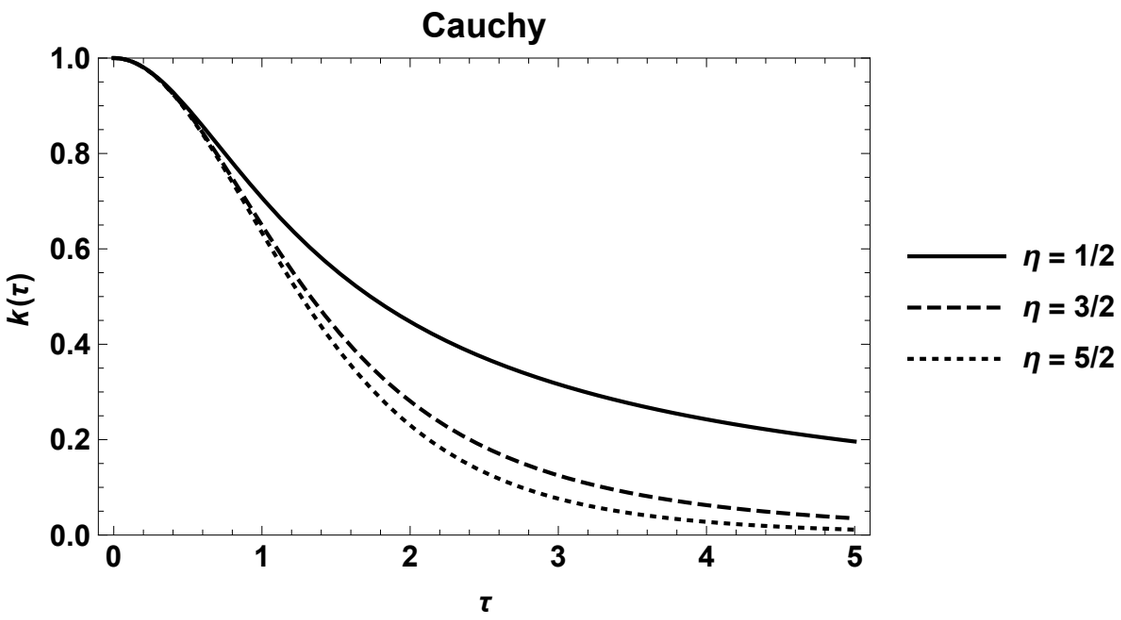
- Matern

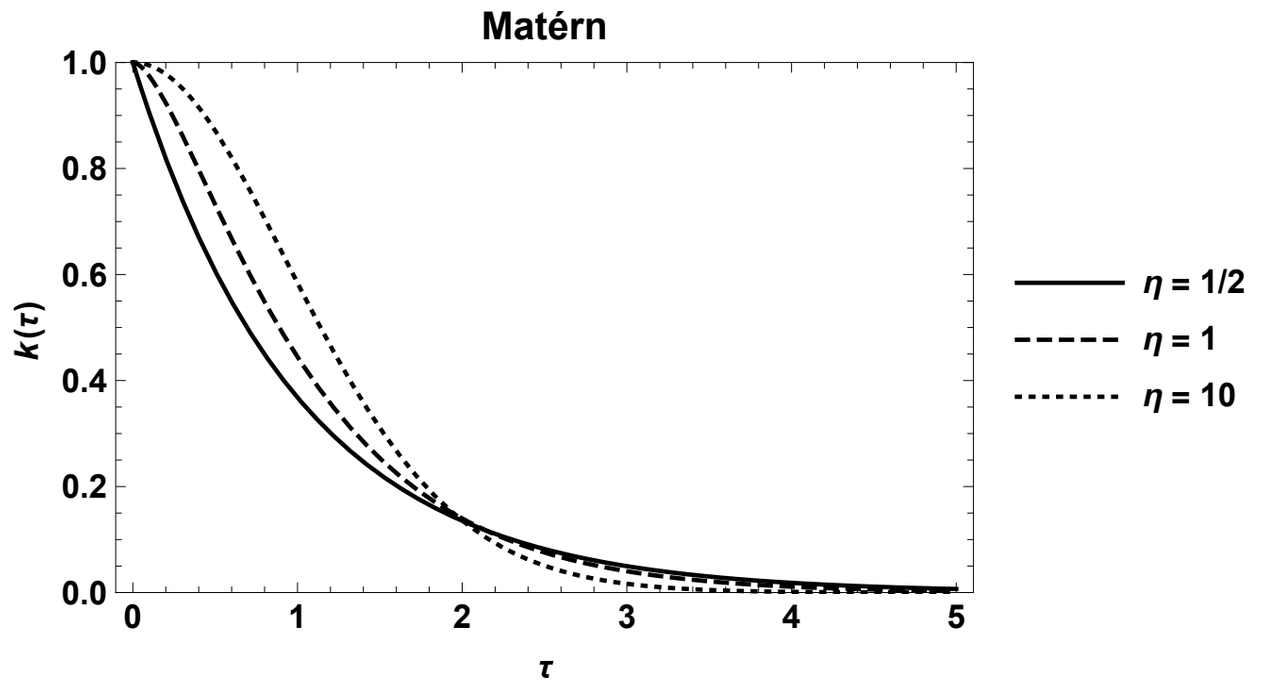
$$k_{\mathrm{Mat}}(\tau) = \frac{\sigma_f^2 2^{1-\eta}}{\Gamma(\eta)}\left(\sqrt{2\eta}\,\tau\right)^\eta K_\eta\left(\sqrt{2\eta}\,\tau\right)$$

- Or covariance functions with compact support, e.g., Wendland polynomials.

Gaussian process kernels







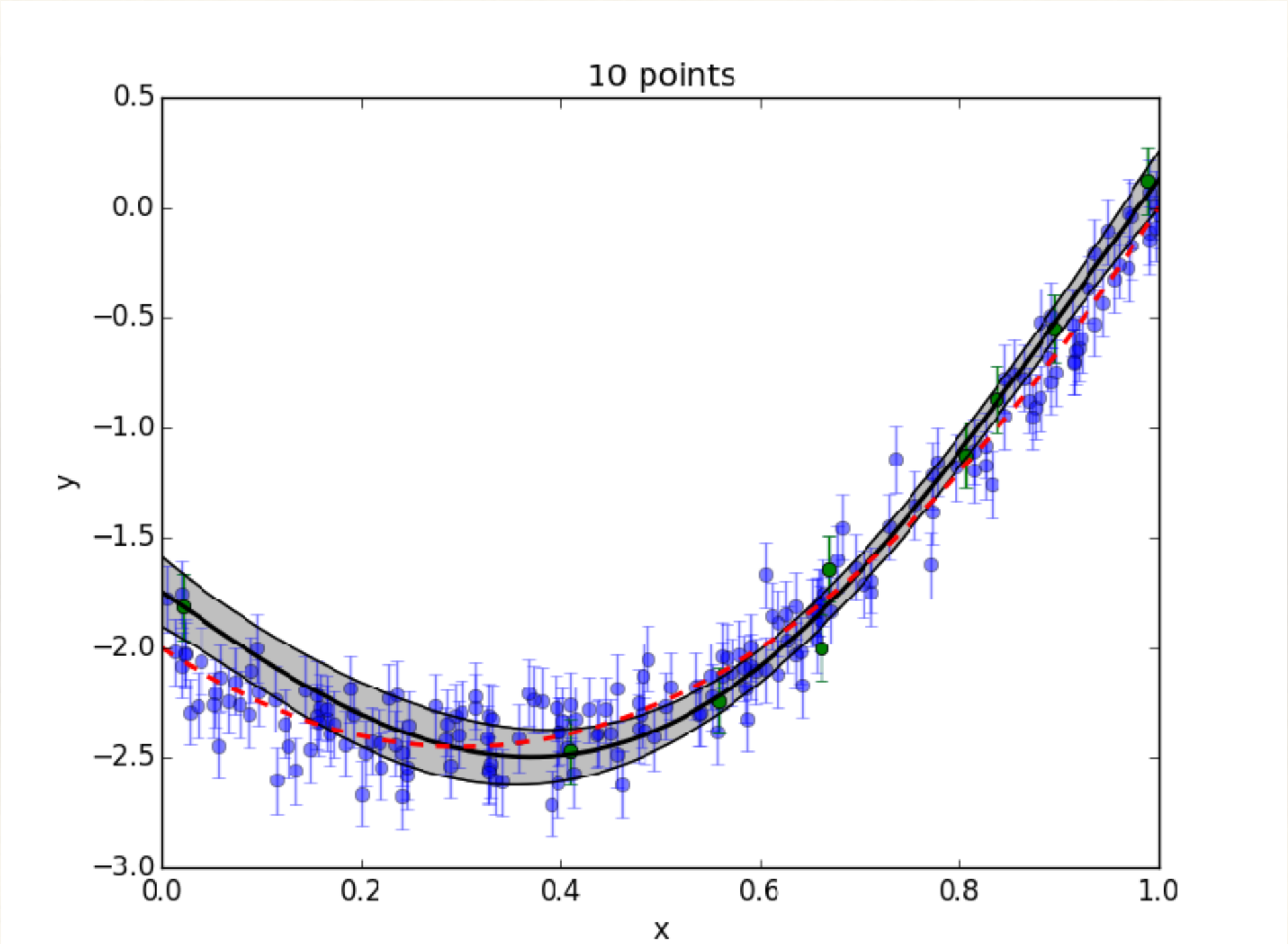
Kernel hyperparameters

- Kernel functions depend on a number of *hyperparameters*.
- Determine these from the *hyperlikelihood* for the training set.

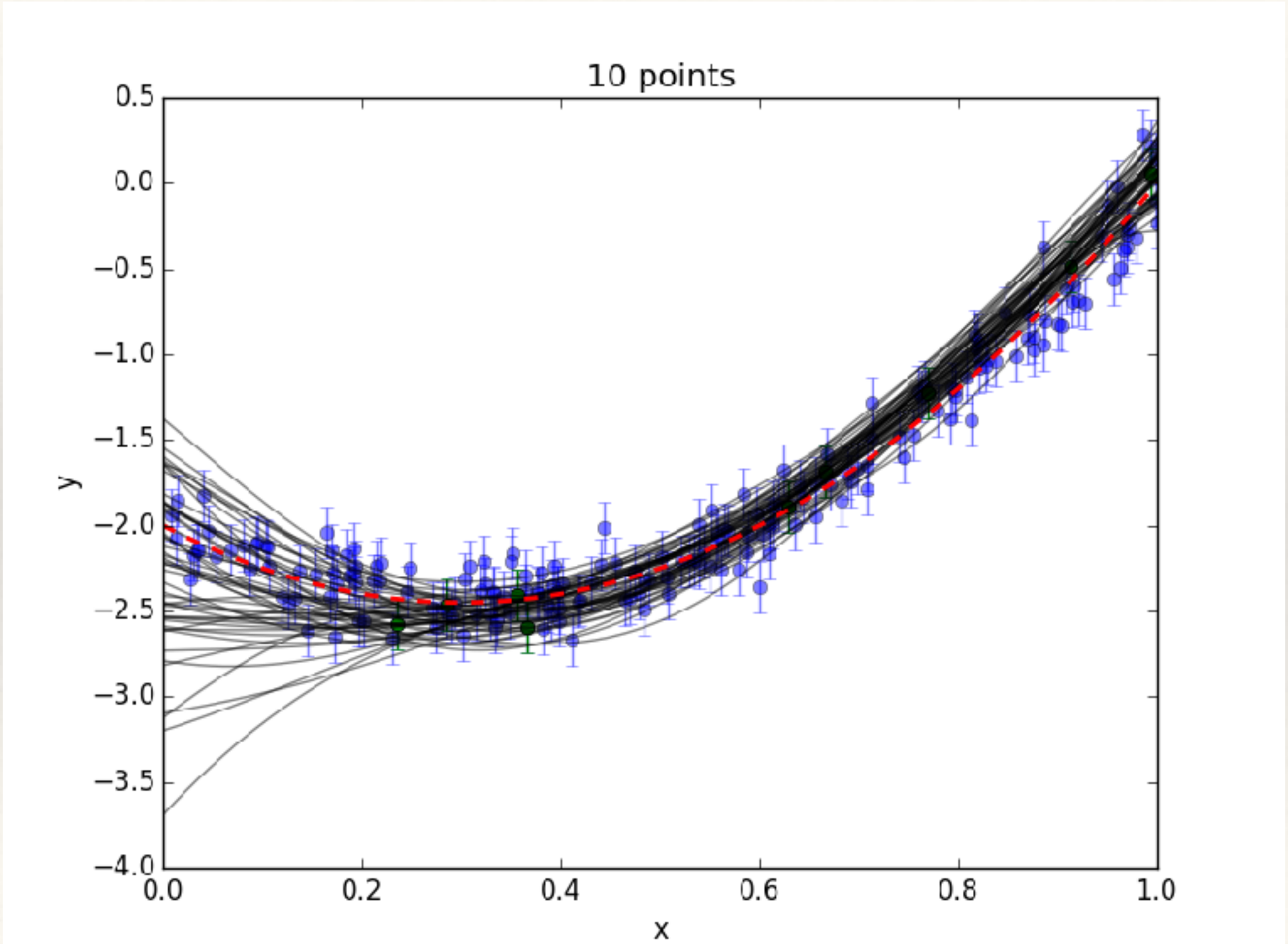
$$p(\{d_i\}) = \frac{1}{\sqrt{(2\pi)^n |\mathbf{K}|}} \exp\left[-\frac{1}{2}\sum_{ij}(d_i - m(\mathbf{x_i}))\,\mathbf{K}^{-1}_{ij}\,(d_j - m(\mathbf{x_j}))\right]$$

- Can find optimal hyperparameters via maximisation or obtain a posterior distribution for the parameters that can subsequently be marginalised over.

