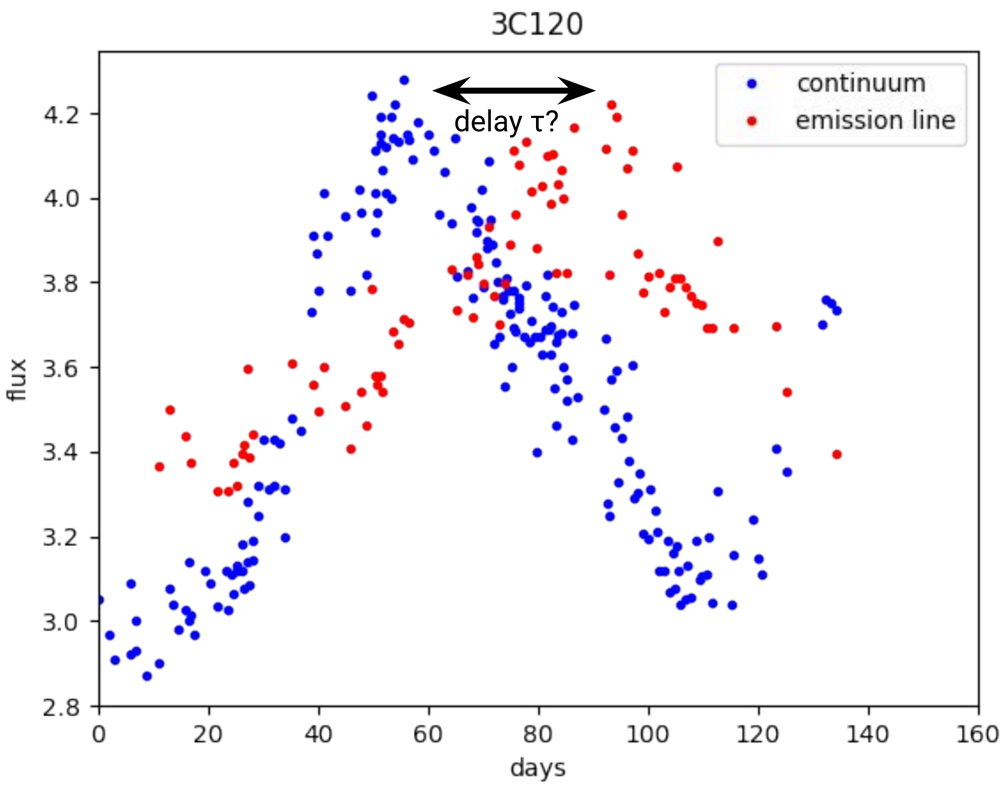
Heidelberg Institute for Theoretical Studies

Probabilistic Cross Correlation for Delay Estimation

Nikos Gianniotis, Francisco Pozo Nunez, Kai Polsterer
Astroinformatics Group
https://www.h-its.org/research/ain/

Overview: what?



Find the delay τ between two lightcurves originating from an AGN

Overview: why?

- Ultimate goal is to (re)estimate the Hubble constant
  This sheds light on the evolution of the universe

- Estimation of black hole masses in AGN is crucial for this estimation
  Delay τ between lightcurves (+ assumptions) helps us infer black hole mass in AGN

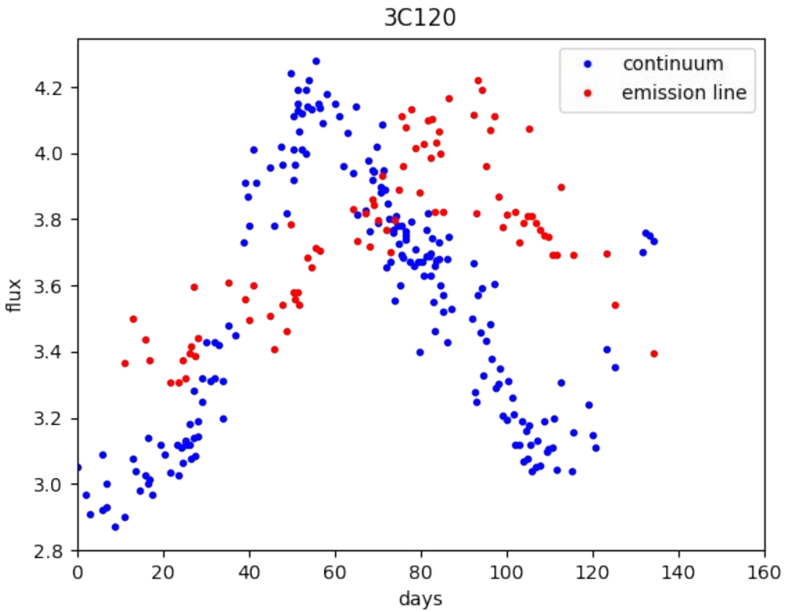
Overview: how?

- Interpolated Cross-Correlation Function (and co) estimates delay, but doesn't propagate the uncertainty (measurement noise) present in the data
    - we're uncertain about the data ⇒ delay estimate will be uncertain

- Probabilistic reformulation of ICCF that overcomes this issue
    - uncertainty present in the data is propagated
    - delay estimate described by probability distribution

- We demonstrate our method on object 3C120

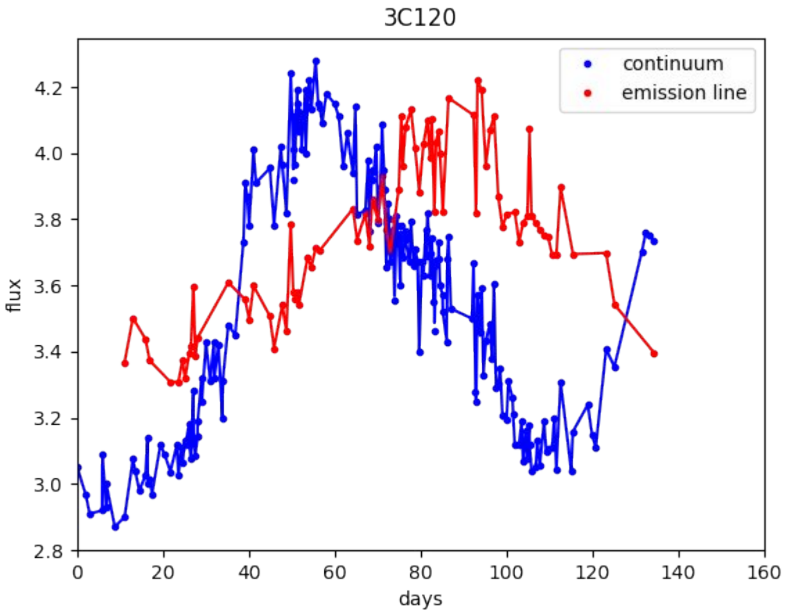
Interpolated Cross Correlation Function



Relies on (linear) interpolation and cross-correlation $$(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$$

[Gaskell & Peterson (1987)]