Kernel hyperparameters

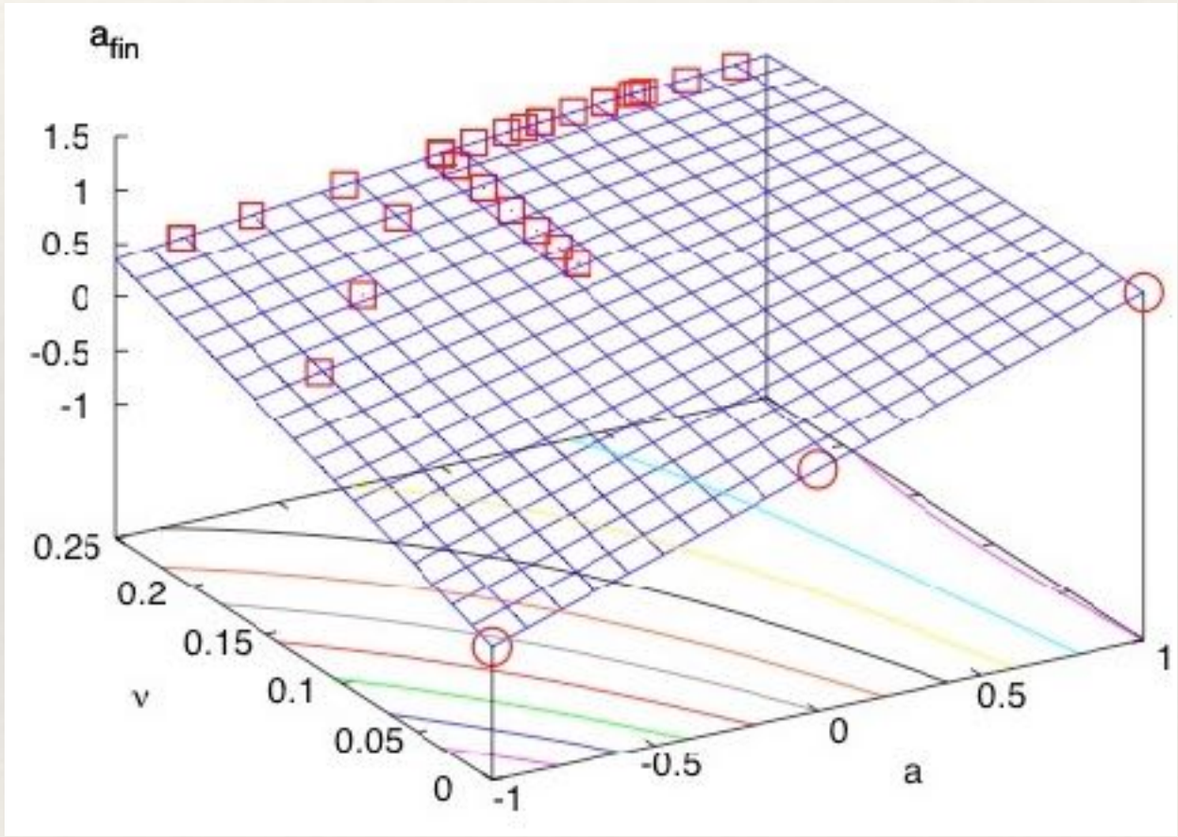


Kernel hyperparameters



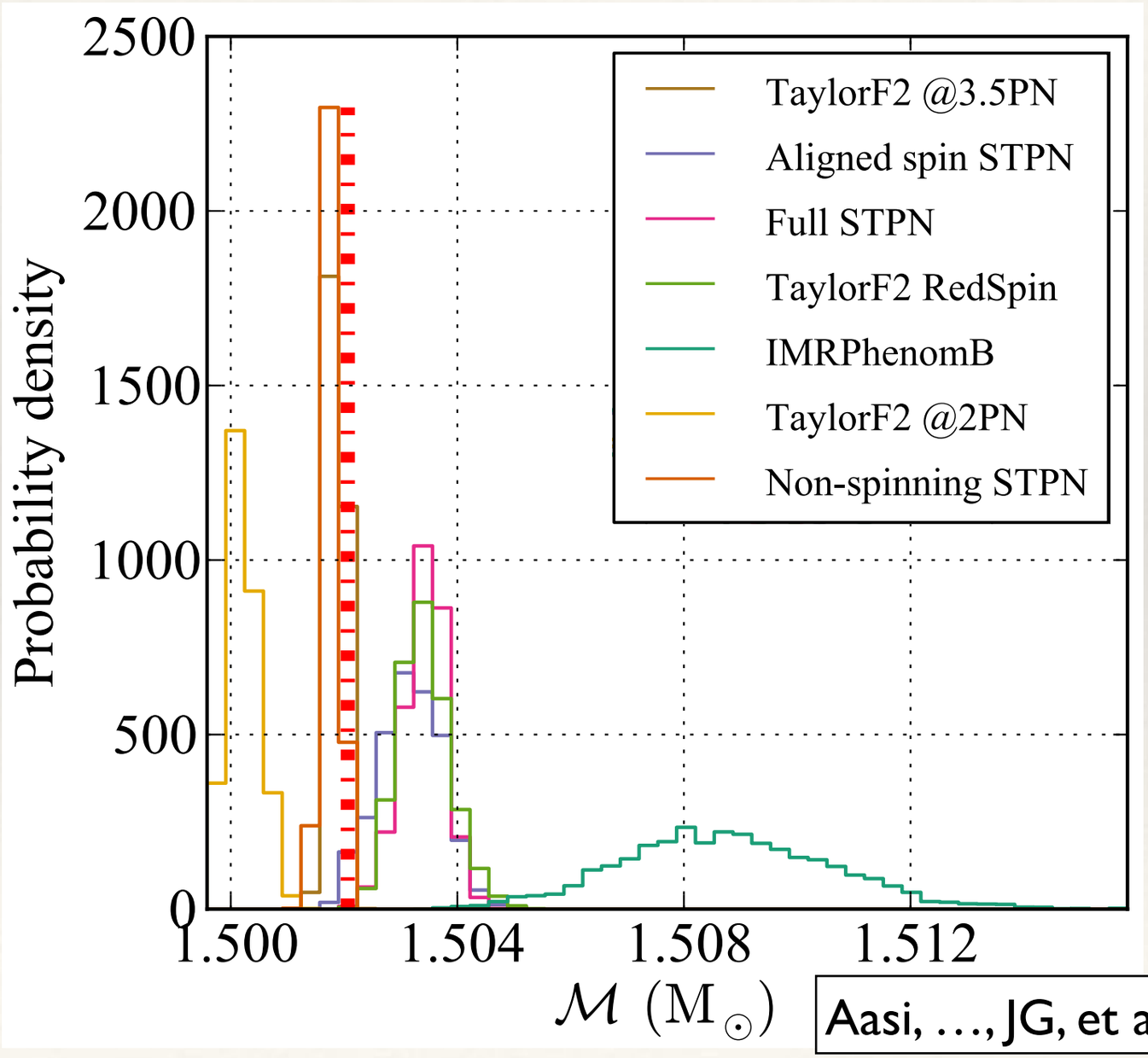
Example: GW model uncertainty

- Inference using accurate models can be computationally prohibitive.
- For example, gravitational wave models can be computed accurately using numerical relativity, but each simulation takes many days.
- Inference must therefore rely on approximations.



The trouble with approximations

- Use of approximate models leads to bias in parameter inference.



Aasi, …, JG, et al. (LVC) (2013)

Example: GW model uncertainty

- Inference for gravitational wave detectors uses a likelihood

$$p(d|\vec{\theta}) \propto \exp\left[-\frac{1}{2}(d-h(\vec{\theta})|d-h(\vec{\theta}))\right] \qquad (a|b) = \int_{-\infty}^{\infty} \frac{\tilde{a}^*(f)\tilde{b}(f)+\tilde{a}(f)\tilde{b}^*(f)}{S_h(f)}\mathrm{d}f$$

- The exact model, $h(\vec{\theta}) = h_{\mathrm{GR}}(\vec{\theta})$, is expensive to compute. Instead we use approximations $h(\vec{\theta}) = h_{\mathrm{AP}}(\vec{\theta})$.

- We can fold model uncertainties into inference by modelling the difference, $\delta h(\vec{\theta}) = h_{\mathrm{GR}}(\vec{\theta}) - h_{\mathrm{AP}}(\vec{\theta})$, as a GP, conditioned on available simulations, and marginalising over the difference distribution.

Example: GW model uncertainty

- Assuming the numerical simulations are perfect, the GP probability distribution for $\delta h(\vec{\theta})$ is

$$p(\delta h(\vec{\theta})) \propto \exp\left[\frac{(\delta h(t;\vec{\theta}) - \mu(t;\vec{\theta})|\delta h(t;\vec{\theta}) - \mu(t;\vec{\theta}))}{2\sigma^2(\vec{\theta})}\right]$$

$$\mu(\vec{\theta}) = [\mathbf{K}_*]_i \left[\mathbf{K}^{-1}\right]_{ij} \delta h(\vec{\theta}_j) \qquad \sigma^2(\vec{\theta}) = \mathbf{K}_{**} - [\mathbf{K}_*]_i \left[\mathbf{K}^{-1}\right]_{ij} [\mathbf{K}_*]_j$$

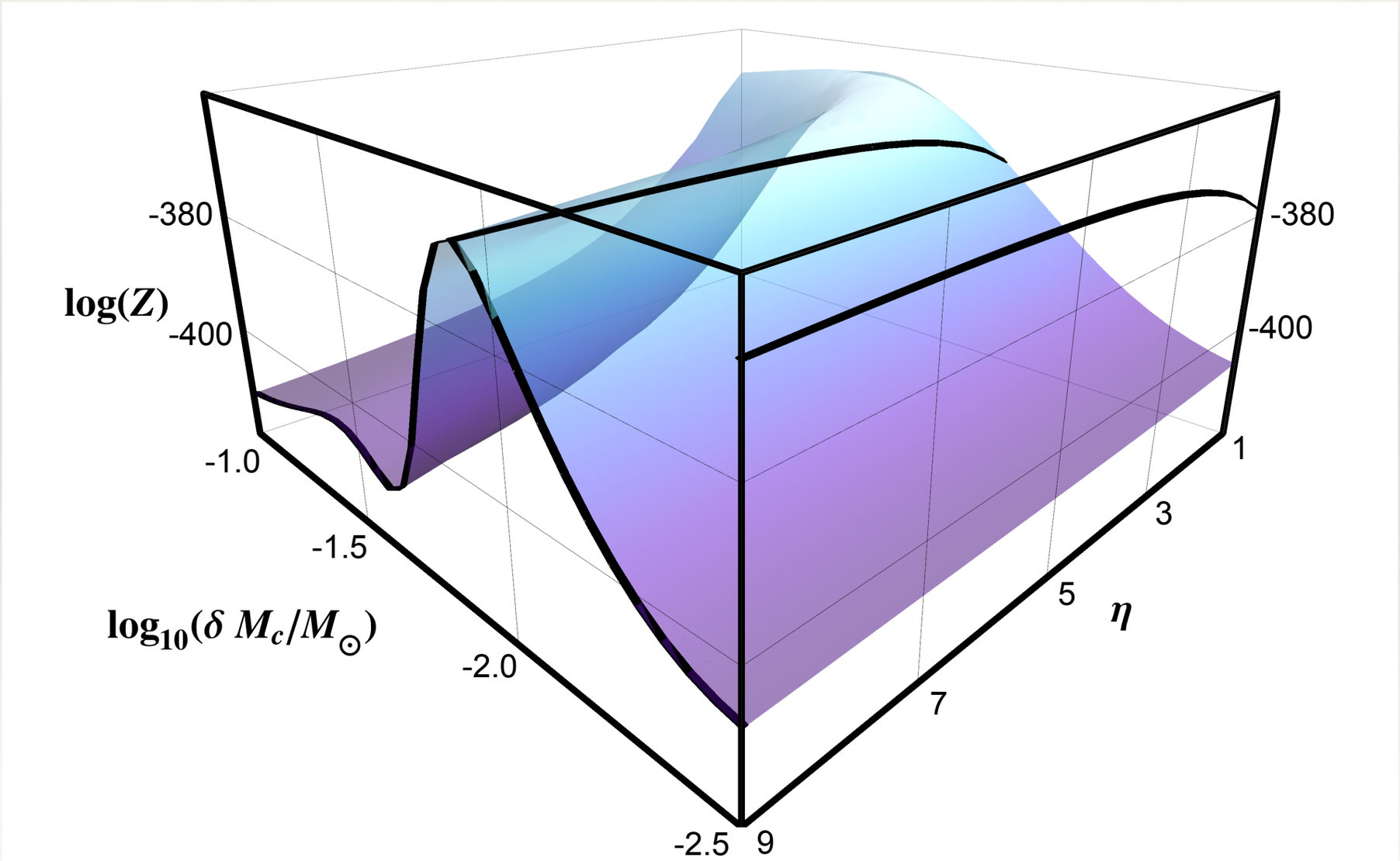
$$[\mathbf{K}]_{ij} = k(\vec{\theta}_i, \vec{\theta}_j) \qquad [\mathbf{K}_*]_i = k(\vec{\theta}, \vec{\theta}_i) \qquad \mathbf{K}_{**} = k(\vec{\theta}, \vec{\theta})$$