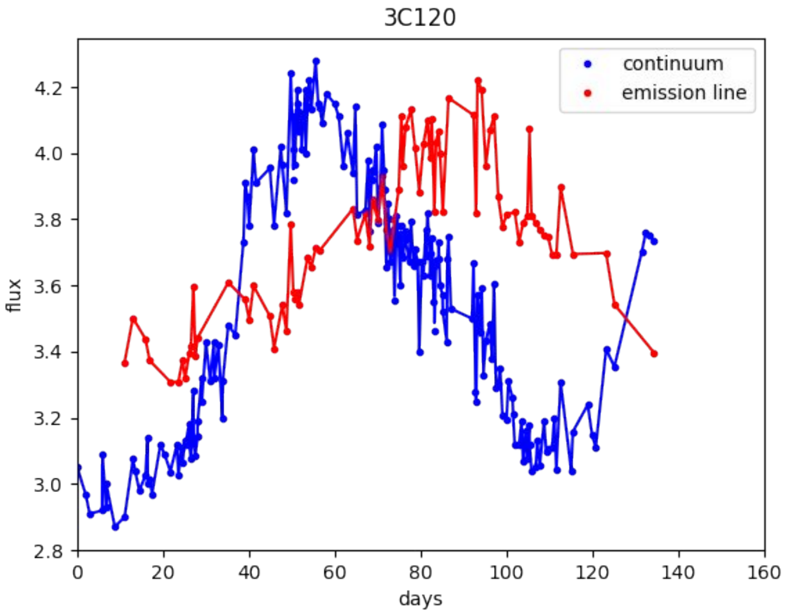
Interpolated Cross Correlation Function

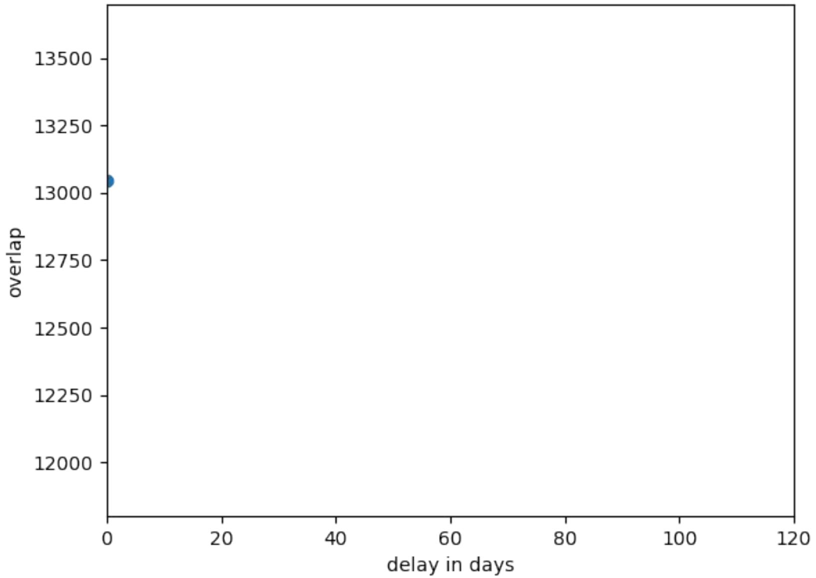


Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

Interpolated Cross Correlation Function

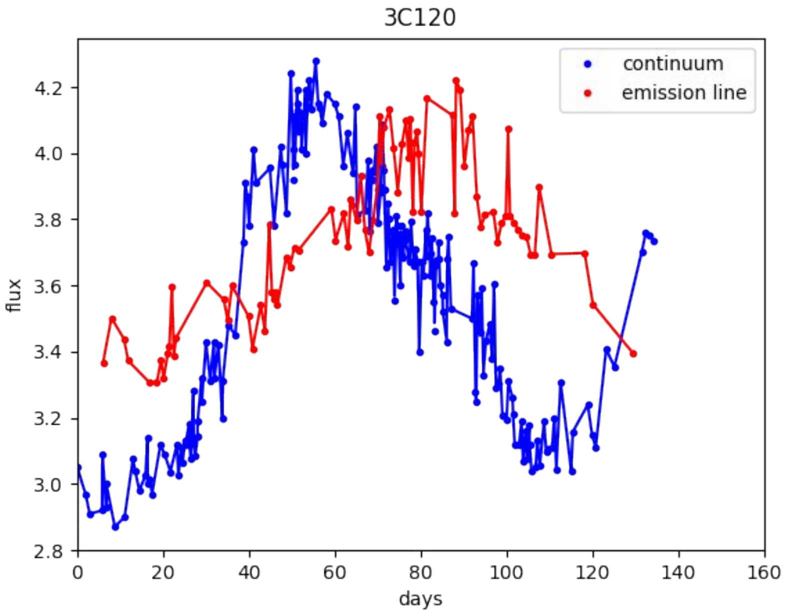


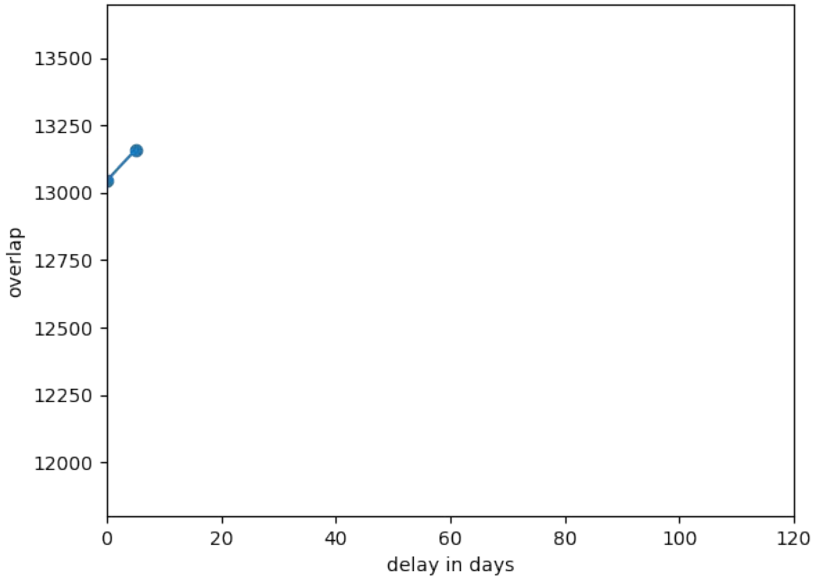


Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

Interpolated Cross Correlation Function

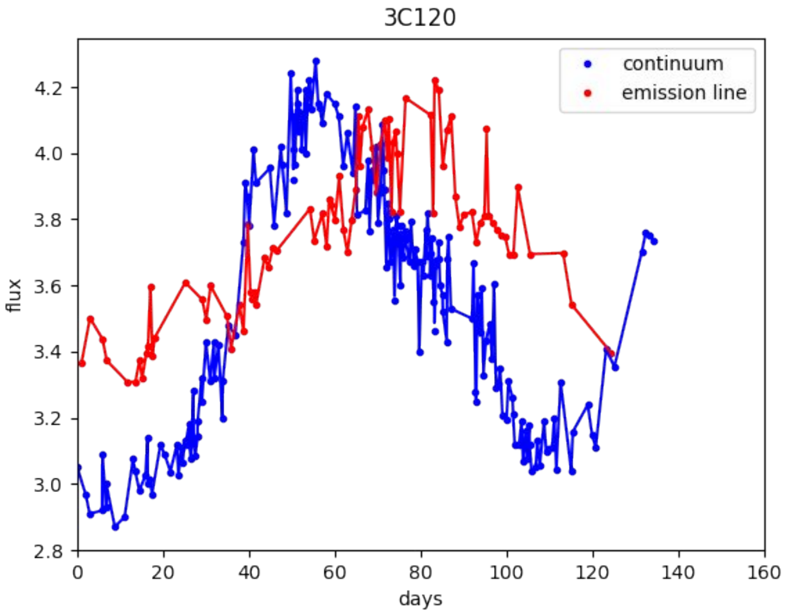


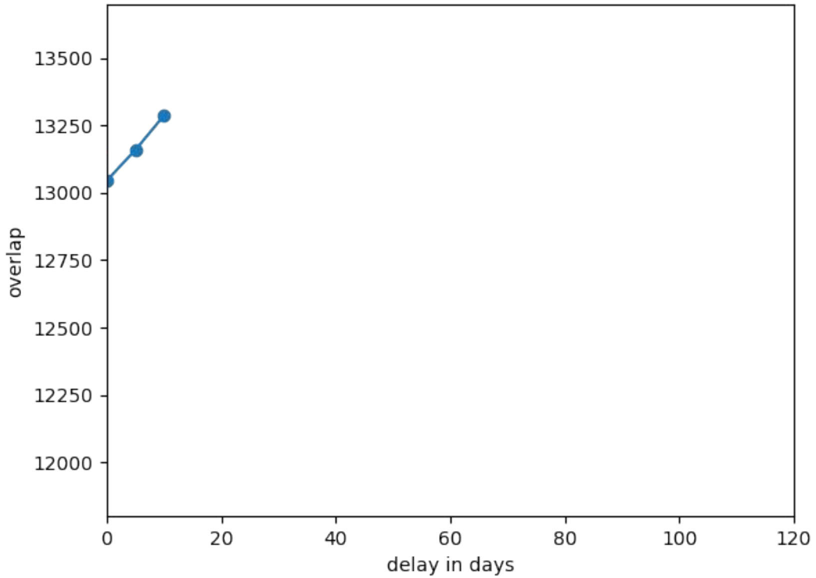


Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

Interpolated Cross Correlation Function

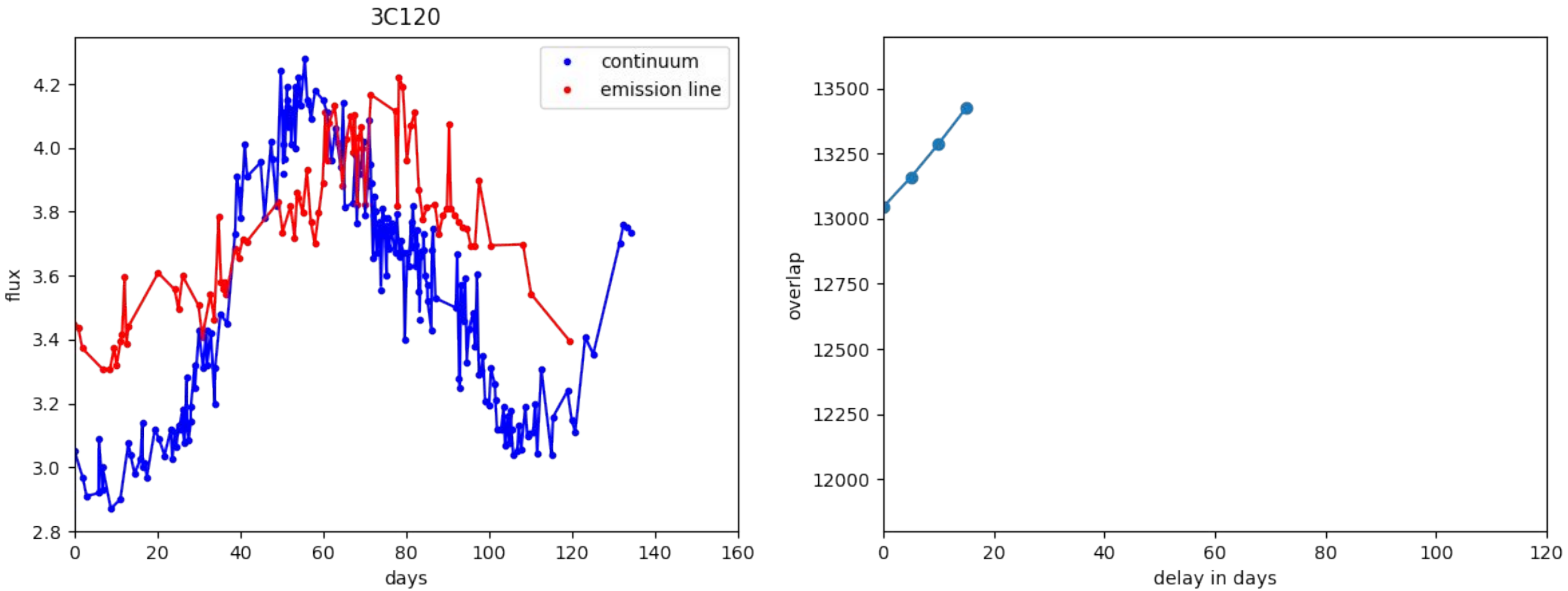




Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

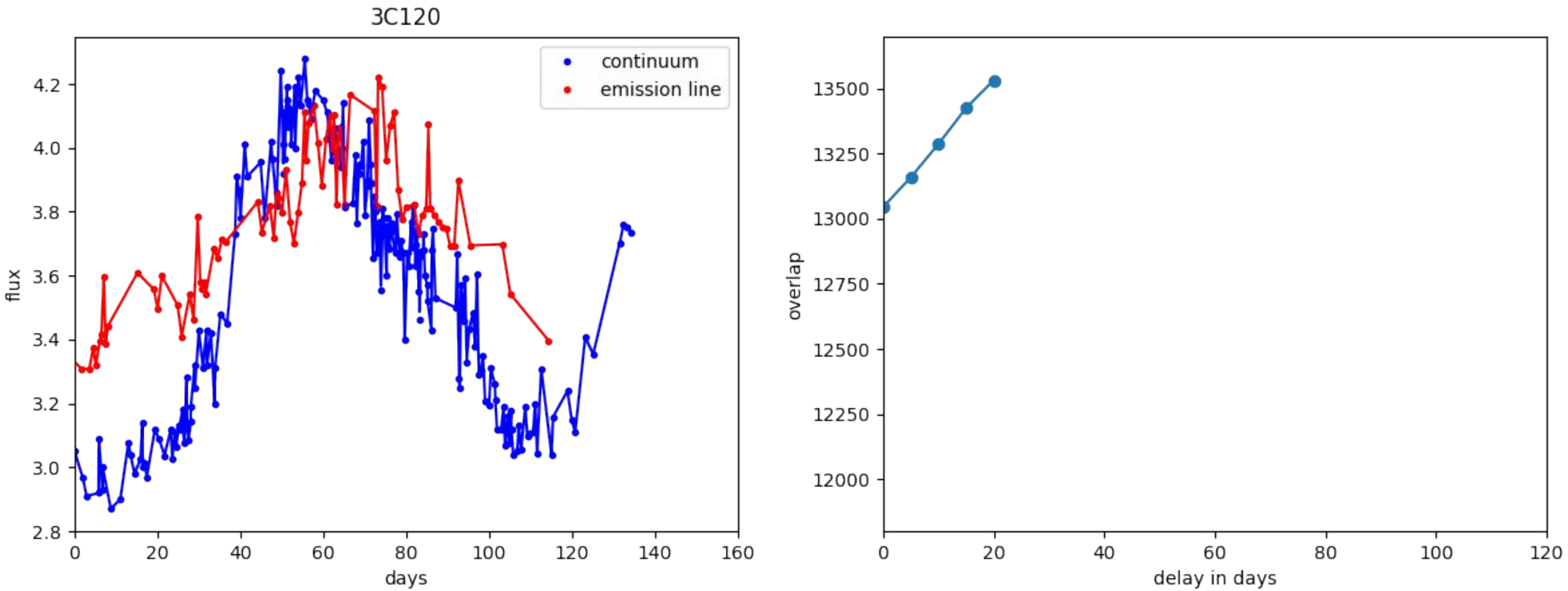
Interpolated Cross Correlation Function



Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

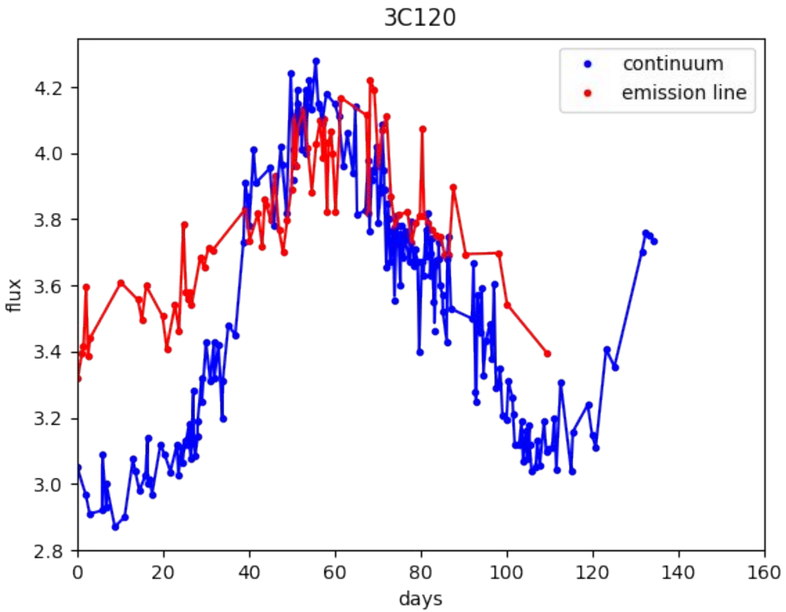
Interpolated Cross Correlation Function

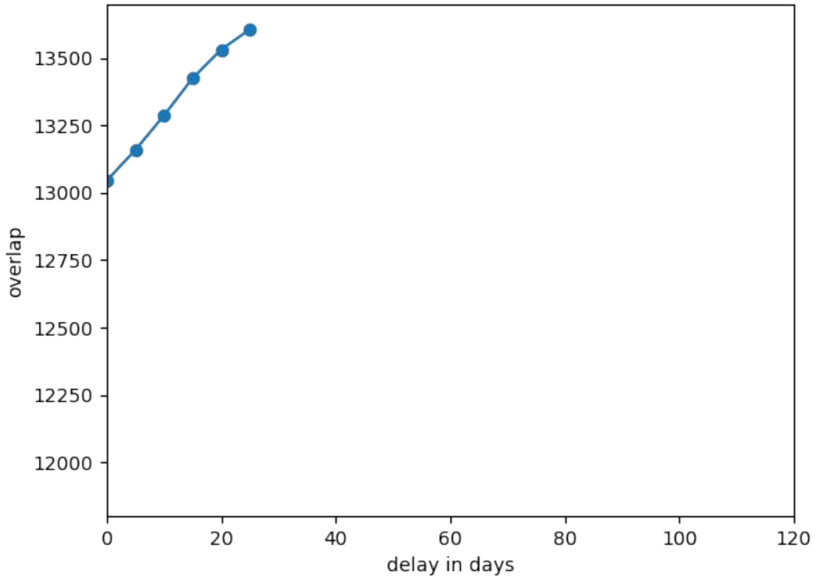


Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

Interpolated Cross Correlation Function

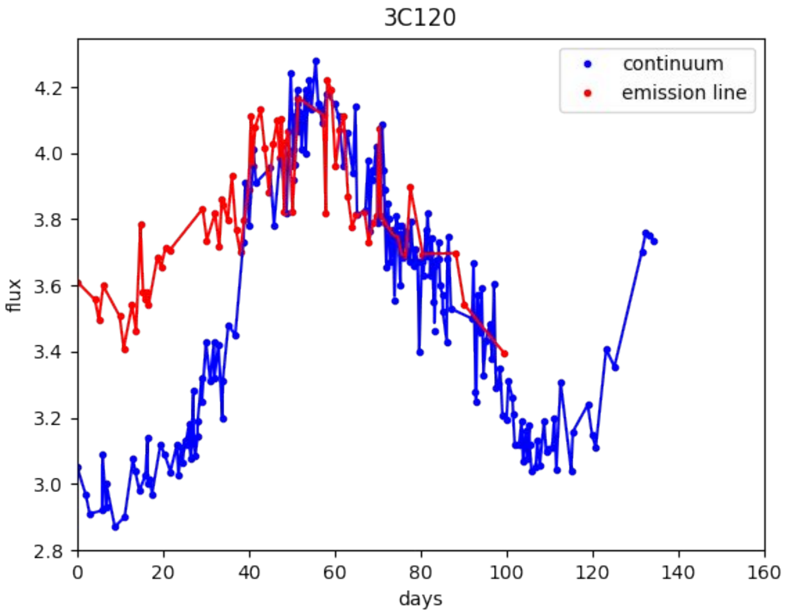


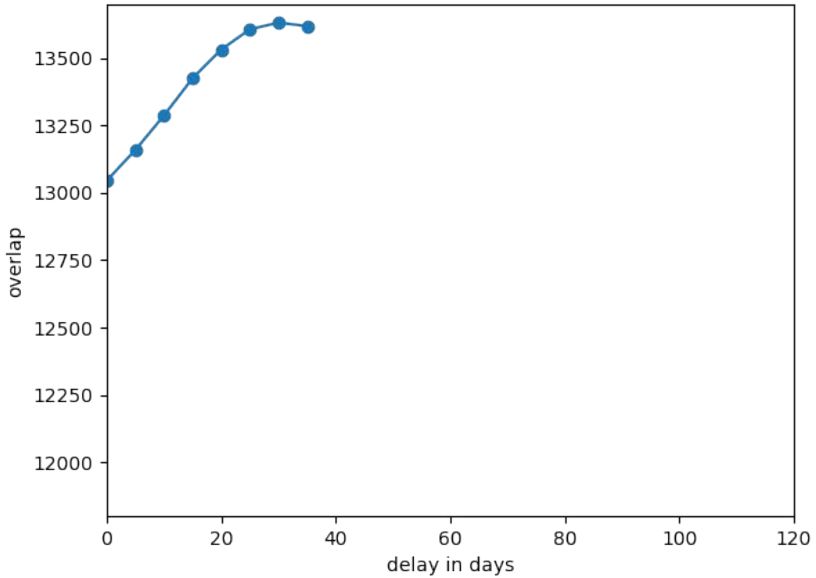


Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

Interpolated Cross Correlation Function

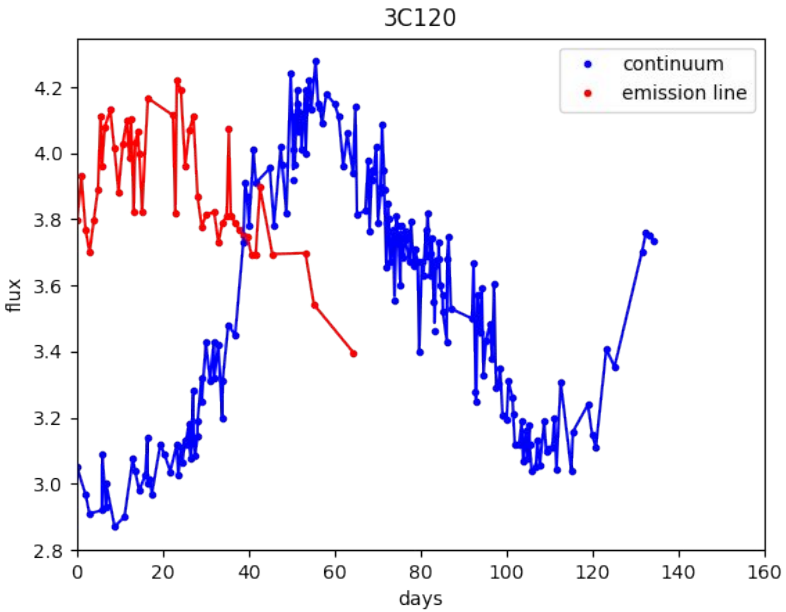