- The corresponding marginalised likelihood for the data is

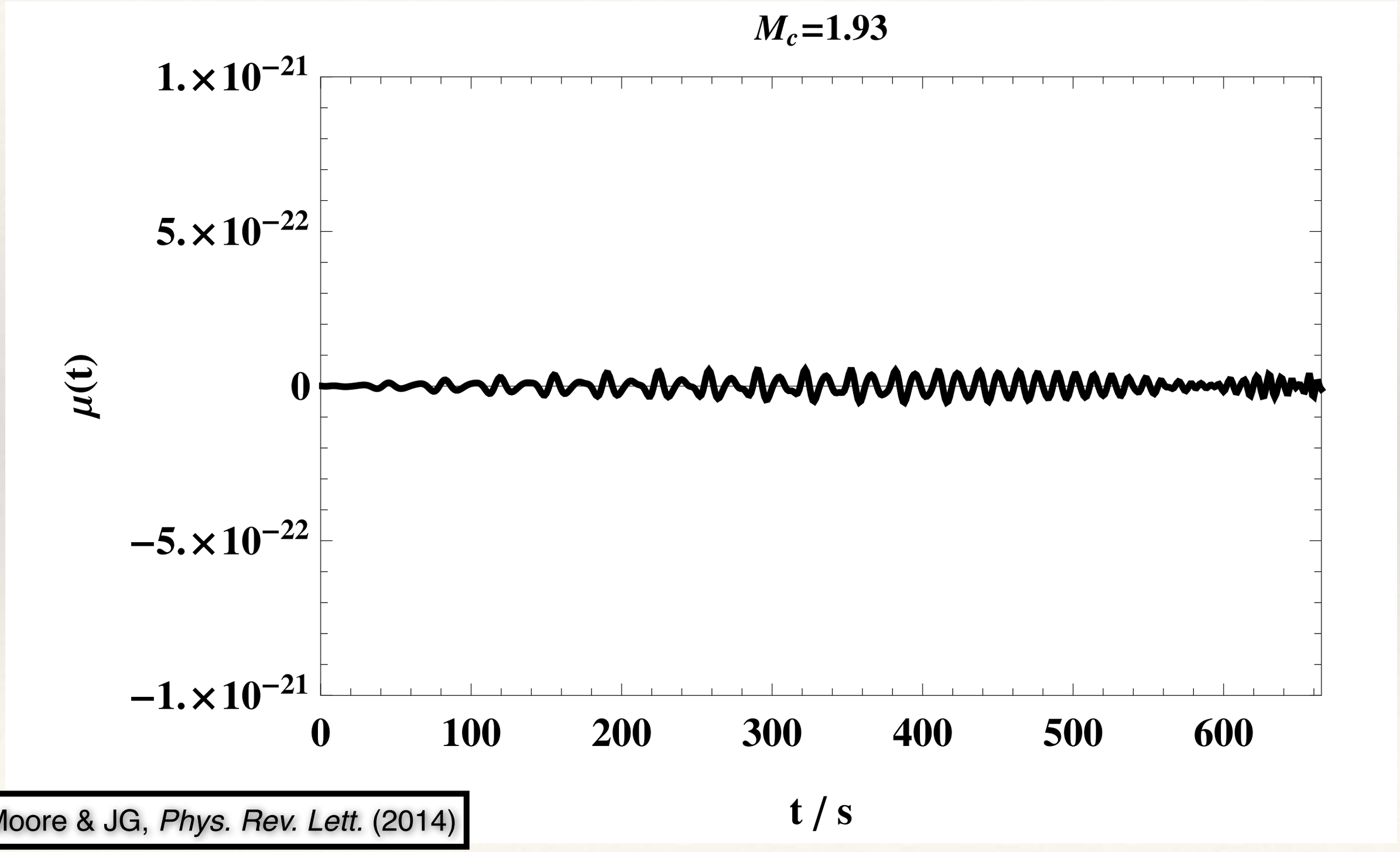
$$\mathcal{L}_{\rm GP}(\vec{\theta}) \propto \frac{1}{1+\sigma^2(\vec{\theta})} \exp\left[-\frac{1}{2}\frac{\left(d - h_{\rm AP}(t;\vec{\theta}) - \mu(t;\vec{\theta})|d - h_{\rm AP}(t;\vec{\theta}) - \mu(t;\vec{\theta})\right)}{1+\sigma^2(\vec{\theta})}\right]$$

GW model uncertainty: hyperparameters

- Use Matern covariance function - hyperparameters favour squared-exponential.

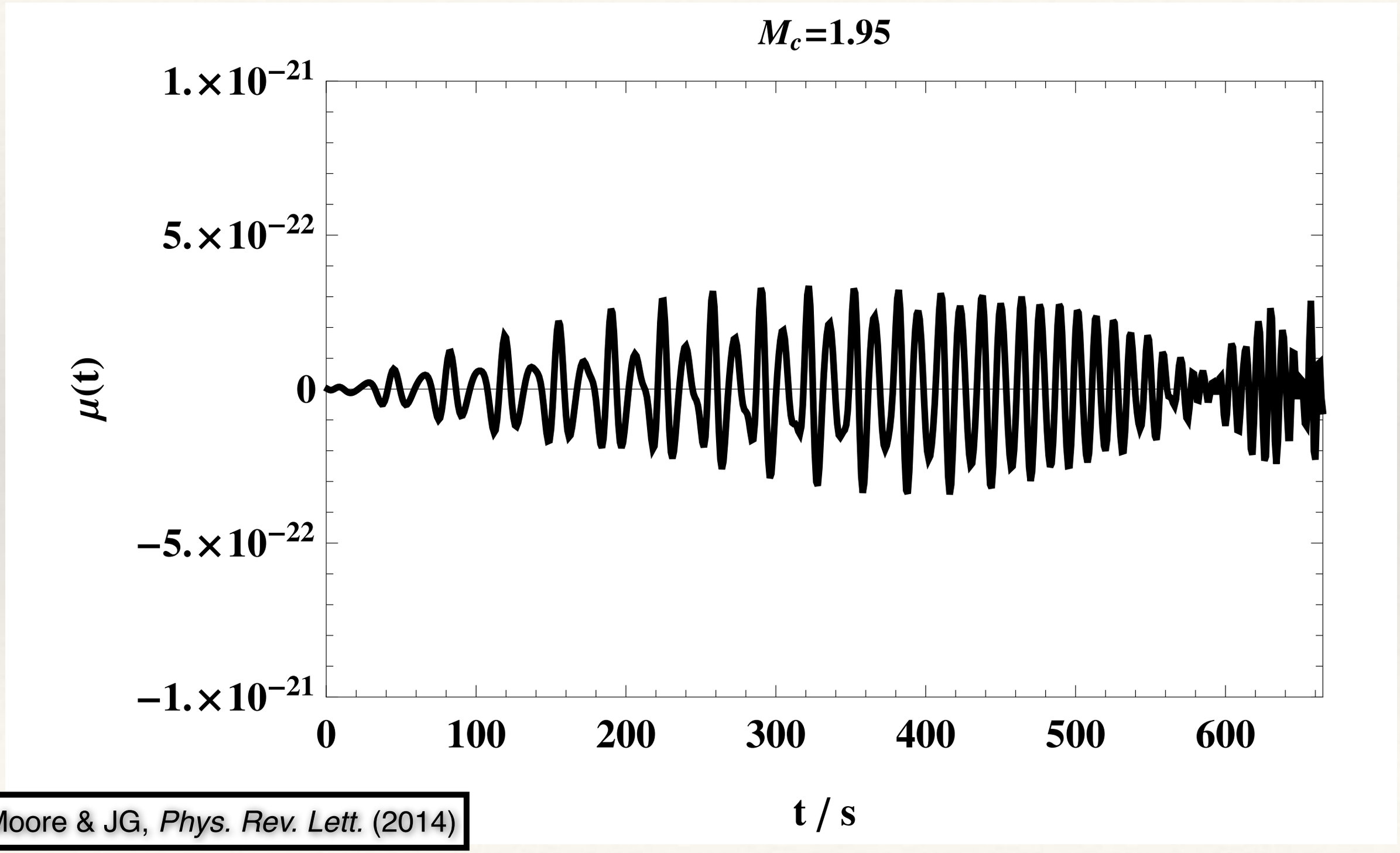


GW model uncertainty: mean



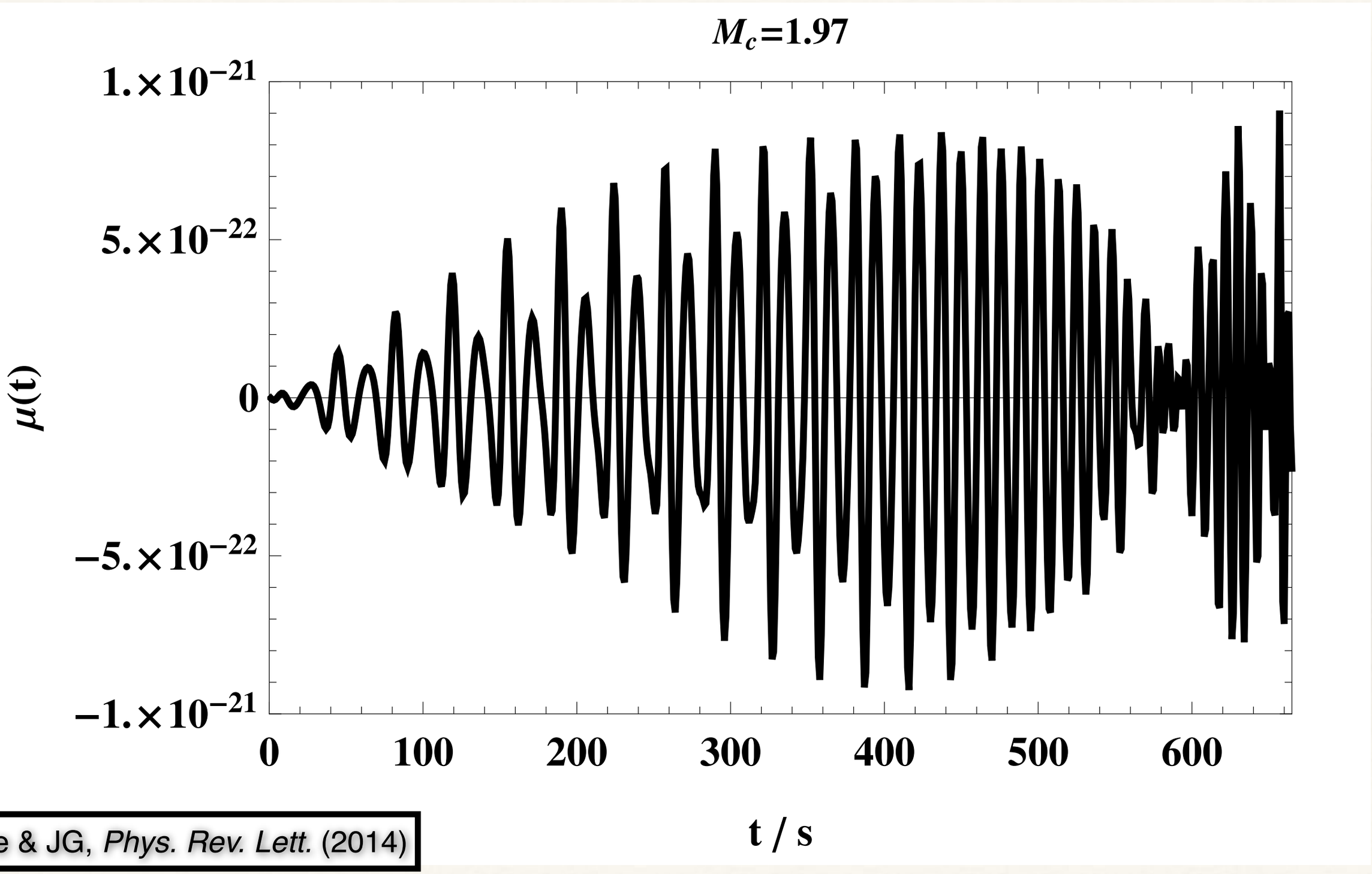
Moore & JG, *Phys. Rev. Lett.* (2014)

GW model uncertainty: mean



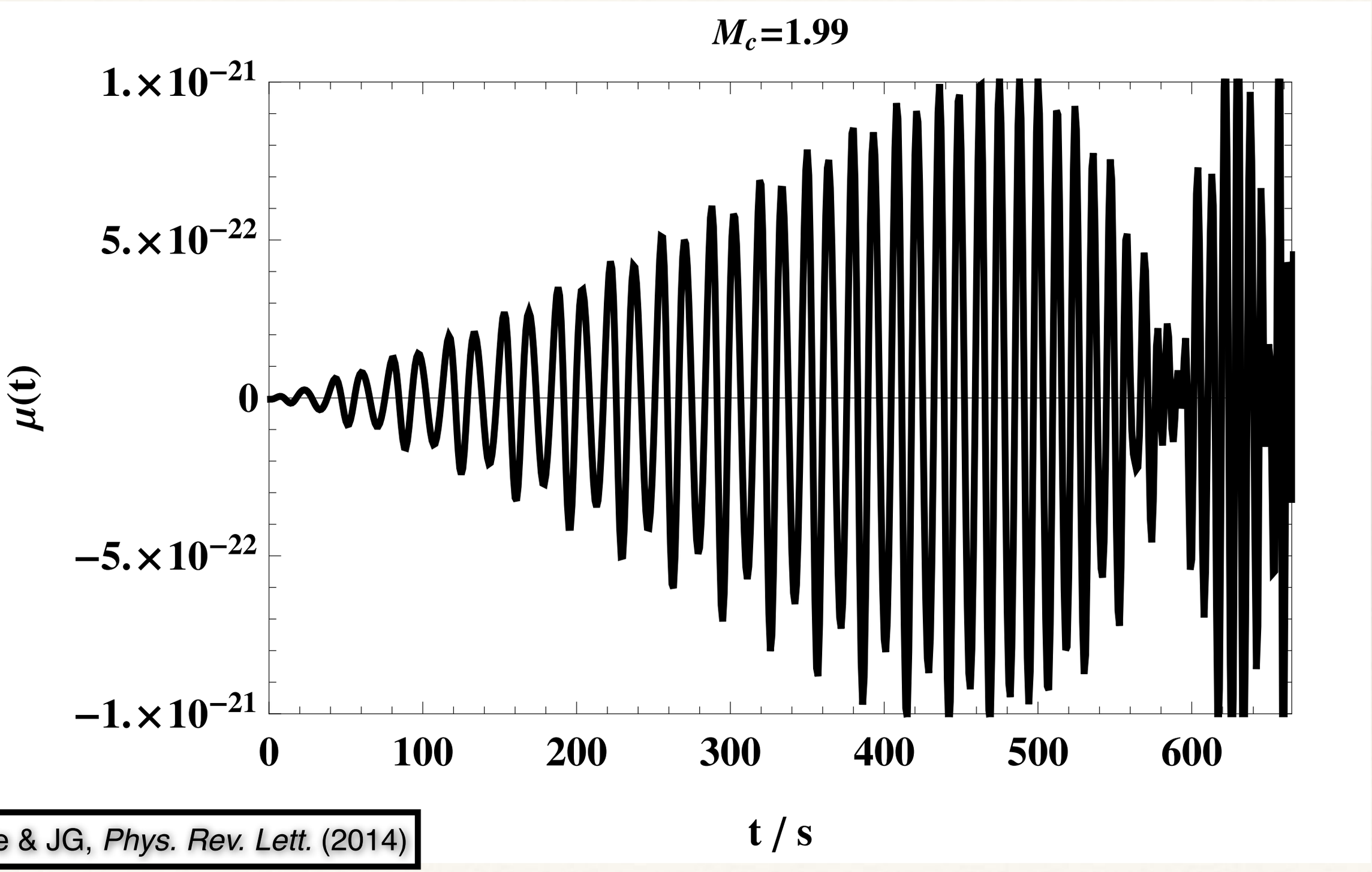
Moore & JG, *Phys. Rev. Lett.* (2014)