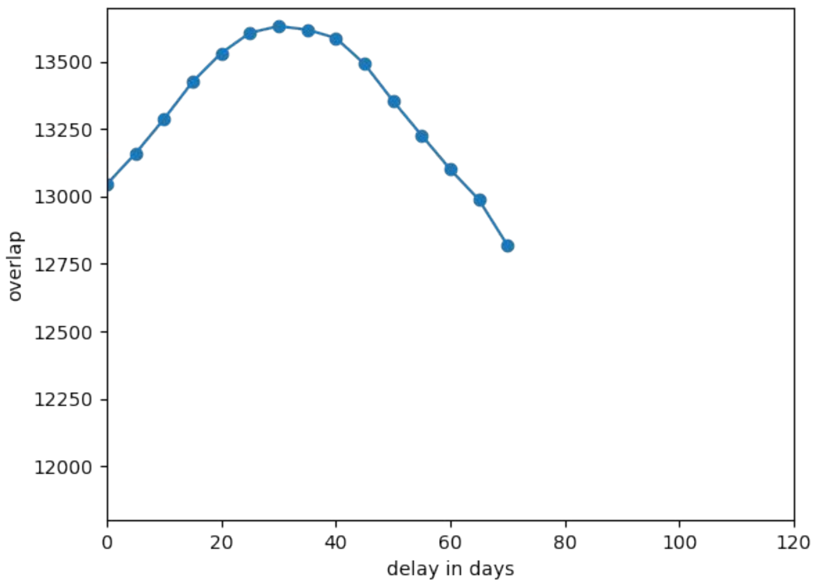


Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

Interpolated Cross Correlation Function

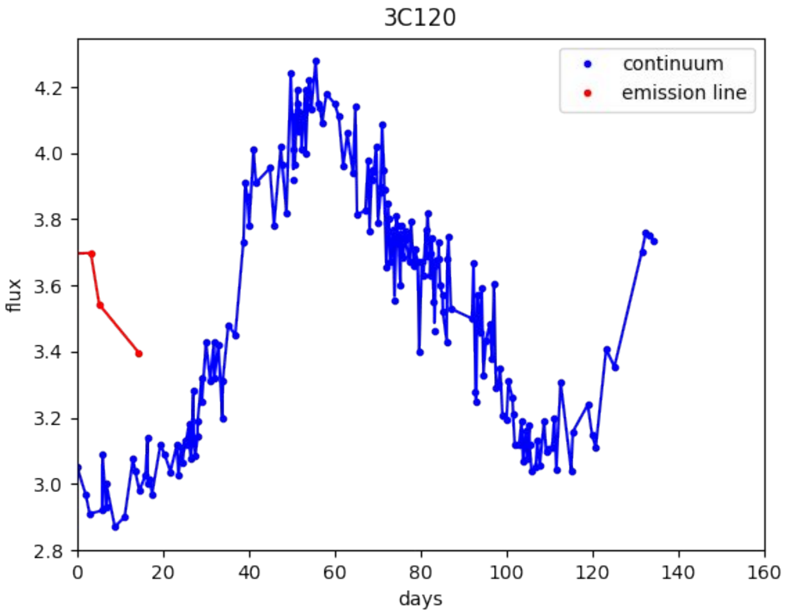


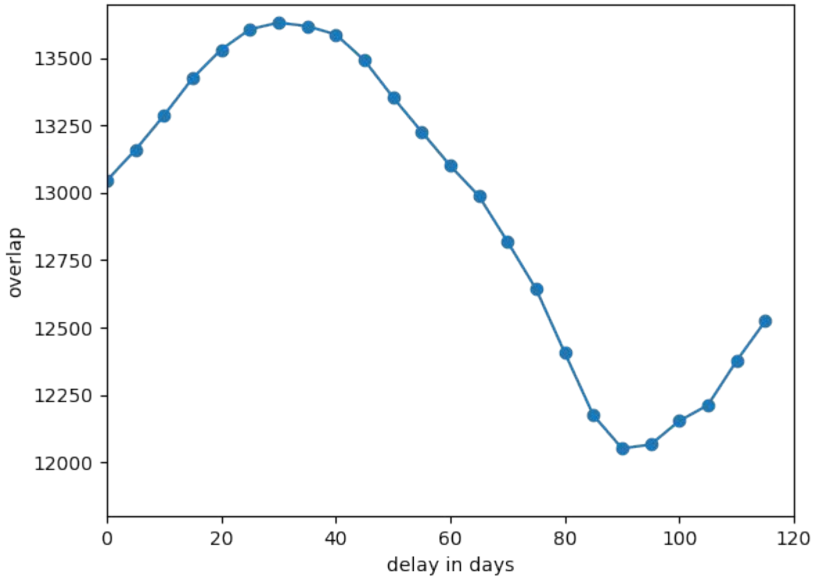


Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

Interpolated Cross Correlation Function

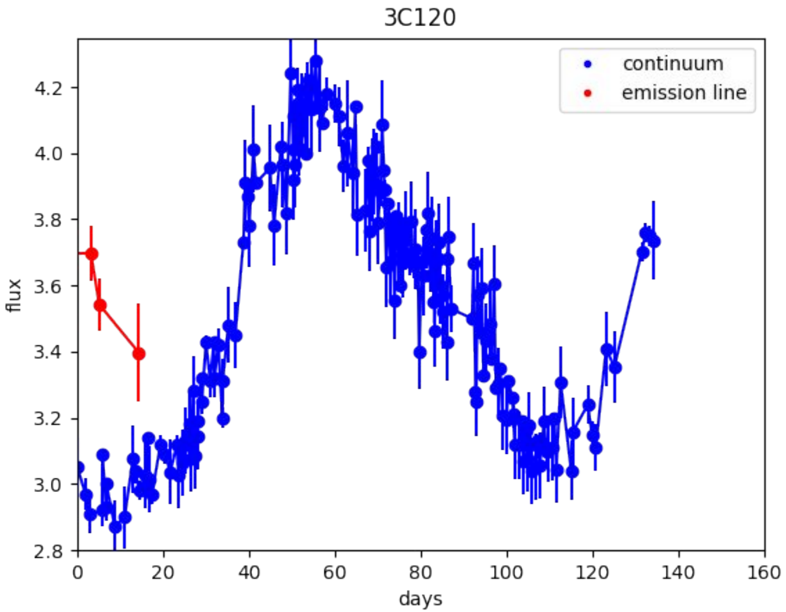


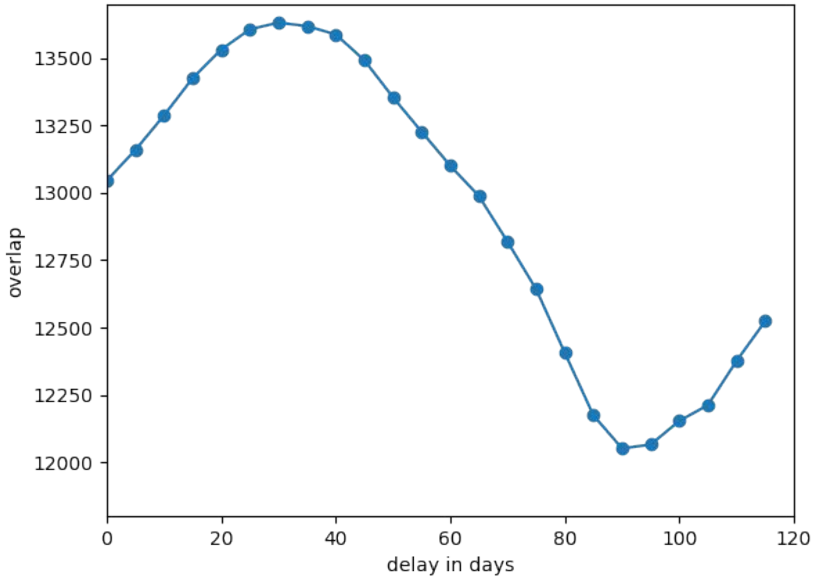


Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

Interpolated Cross Correlation Function





Relies on (linear) interpolation and cross-correlation $(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$