GW model uncertainty: mean



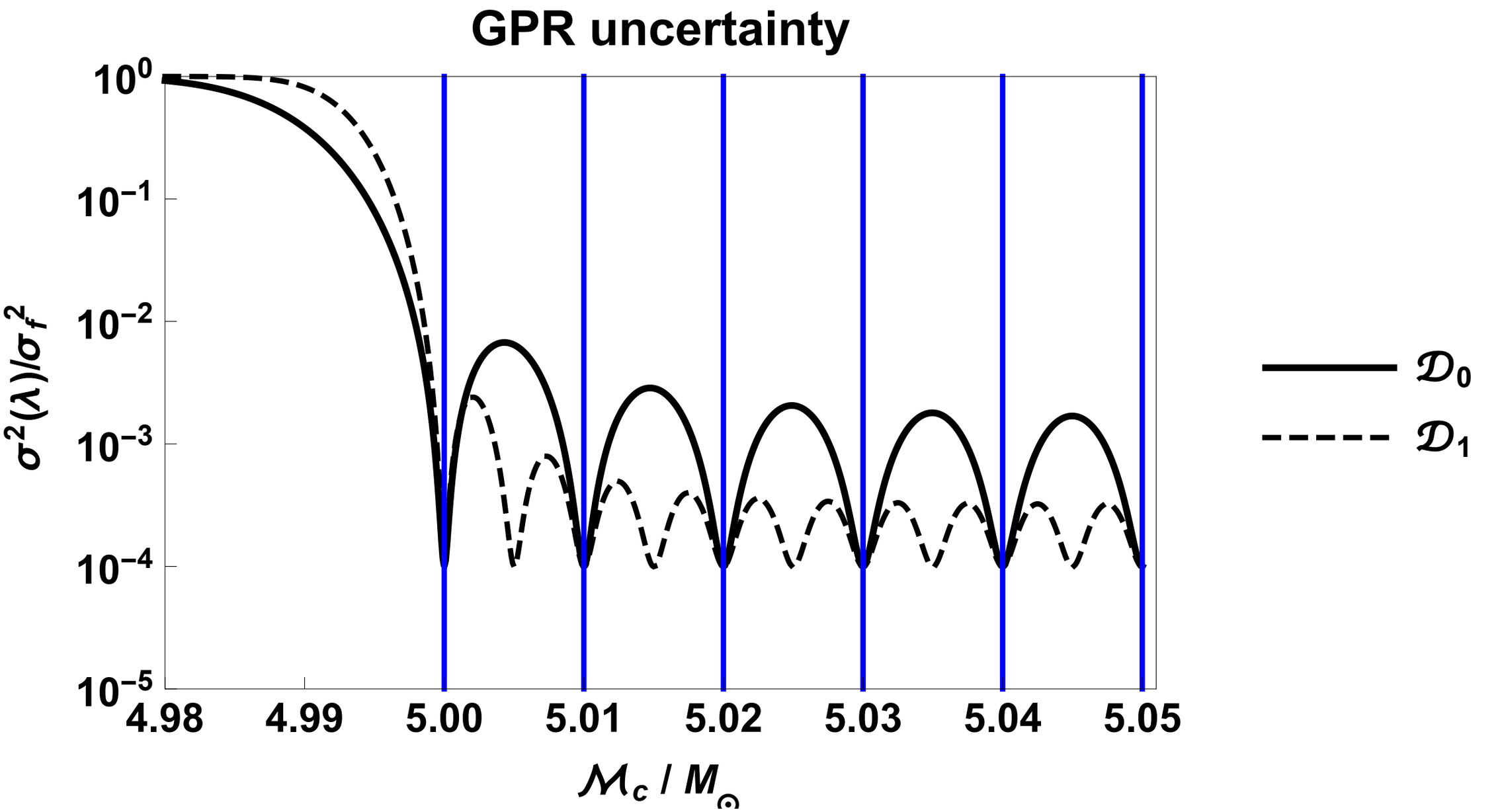
Moore & JG, *Phys. Rev. Lett.* (2014)

GW model uncertainty: mean

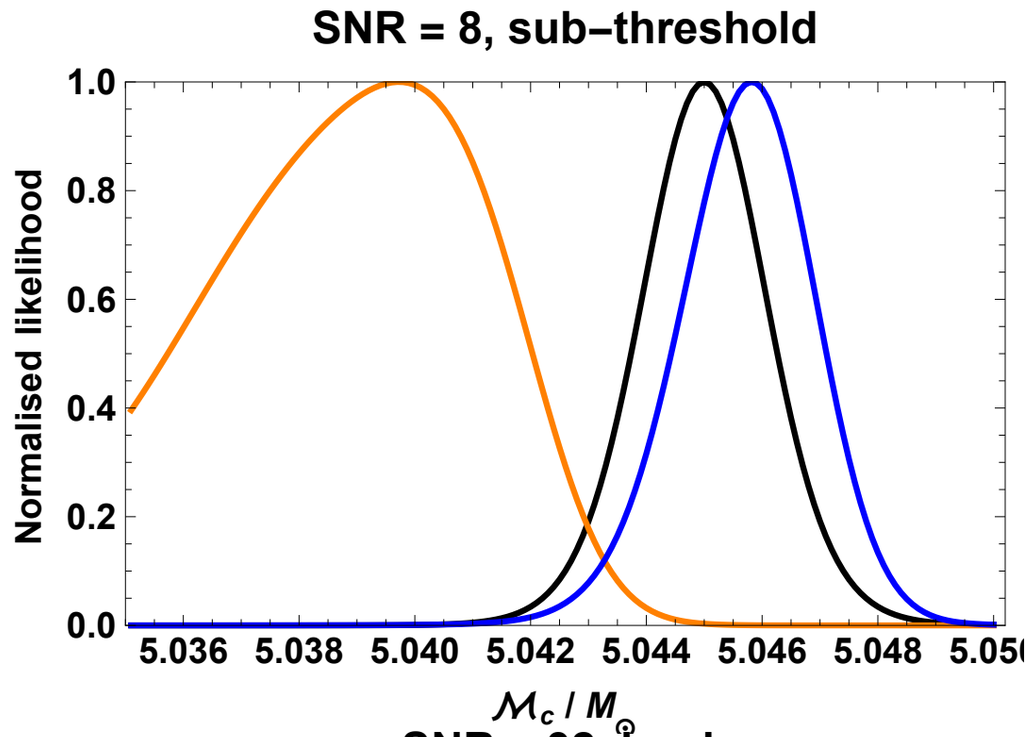


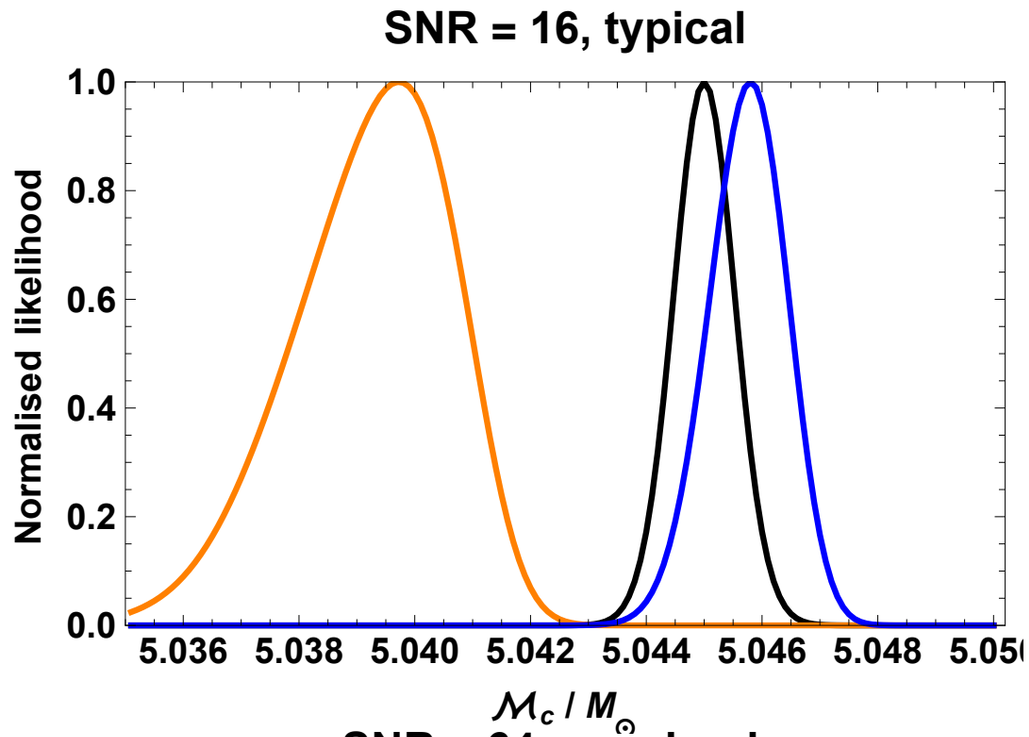
Moore & JG, *Phys. Rev. Lett.* (2014)