[Gaskell & Peterson (1987)]

ICCF does not propagate noise uncertainty

Assumptions behind ICCF

- ICCF finds delay via

$$(y_c \star y_e)(\tau) = \int_{-\infty}^{\infty} y_c(t) y_e(t+\tau) dt$$

- If we accept **cross-correlation** as a means to find the delay, then we assume that one lightcurve is a **delayed**, **scaled** and **offsetted** version of the other

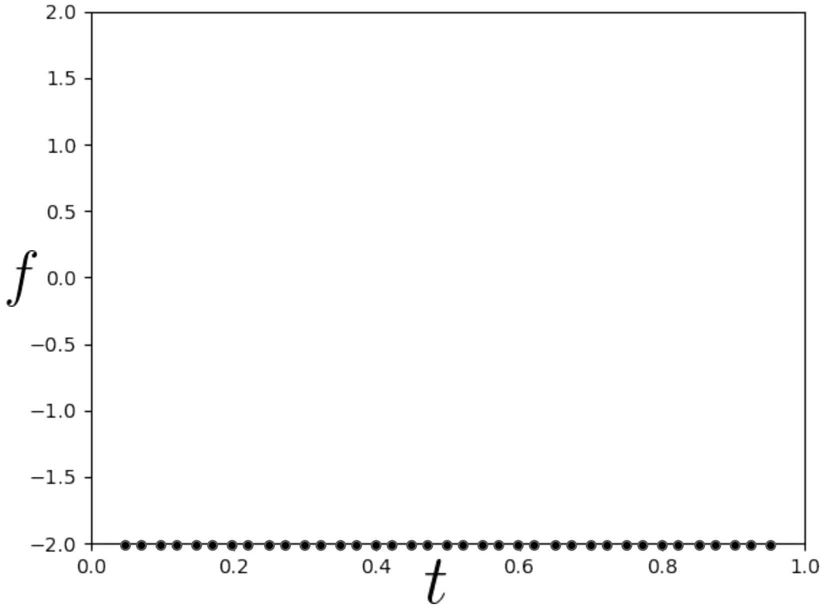
$$y_e(t-\tau) = \alpha y_c(t) + b$$

- We can rewrite this relationship in terms of an unobserved, latent signal f(t)

$$y_e(t) = \alpha_e f(t-\tau) + b_e$$
$$y_c(t) = \alpha_c f(t) + b_c$$

Gaussian Process Modelling

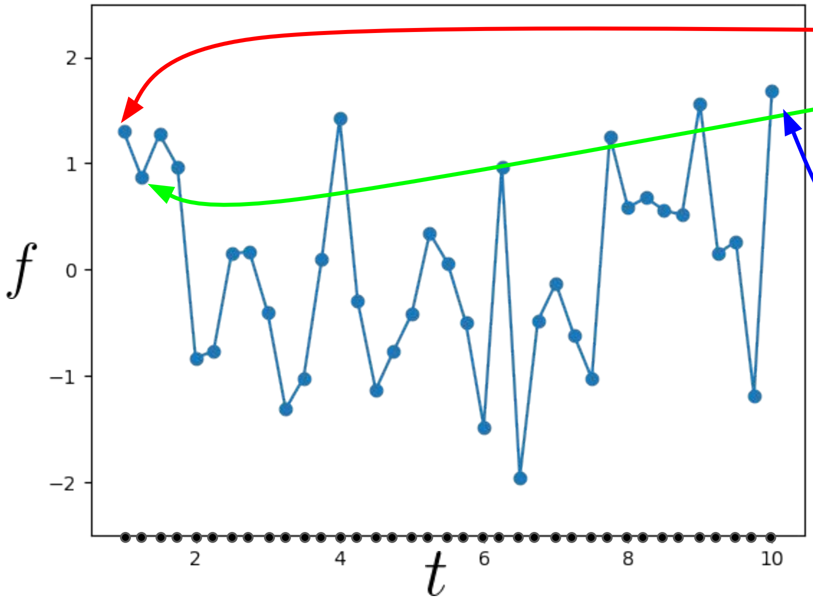
- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the values of a function as a Gaussian distribution
- A GP is a tool to model distributions of functions



$$f = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{bmatrix} \right)$$

Gaussian Process Modelling

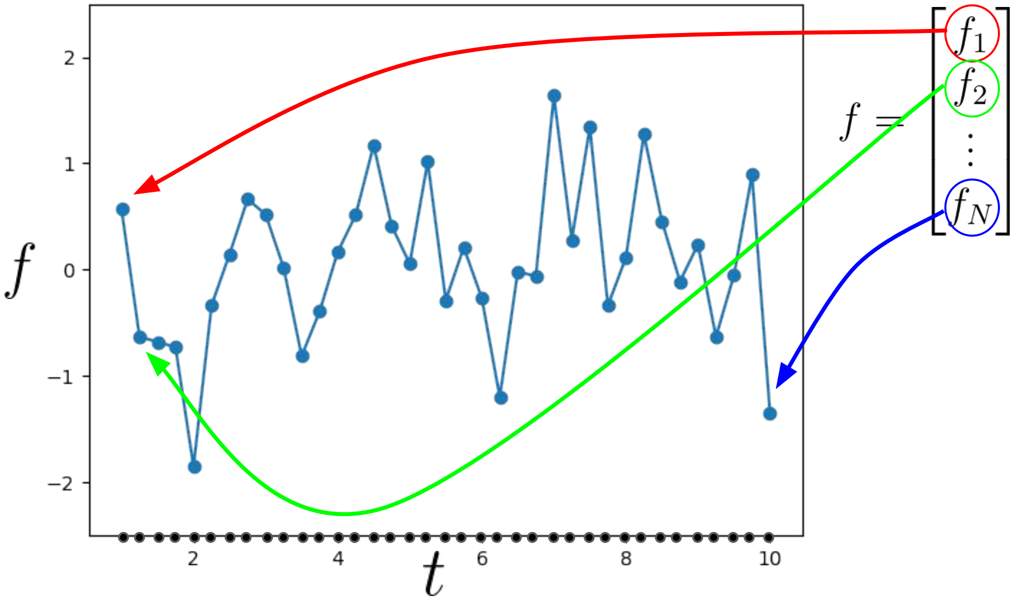
- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions



$$f = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{bmatrix} \right)$$

Gaussian Process Modelling

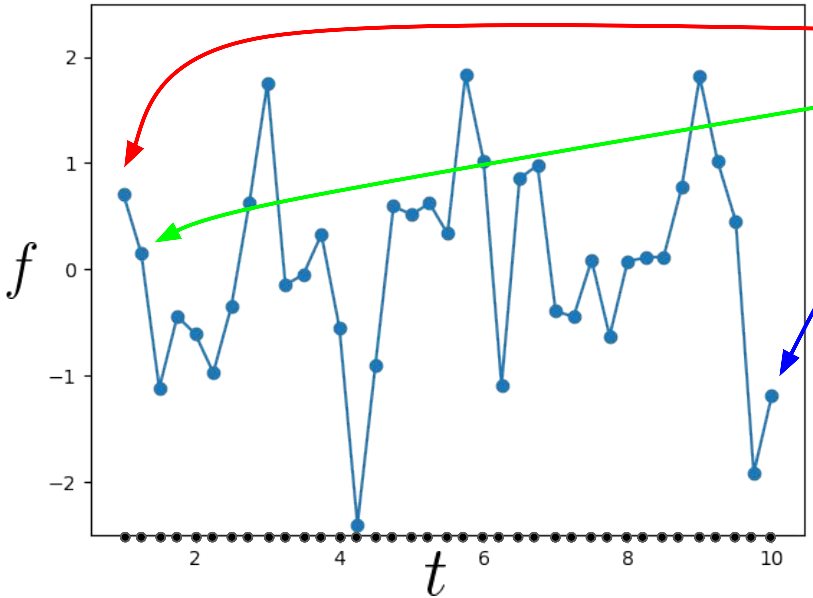
- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions



$$f = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{bmatrix} \right)$$

Gaussian Process Modelling

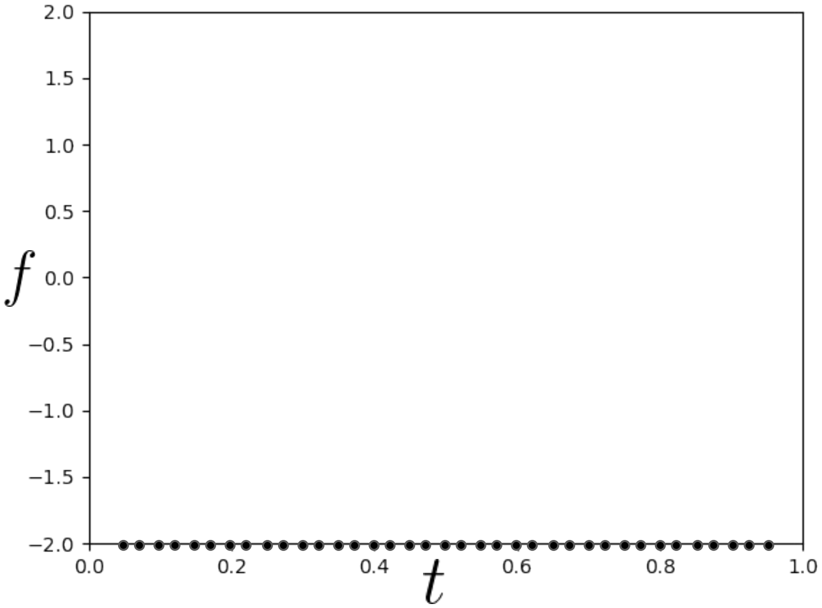
- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions



$$f = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{bmatrix} \right)$$

Gaussian Process Modelling

- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions

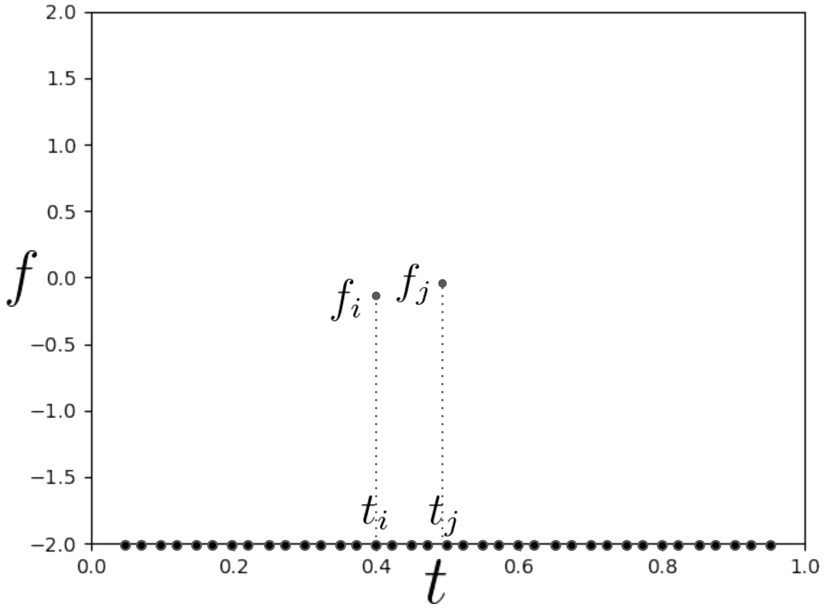


$$f \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} k(t_1, t_1) & k(t_1, t_2) & \dots & k(t_1, t_N) \\ k(t_2, t_1) & k(t_2, t_2) & \dots & k(t_2, t_N) \\ \vdots & \vdots & \dots & \vdots \\ k(t_N, t_1) & k(t_N, t_2) & \dots & k(t_N, t_N) \end{bmatrix} \right)$$

$$k(t_i, t_j) = \exp(-\|t_i - t_j\|^2)$$

Gaussian Process Modelling

- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions

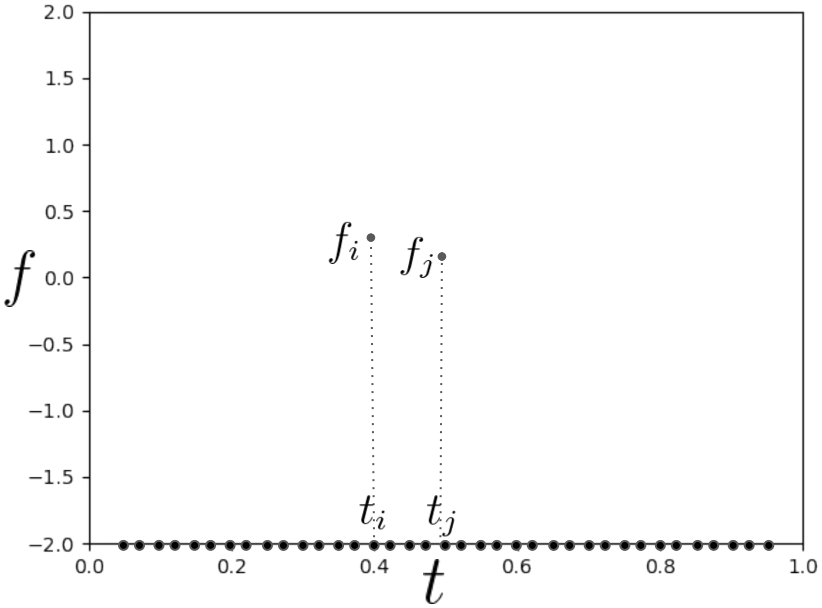


$$f \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} k(t_1,t_1) & k(t_1,t_2) & \dots & k(t_1,t_N) \\ k(t_2,t_1) & k(t_2,t_2) & \dots & k(t_2,t_N) \\ \vdots & \vdots & \dots & \vdots \\ k(t_N,t_1) & k(t_N,t_2) & \dots & k(t_N,t_N) \end{bmatrix} \right)$$

$$k(t_i, t_j) = \exp(-\|t_i - t_j\|^2)$$

Gaussian Process Modelling

- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions

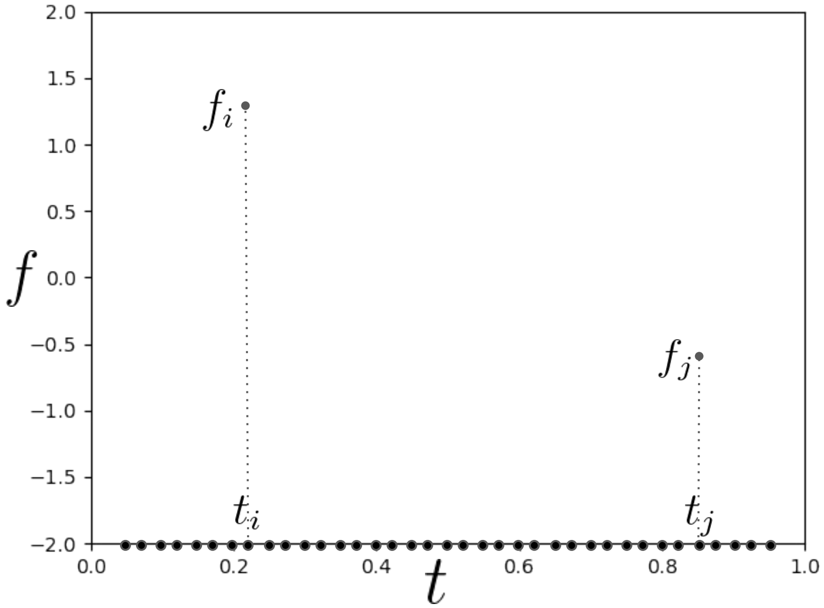


$$f \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} k(t_1,t_1) & k(t_1,t_2) & \dots & k(t_1,t_N) \\ k(t_2,t_1) & k(t_2,t_2) & \dots & k(t_2,t_N) \\ \vdots & \vdots & \dots & \vdots \\ k(t_N,t_1) & k(t_N,t_2) & \dots & k(t_N,t_N) \end{bmatrix}\right)$$

$$k(t_i, t_j) = \exp(-\|t_i - t_j\|^2)$$

Gaussian Process Modelling

- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions

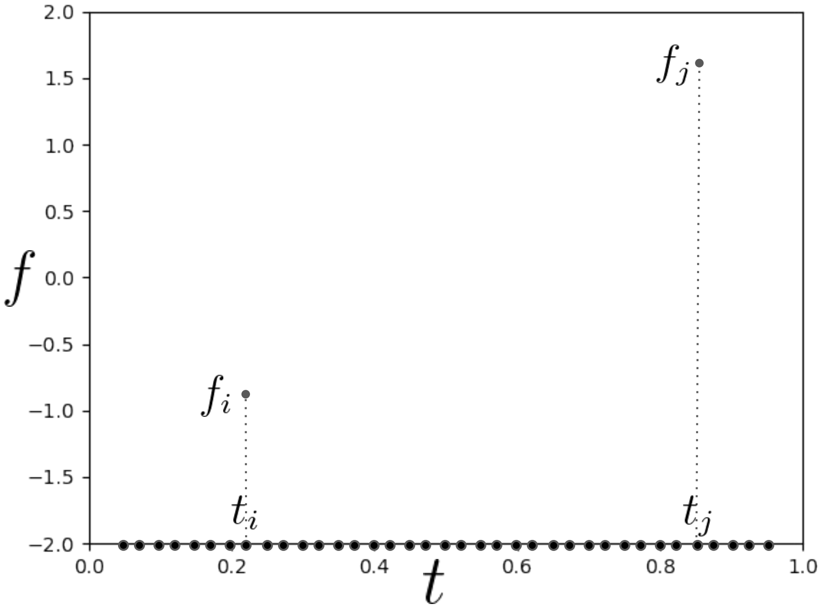


$$f \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} k(t_1,t_1) & k(t_1,t_2) & \dots & k(t_1,t_N) \\ k(t_2,t_1) & k(t_2,t_2) & \dots & k(t_2,t_N) \\ \vdots & \vdots & \dots & \vdots \\ k(t_N,t_1) & k(t_N,t_2) & \dots & k(t_N,t_N) \end{bmatrix} \right)$$

$$k(t_i, t_j) = \exp(-\|t_i - t_j\|^2)$$

Gaussian Process Modelling

- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions

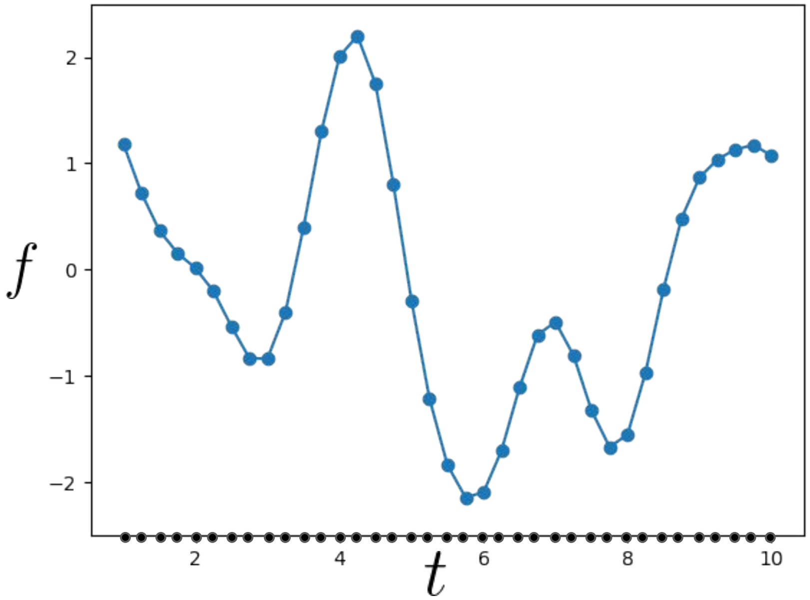


$$f \sim \mathcal{N}\left(\begin{bmatrix}0\\ \vdots \\ 0\end{bmatrix}, \begin{bmatrix} k(t_1,t_1) & k(t_1,t_2) & \dots & k(t_1,t_N) \\ k(t_2,t_1) & k(t_2,t_2) & \dots & k(t_2,t_N) \\ \vdots & \vdots & \dots & \vdots \\ k(t_N,t_1) & k(t_N,t_2) & \dots & k(t_N,t_N) \end{bmatrix}\right)$$

$$k(t_i, t_j) = \exp(-\|t_i - t_j\|^2)$$

Gaussian Process Modelling

- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions

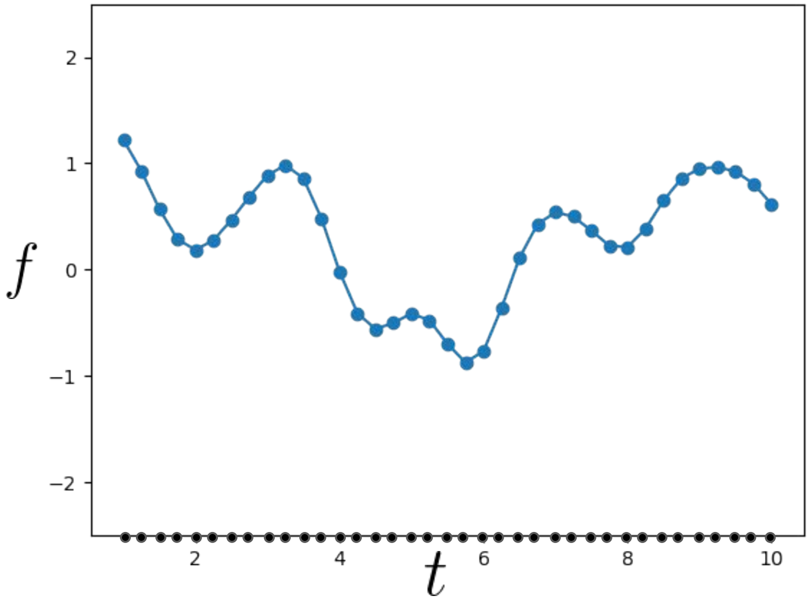


$$f \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} k(t_1,t_1) & k(t_1,t_2) & \dots & k(t_1,t_N) \\ k(t_2,t_1) & k(t_2,t_2) & \dots & k(t_2,t_N) \\ \vdots & \vdots & \dots & \vdots \\ k(t_N,t_1) & k(t_N,t_2) & \dots & k(t_N,t_N) \end{bmatrix} \right)$$

$$k(t_i, t_j) = \exp(-\|t_i - t_j\|^2)$$

Gaussian Process Modelling

- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions

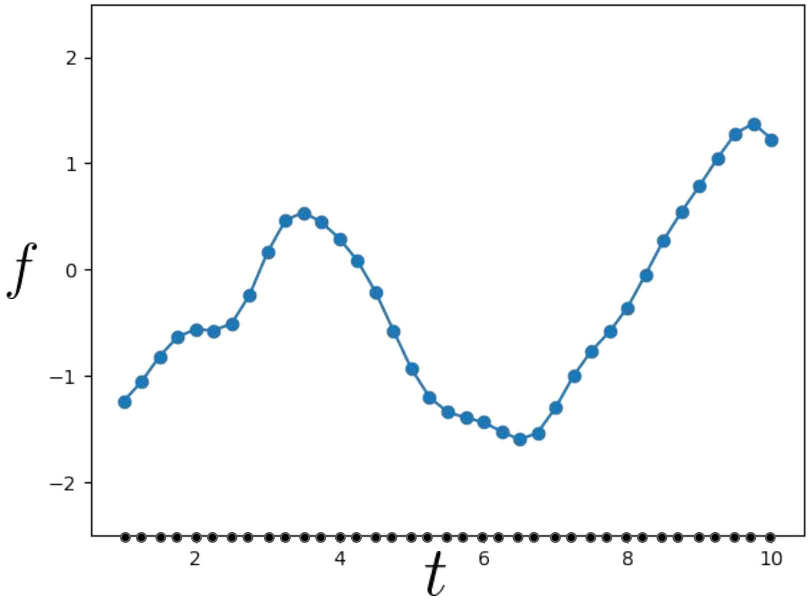


$$f \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} k(t_1, t_1) & k(t_1, t_2) & \dots & k(t_1, t_N) \\ k(t_2, t_1) & k(t_2, t_2) & \dots & k(t_2, t_N) \\ \vdots & \vdots & \dots & \vdots \\ k(t_N, t_1) & k(t_N, t_2) & \dots & k(t_N, t_N) \end{bmatrix} \right)$$

$$k(t_i, t_j) = \exp(-\|t_i - t_j\|^2)$$

Gaussian Process Modelling

- We model latent signal $f(t)$ as a sample from a Gaussian process (GP)
- A GP models the joint distribution of function values as a Gaussian distribution
- A GP is a tool to model distributions of functions



$$f \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} k(t_1, t_1) & k(t_1, t_2) & \dots & k(t_1, t_N) \\ k(t_2, t_1) & k(t_2, t_2) & \dots & k(t_2, t_N) \\ \vdots & \vdots & \dots & \vdots \\ k(t_N, t_1) & k(t_N, t_2) & \dots & k(t_N, t_N) \end{bmatrix} \right)$$

$$k(t_i, t_j) = \exp(-\|t_i - t_j\|^2)$$

Gaussian Process Modelling

- Remember, Gaussian distribution is closed under affine transformations

$$\begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \sigma_1^2 & \sigma_{1,2}^2 \\ \sigma_{2,1}^2 & \sigma_2^2 \end{bmatrix}\right) \longrightarrow \begin{bmatrix} \alpha_1 f_1 + b_1 \\ \alpha_2 f_2 + b_2 \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \alpha_1 \mu_1 + b_1 \\ \alpha_2 \mu_2 + b_2 \end{bmatrix}, \begin{bmatrix} \alpha_1^2 \sigma_1^2 & \alpha_1 \alpha_2 \sigma_{1,2}^2 \\ \alpha_2 \alpha_1 \sigma_{2,1}^2 & \alpha_2^2 \sigma_2^2 \end{bmatrix}\right)$$