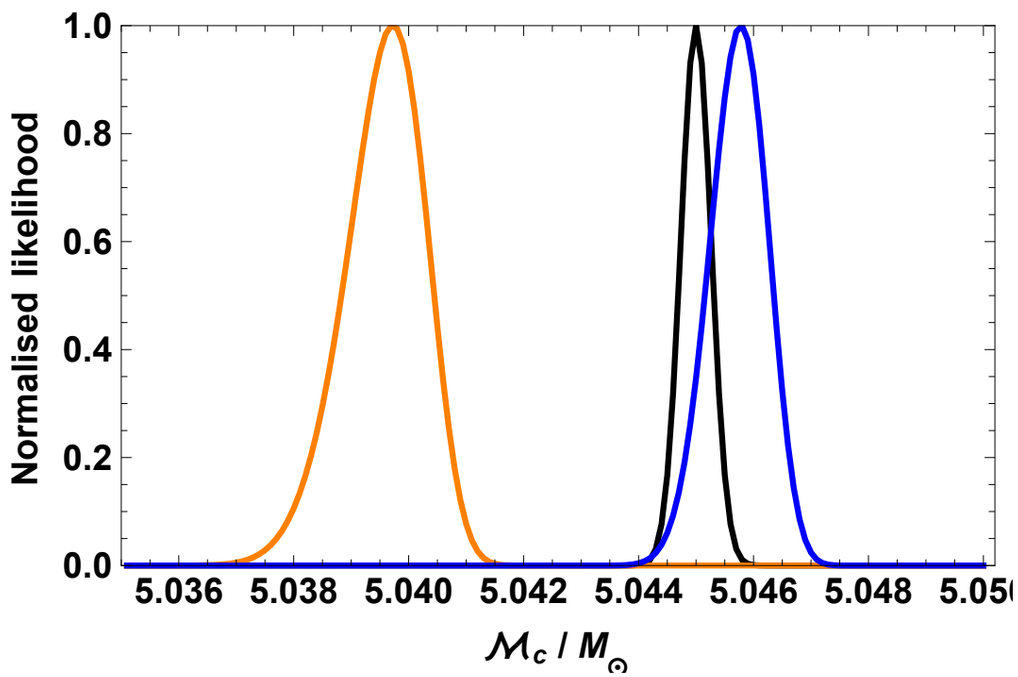
GW model uncertainty: variance

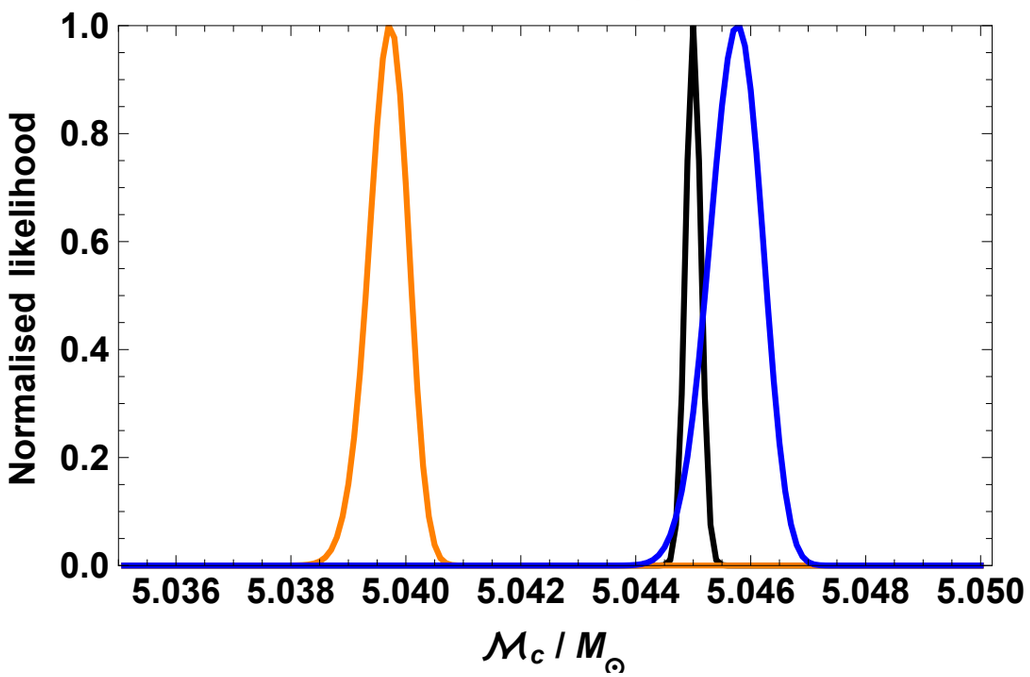


GW model uncertainty: marginalised likelihood



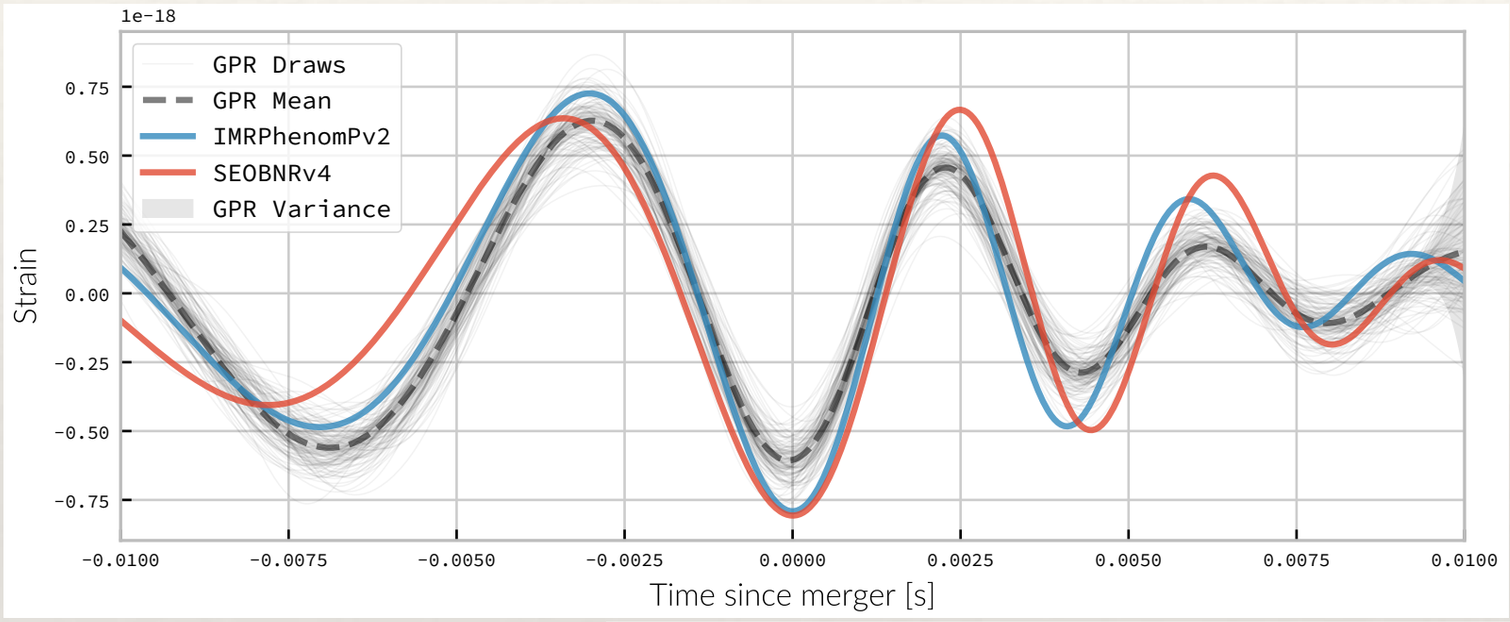


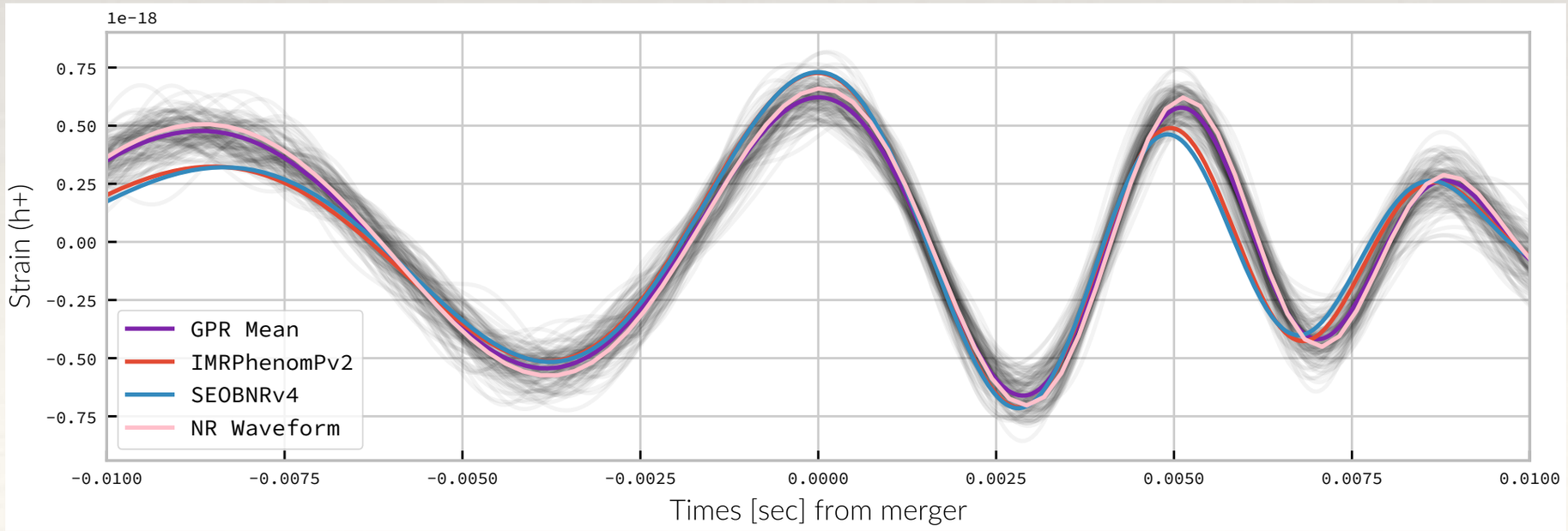




Other examples from gravitational waves

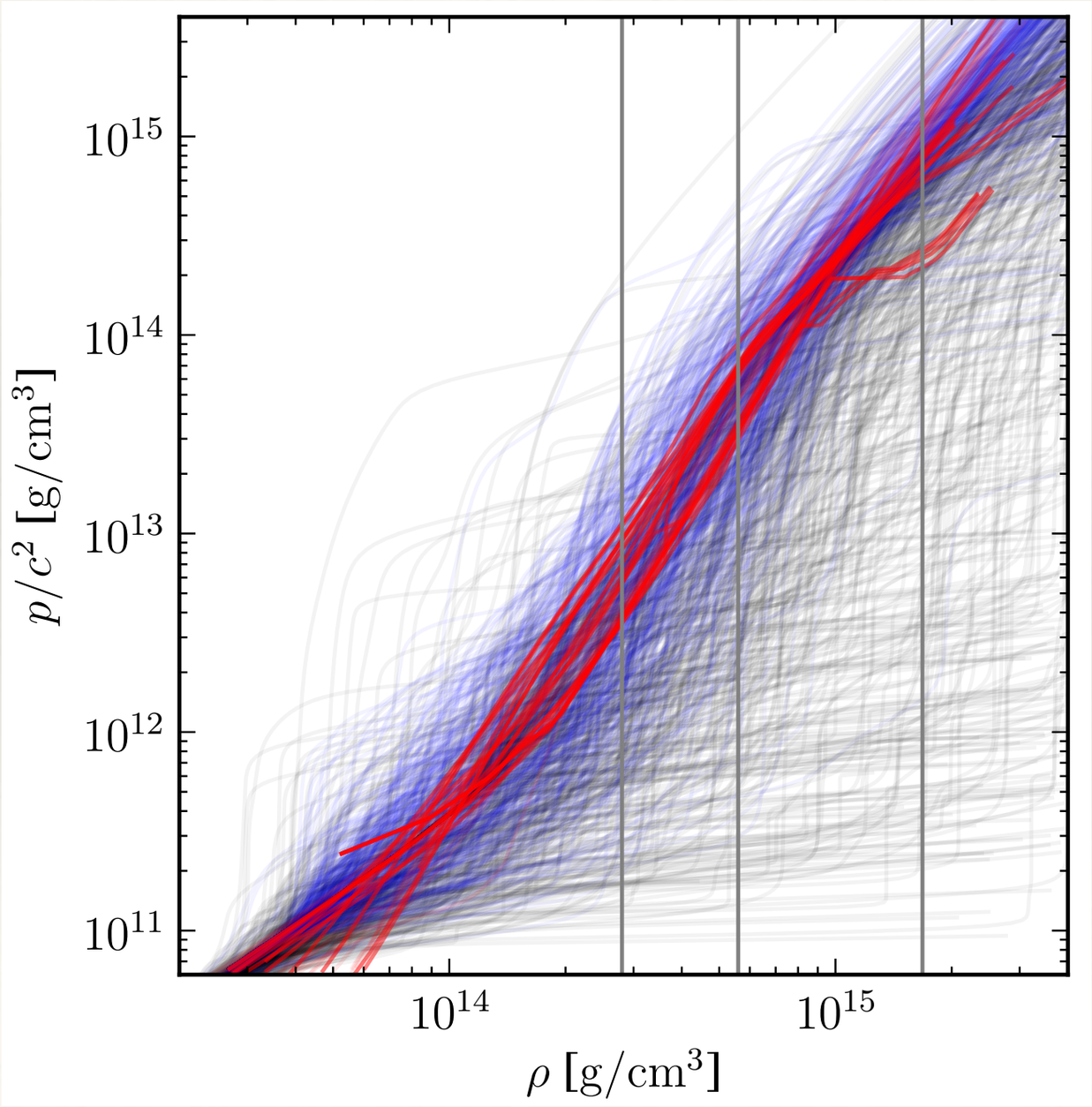
- Williams et al. (2020) also used GPs to build a waveform model with associated uncertainty estimates, but by directly training on numerical relativity simulations.





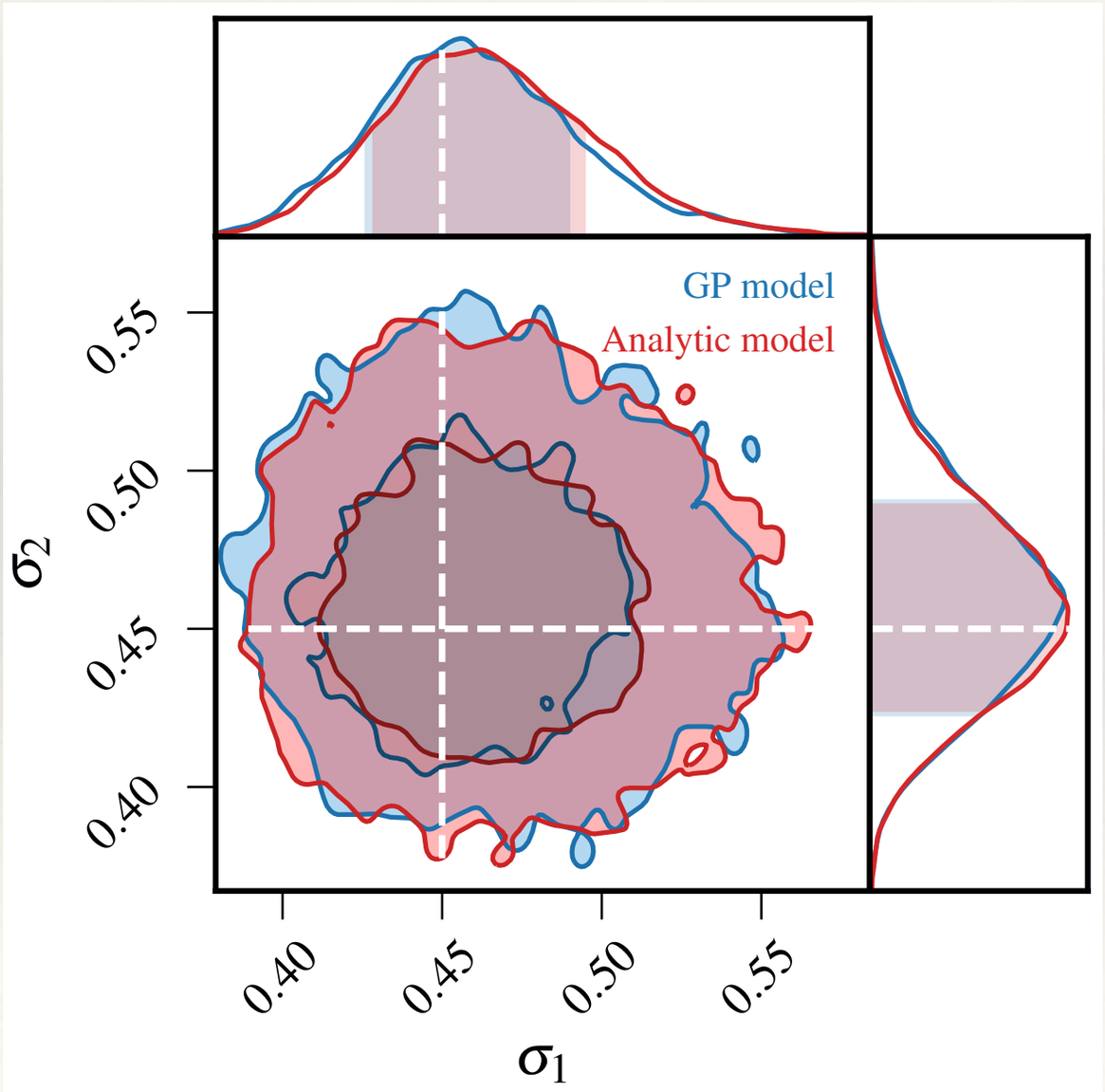
Other examples from gravitational waves

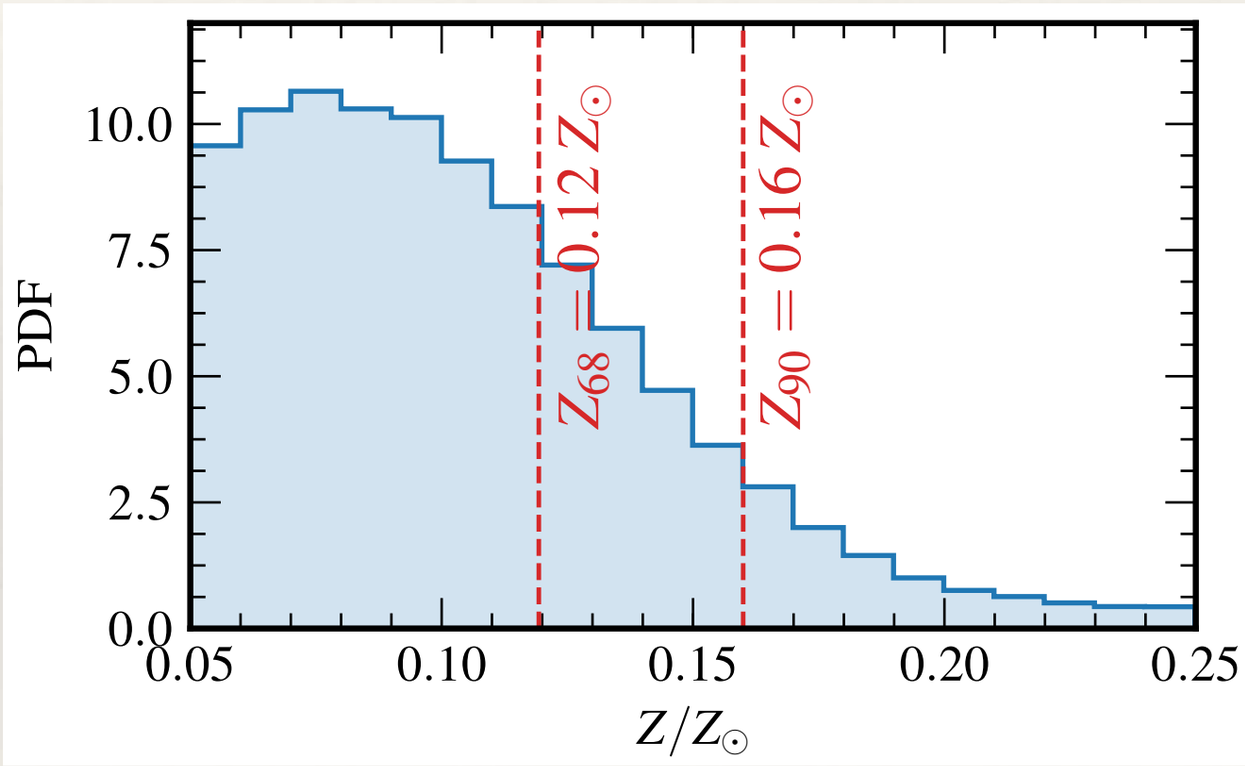
- Landry & Essick (2019) and Essick, Landry and Holz (2019) used GPs to interpolate equation of state models and represent uncertainties in the EoS family.



Other examples from gravitational waves

- Taylor & Gerosa (2018) used GPs to emulate the output of binary population synthesis codes for use in constraining astrophysical models with GW observations.





Etc. etc…….there are now many GW related papers that use GPs, all written in the past 5 years.

Dirichlet processes

- In the same way that a Gaussian process is an infinite dimensional extension of a Gaussian distribution, a Dirichlet process is an infinite dimensional extension of a Dirichlet distribution, which is a multivariate distribution, generating K samples $\{x_i\}$ constrained such that $0 < x_i < 1$ and

$$\sum_{i=1}^{K} x_i = 1$$

- The distribution depends on a vector of concentration parameters

$$\vec{\alpha} = (\alpha_1, \ldots, \alpha_K)$$

- and has pdf

$$p(x) = \frac{1}{B(\vec{\alpha})} \prod_{i=1}^{K} x_i^{\alpha_i - 1}, \qquad \text{where } B(\vec{\alpha}) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma\left(\sum_{j=1}^{K} \alpha_j\right)}.$$

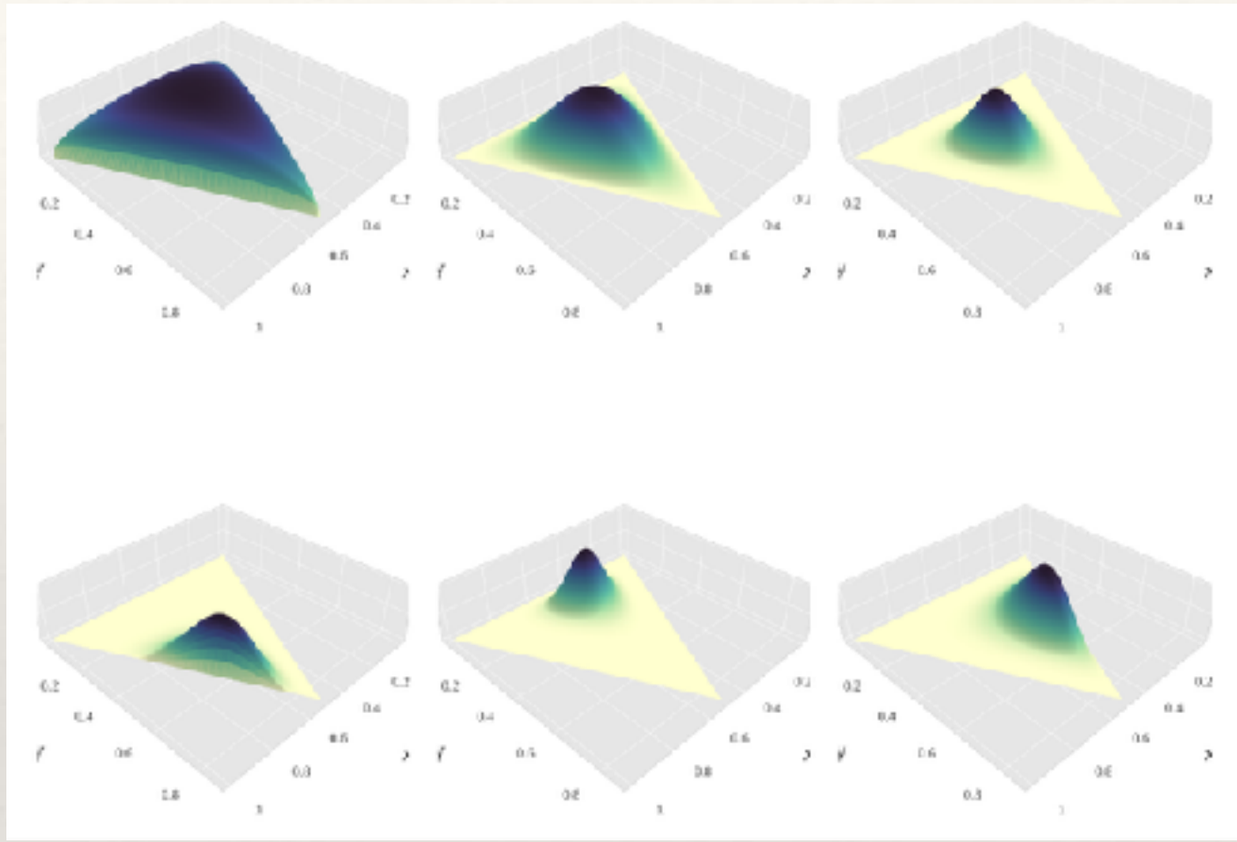


Figure from *Wikipedia*

Dirichlet processes

- The constraint that the sum of the $\{x_i\}$'s is unity means that realisations of a Dirichlet distribution are discrete probability distributions. Realisations of a Dirichlet process are continuous probability distributions.

- A Dirichlet process, X, is determined by a *base distribution*, P, and a *concentration parameter*, a, and is written $X \sim DP(P, a)$.

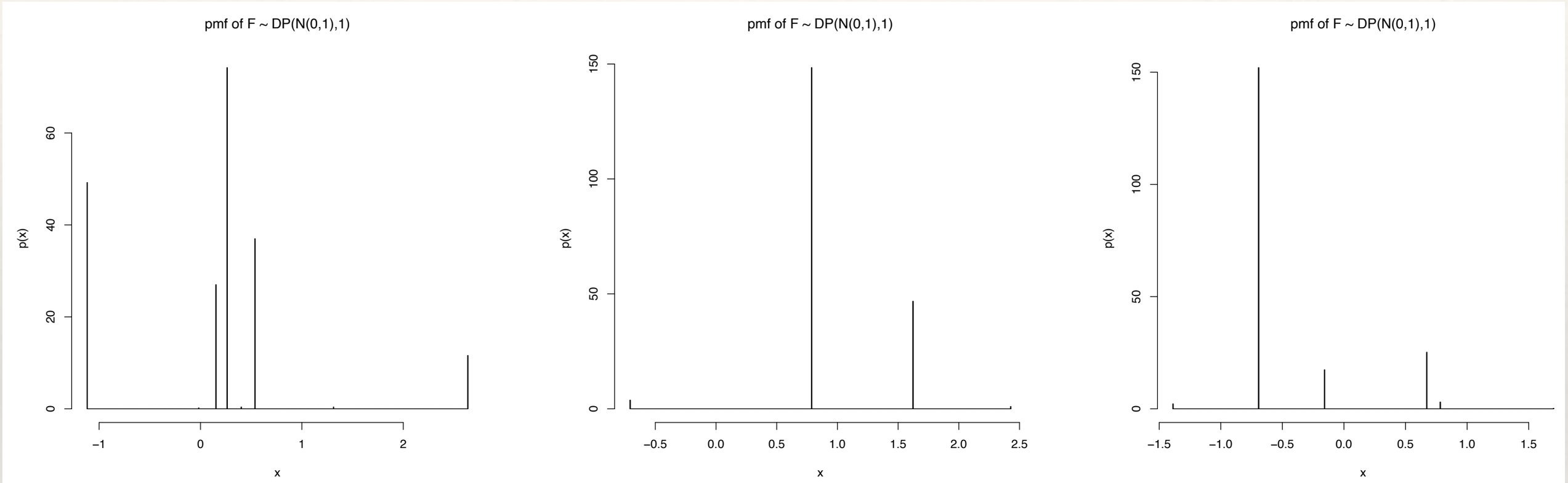
- Formally, X is a Dirichlet process on the set S if for any measurable finite partition of S, $S = \{B_i\}_{i=1}^{n}$, the probability distribution generated by a realisation of X is

$$(X(B_1), X(B_2), \ldots, X(B_n)) \sim \mathrm{Dir}(aP(B_1), aP(B_2), \ldots, aP(B_n))$$

- For small a the realised distributions become increasingly discretized, and are concentrated at a small number of points.

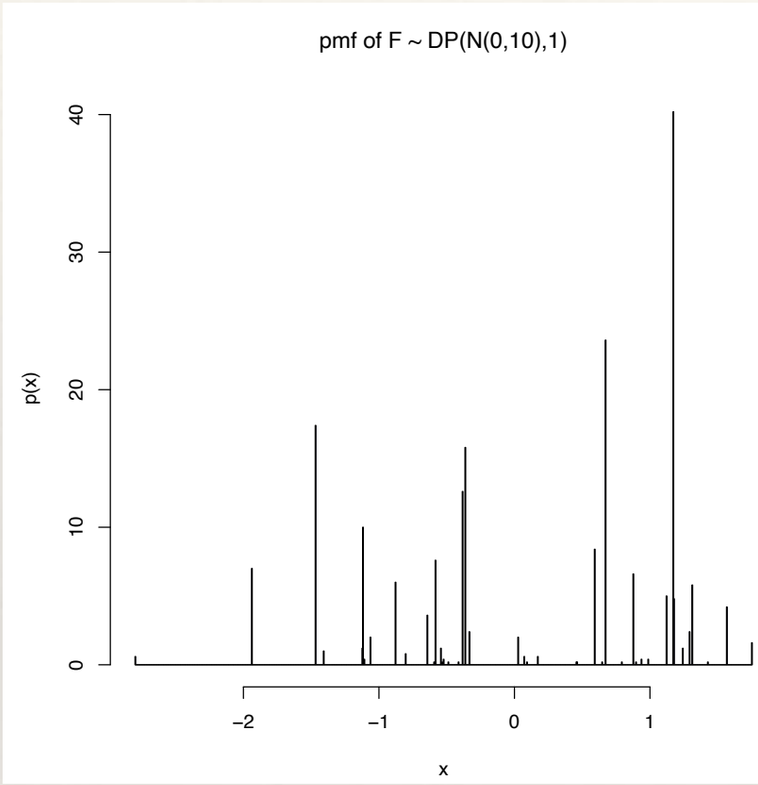
- For large a the realised distributions become increasingly continuous and close to the base distribution.

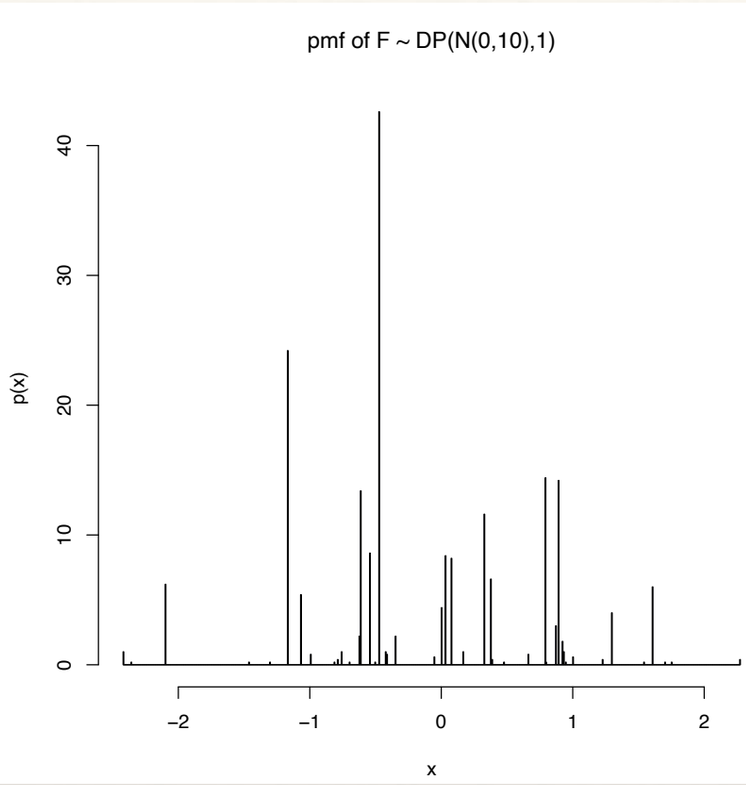
Dirichlet processes: example draws

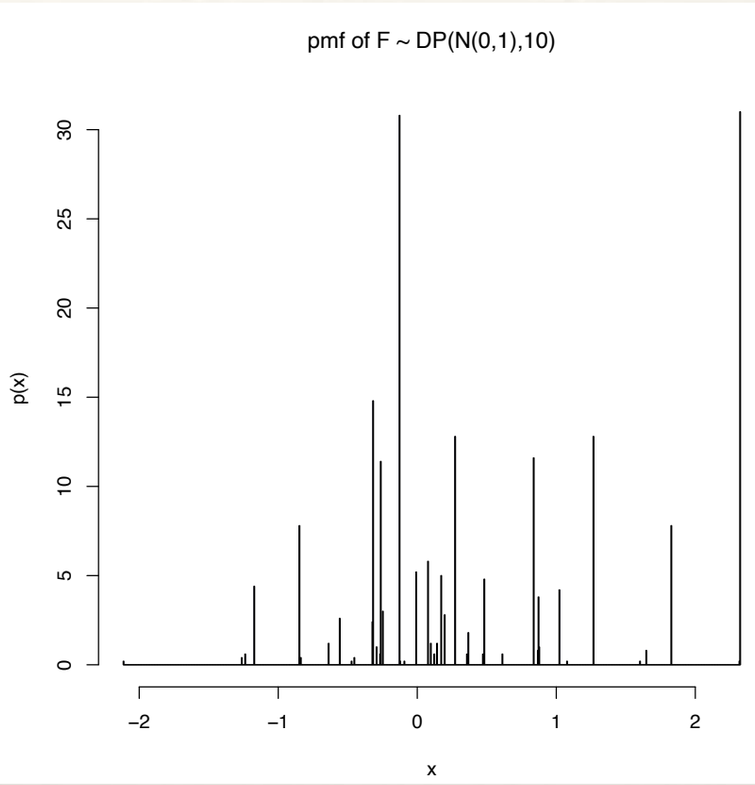


X ~ DP(N(0,1),1)

Dirichlet processes: example draws

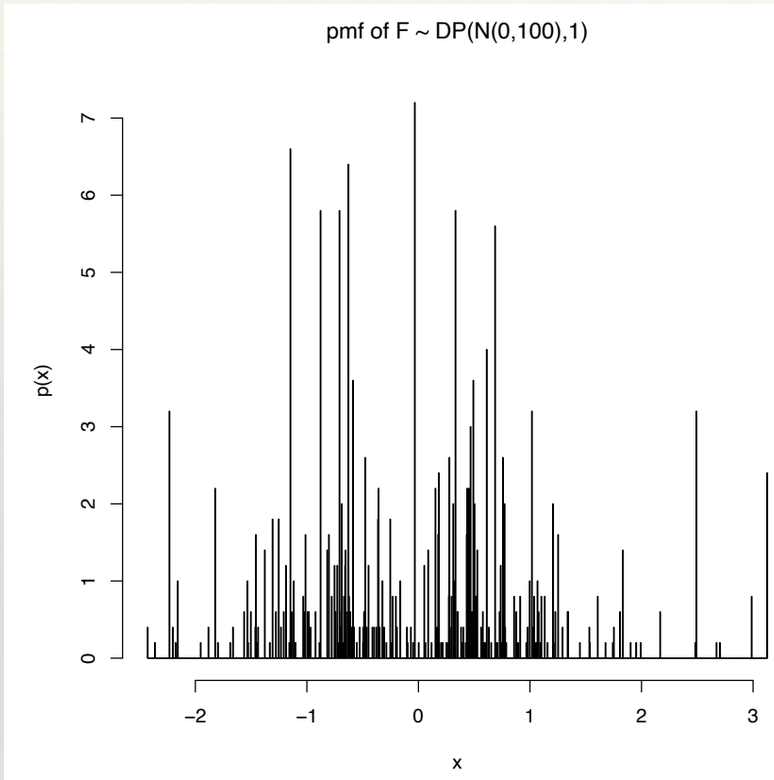


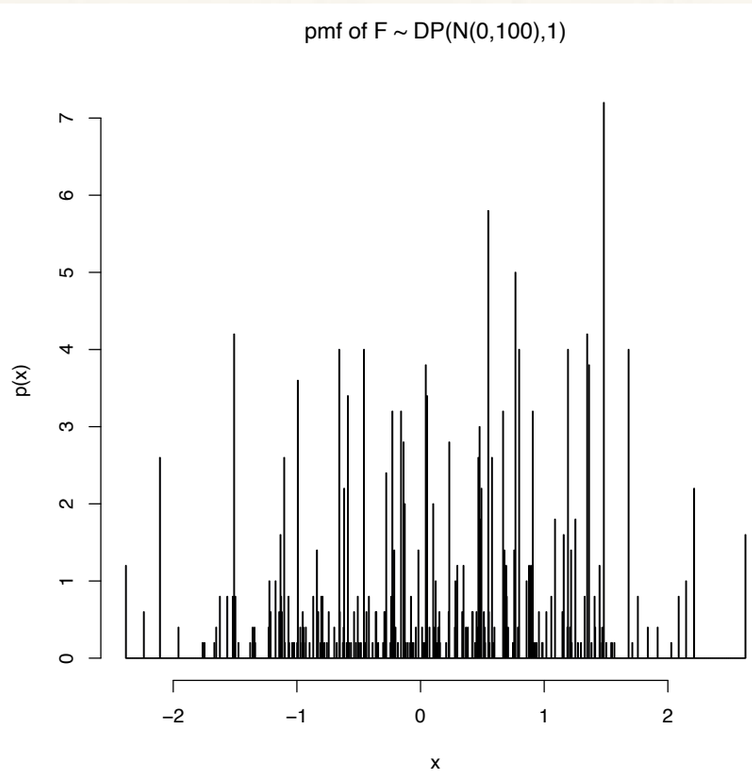


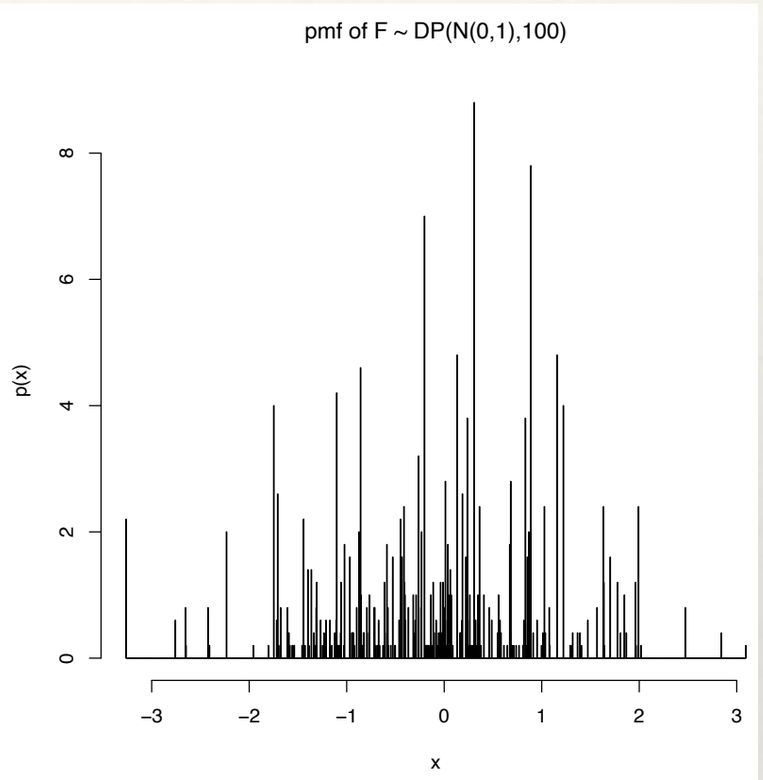


X ~ DP(N(0,1),10)

Dirichlet processes: example draws

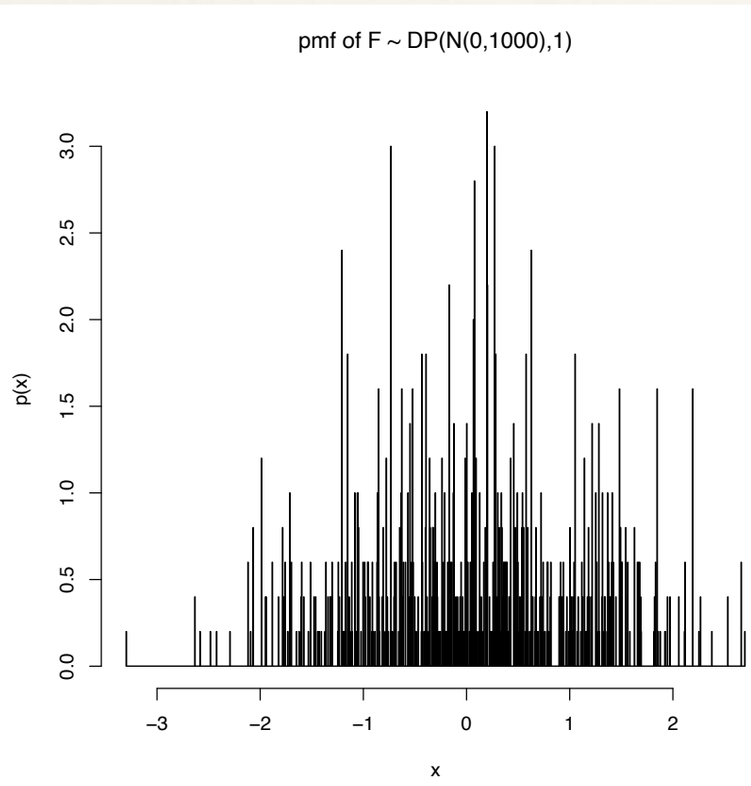


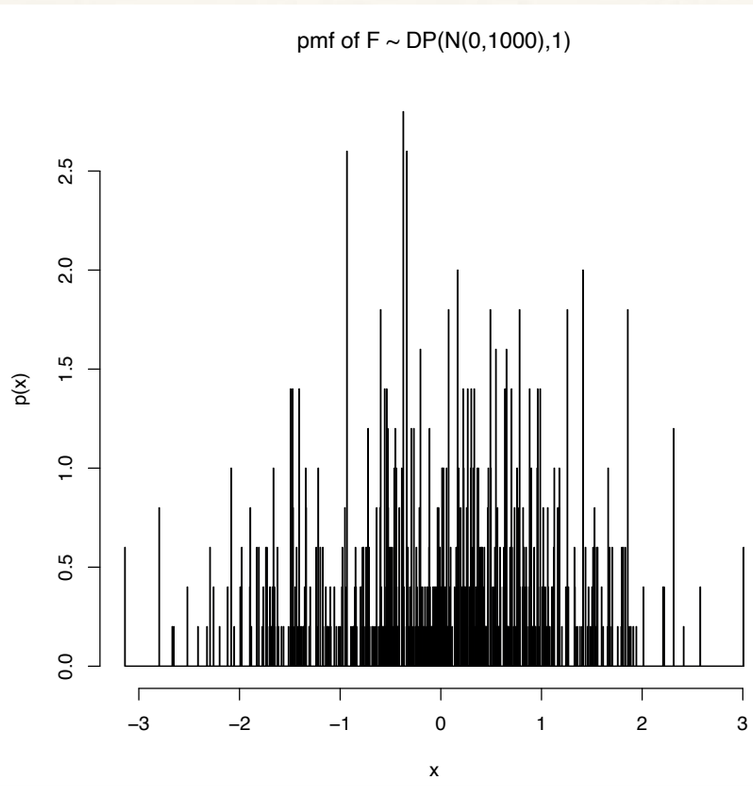


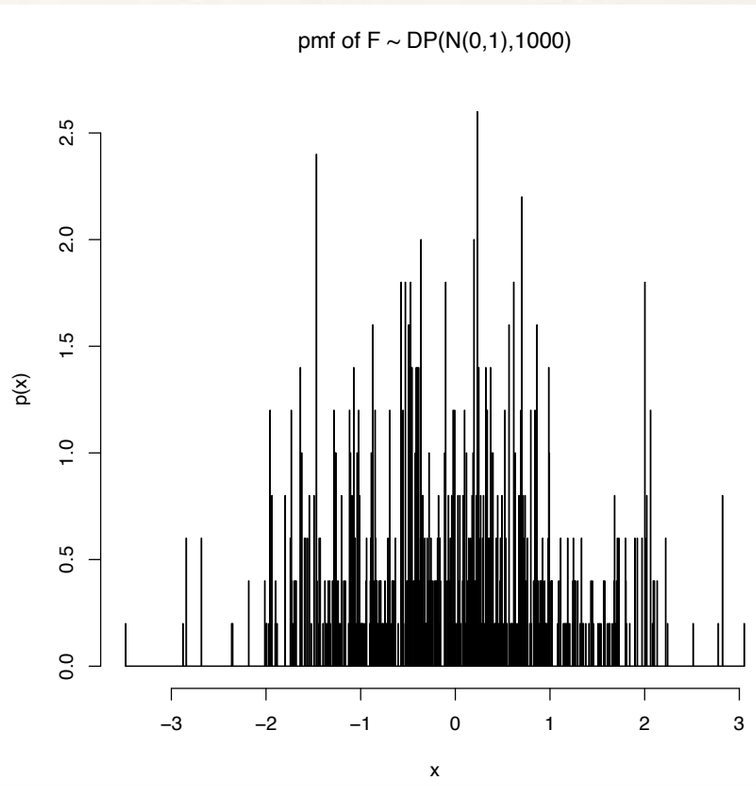


X ~ DP(N(0,1),100)

Dirichlet processes: example draws

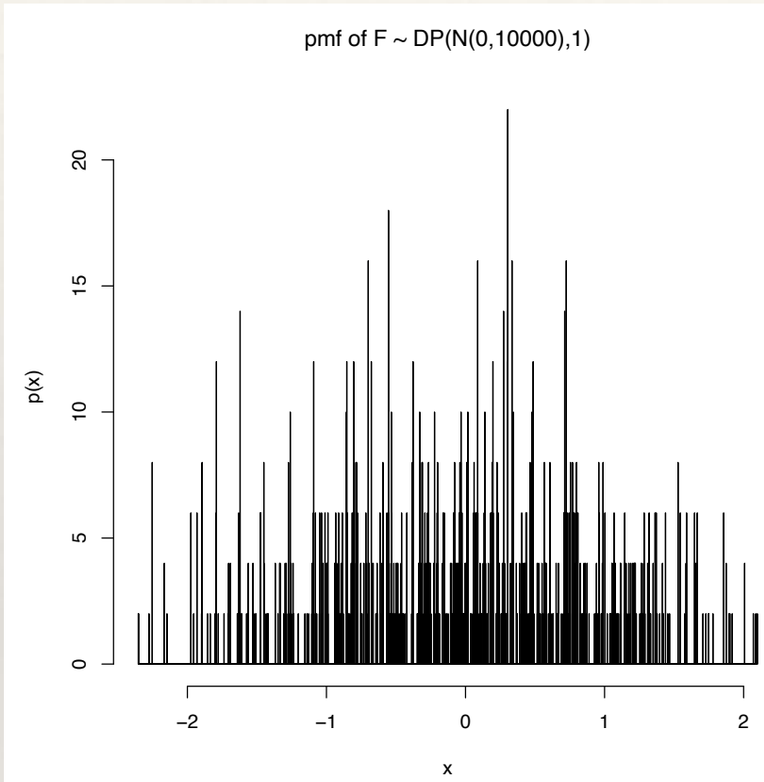


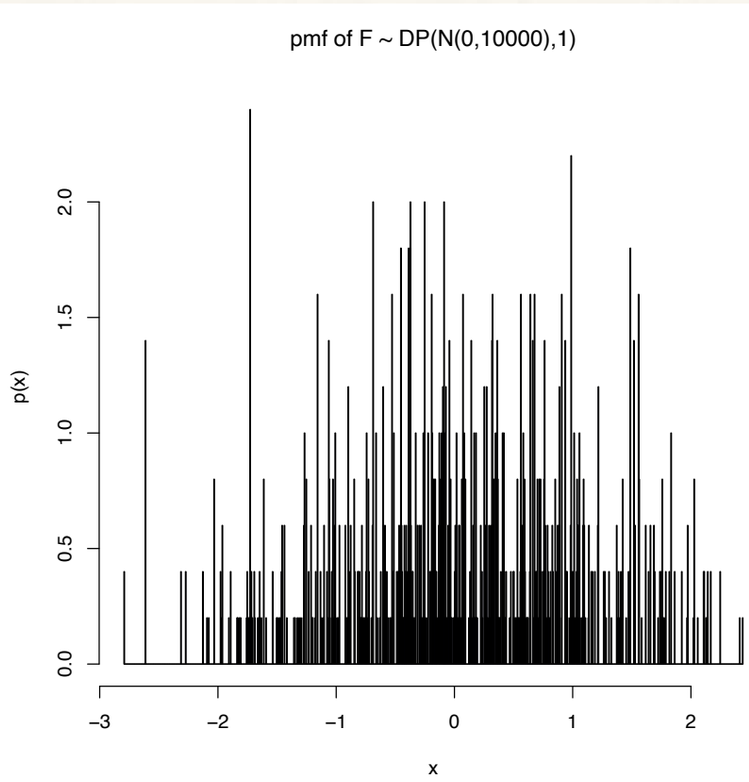


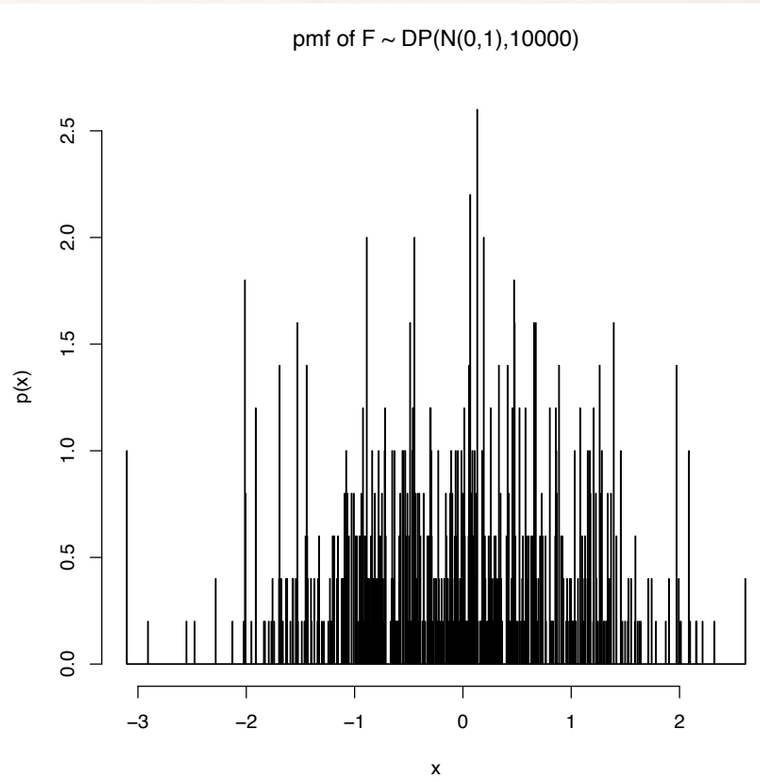


X ~ DP(N(0,1),1000)

Dirichlet processes: example draws







X ~ DP(N(0,1),10000)

Sampling Dirichlet processes

- In practice, we rarely need the full probability distribution corresponding to a particular realisation of the Dirichlet process, but a sequence of samples, $\{x_i\}$, from it. The distribution can be reconstructed from the set of samples if needed.

- There are several algorithms that can do this. One is the **Chinese restaurant process**.

- The thought experiment for the Chinese restaurant process is a restaurant with an infinite number of tables, each with an infinite number of seats. As each customer arrives, they may sit at an empty table, or join an existing table. The probability of joining an existing table is proportional to the number of people already at the table.

- To generate $DP(P,a)$ from this process, x_1 is simulated from P. For $n > 1$:

  - with probability $a/(a+n-1)$ draw x_n from P;

  - with probability $n_x/(a+n-1)$, set $x_n = x$, where n_x is the number of previous observations of x.

- The **Polya-Urn** construction is closely related to this procedure.

Stick breaking construction

- An alternative way to generate a realisation of a Dirichlet process is through the **stick-breaking** construction.
- The construction is based on successively breaking a stick into an arbitrarily large number of pieces, with break points determined by a Beta*(1,a)* distribution. The lengths of the stick segments then provide weights for point masses, at locations drawn from the base distribution.

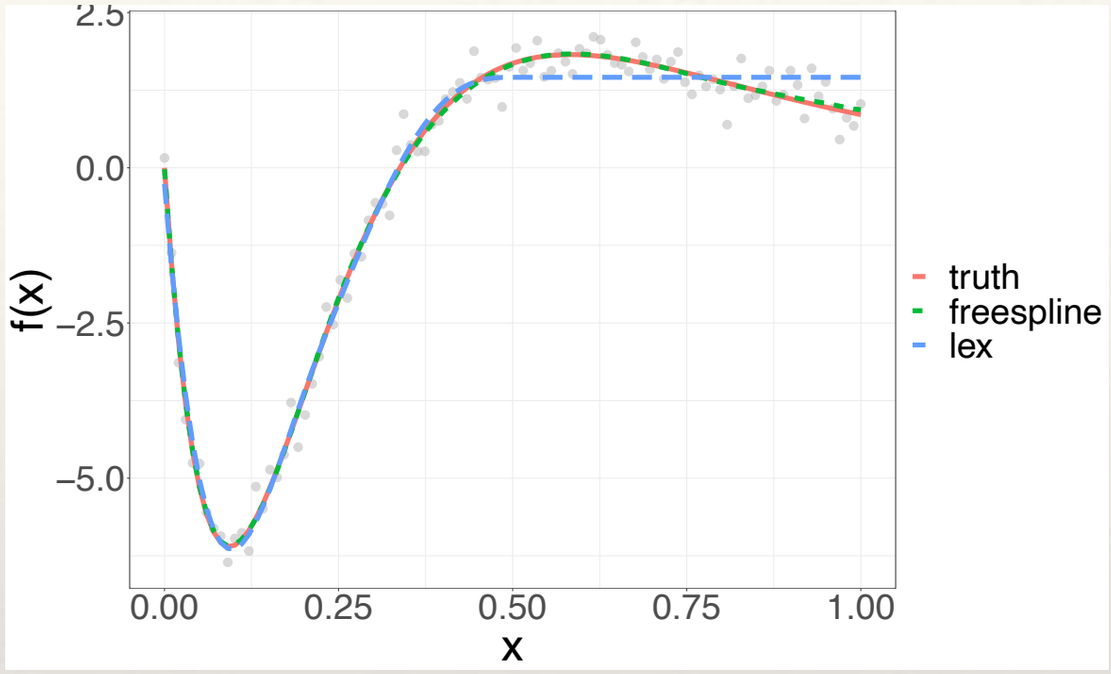
$$H = \left(\sum_{l=1}^{L_H} p_l \delta_{U_l}\right) + \left(1 - \sum_{l=1}^{L_H} p_l\right) \delta_{U_0}$$

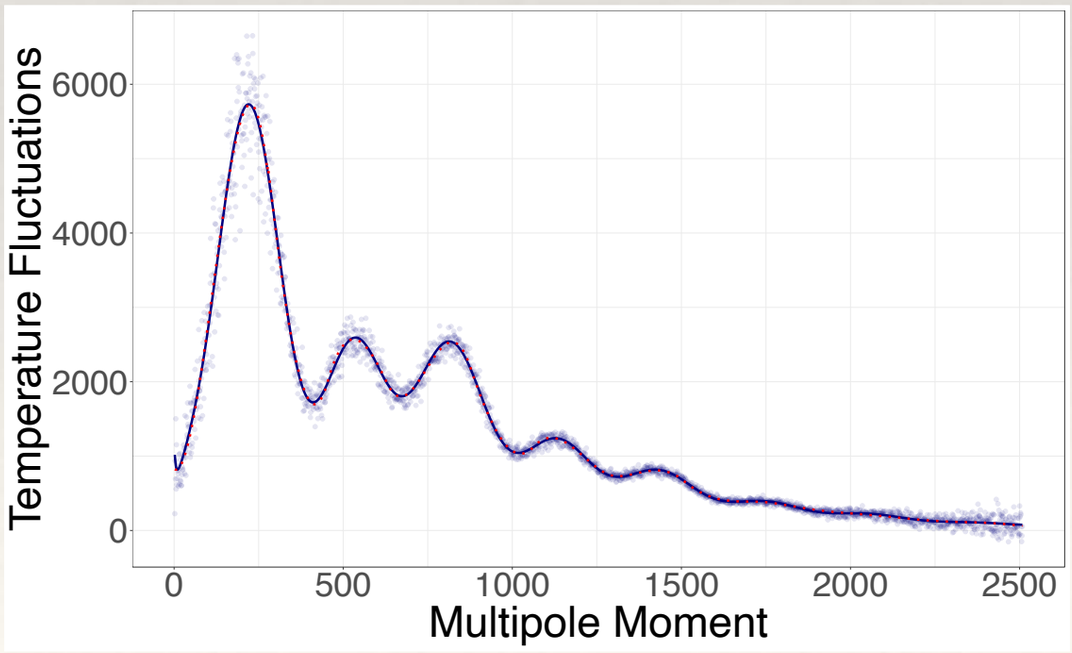
$$p_1 = V_1, \qquad p_l = \left(\prod_{j=1}^{l-1}(1 - V_j)\right) V_l, \;\; l \geq 2, \qquad p_0 = 1 - \sum_{l=1}^{L_H} p_l$$

$$V_l \sim \text{Beta}(1, a), \;\; l = 1, \ldots, L_H, \qquad U_l \sim P, \;\; l = 0, 1, \ldots, L_H,$$

Dirichlet process: applications

- Widely used in Bayesian non-parametric statistics as a prior for probability distributions.
- **Example**: B-spline knot locations. Use Dirichlet process as a prior for locations of knots for non-parametric B-spline regression (Edwards & JG 2020).
- Applied analysis to obtain non-parametric fit to Planck data.
- Exploring application to population inference and noise modelling for GW detectors.





Applications: Gravitational Waves

- Del Pozzo et al. (2018) used a **Dirichlet Process Gaussian mixture model** to represent source localisation posteriors generated by LALInference.
- The Dirichlet process was used to generate the centres of a set of Gaussian kernels that were superimposed used to represent the true posterior distribution.