- GP also closed under affine transformation

$$f \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} k(t_1, t_1) & k(t_1, t_2) & \dots & k(t_1, t_N) \\ k(t_2, t_1) & k(t_2, t_2) & \dots & k(t_2, t_N) \\ \vdots & \vdots & \dots & \vdots \\ k(t_N, t_1) & k(t_N, t_2) & \dots & k(t_N, t_N) \end{bmatrix}\right)$$

$$\downarrow$$

$$\begin{bmatrix} y_c(t_i) \\ \vdots \\ y_e(t_j) \end{bmatrix} = \begin{bmatrix} \alpha_c f(t_i) + b_c \\ \vdots \\ \alpha_e f(t_j - \tau) + b_e \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} b_c \\ \vdots \\ b_e \end{bmatrix}, \begin{bmatrix} \alpha_c^2 k(t_i, t_i) & \dots & \alpha_c \alpha_e k(t_i, t_j - \tau) \\ \vdots & \vdots & \vdots \\ \alpha_e \alpha_c k(t_j - \tau, t_i) & \dots & \alpha_e^2 k(t_j - \tau, t_j - \tau) \end{bmatrix}\right)$$

Gaussian Process Cross Correlation

- Likelihood reads

$$\begin{bmatrix} y_c(t_i) \\ \vdots \\ y_e(t_j) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} b_c \\ \vdots \\ b_e \end{bmatrix}, \begin{bmatrix} \alpha_c^2 k(t_i, t_i) & \dots & \alpha_c \alpha_e k(t_i, t_j - \tau) \\ \vdots & \vdots & \vdots \\ \alpha_e \alpha_c k(t_j - \tau, t_i) & \dots & \alpha_e^2 k(t_j - \tau, t_j - \tau) \end{bmatrix} \right)$$

Gaussian Process Cross Correlation

- Likelihood reads

$$\begin{bmatrix} y_c(t_i) \\ \vdots \\ y_e(t_j) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} b_c \\ \vdots \\ b_e \end{bmatrix}, \begin{bmatrix} \alpha_c^2 k(t_i, t_i) + \sigma_{c,i}^2 & \dots & \alpha_c \alpha_e k(t_i, t_j - \tau) \\ \vdots & \vdots & \vdots \\ \alpha_e \alpha_c k(t_j - \tau, t_i) & \dots & \alpha_e^2 k(t_j - \tau, t_j - \tau) + \sigma_{e,j}^2 \end{bmatrix} \right)$$

Gaussian Process Cross Correlation

- Likelihood reads

$$\begin{bmatrix} y_c(t_i) \\ \vdots \\ y_e(t_j) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} b_c \\ \vdots \\ b_e \end{bmatrix}, \begin{bmatrix} \alpha_c^2 k(t_i, t_i) + \sigma_{c,i}^2 & \dots & \alpha_c \alpha_e k(t_i, t_j - \tau) \\ \vdots & \vdots & \vdots \\ \alpha_e \alpha_c k(t_j - \tau, t_i) & \dots & \alpha_e^2 k(t_j - \tau, t_j - \tau) + \sigma_{e,j}^2 \end{bmatrix} \right)$$

- or more compact:

$$p(\mathcal{D}|\alpha_c, \alpha_e, b_c, b_e, \tau)$$

Gaussian Process Cross Correlation

- Likelihood reads

$$\begin{bmatrix} y_c(t_i) \\ \vdots \\ y_e(t_j) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} b_c \\ \vdots \\ b_e \end{bmatrix}, \begin{bmatrix} \alpha_c^2 k(t_i, t_i) + \sigma_{c,i}^2 & \dots & \alpha_c \alpha_e k(t_i, t_j - \tau) \\ \vdots & \vdots & \vdots \\ \alpha_e \alpha_c k(t_j - \tau, t_i) & \dots & \alpha_e^2 k(t_j - \tau, t_j - \tau) + \sigma_{e,j}^2 \end{bmatrix} \right)$$

- or more compact:

$$p(\mathcal{D} | \alpha_c, \alpha_e, b_c, b_e, \tau)$$

- Posterior delay:

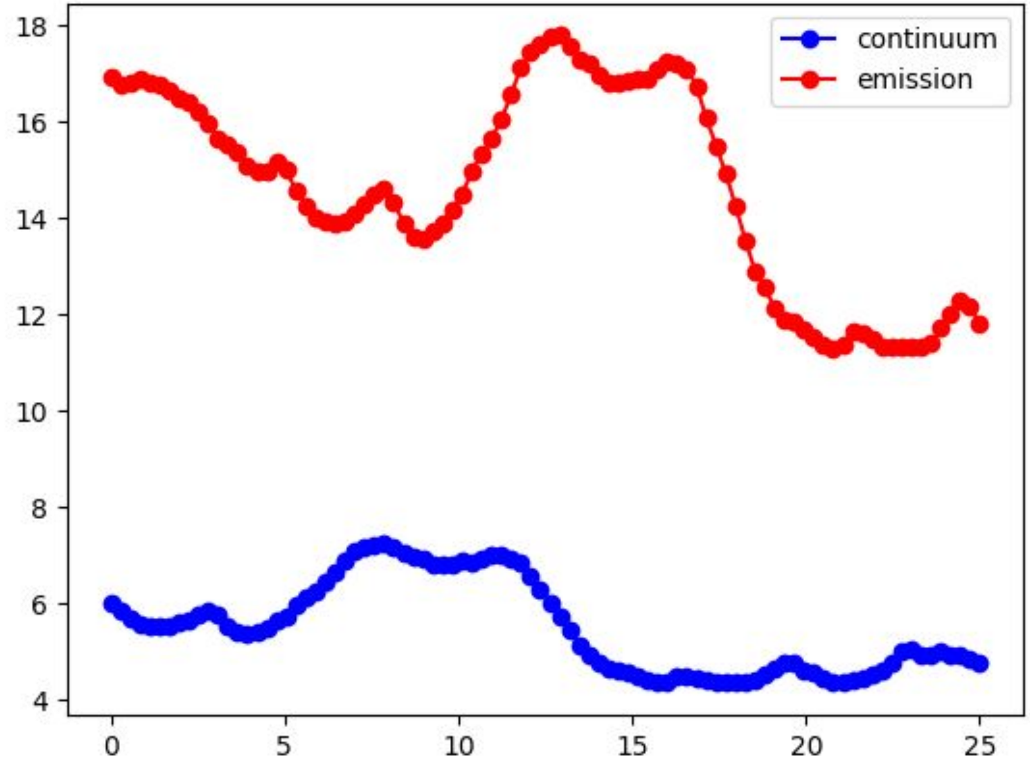
$$p(\tau|\mathcal{D}) \propto \int \overbrace{p(\mathcal{D}|\alpha_c, \alpha_e, b_c, b_e, \tau)}^{\text{likelihood}} \overbrace{p(\tau)p(\alpha_c)p(\alpha_e)p(b_c)p(b_e)p(\theta_\kappa)}^{\text{prior}} d\alpha_c d\alpha_e db_c db_e$$

- Integral cannot be analytically done, we resort to approximations not detailed here

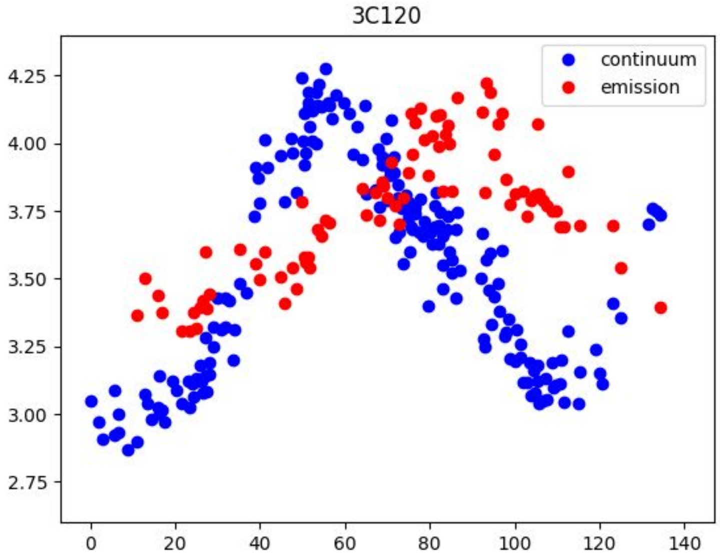
For brevity we ignore GP hyperparameters

Gaussian Process Cross Correlation

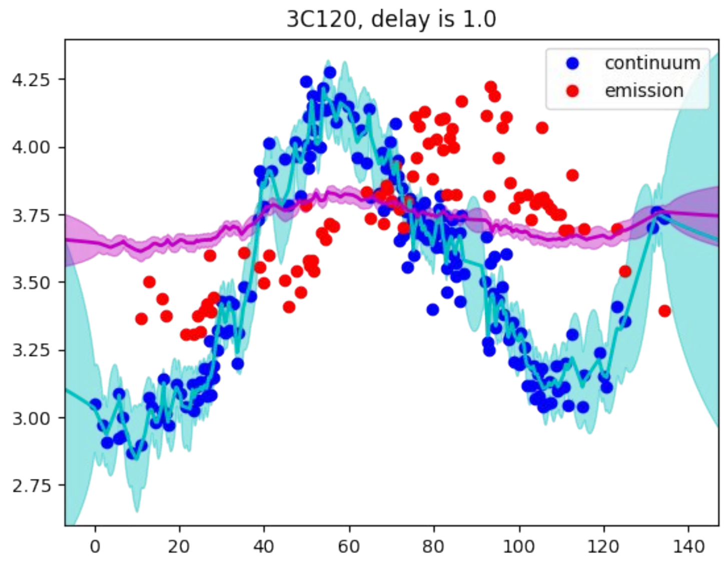
Sampling from our model generates

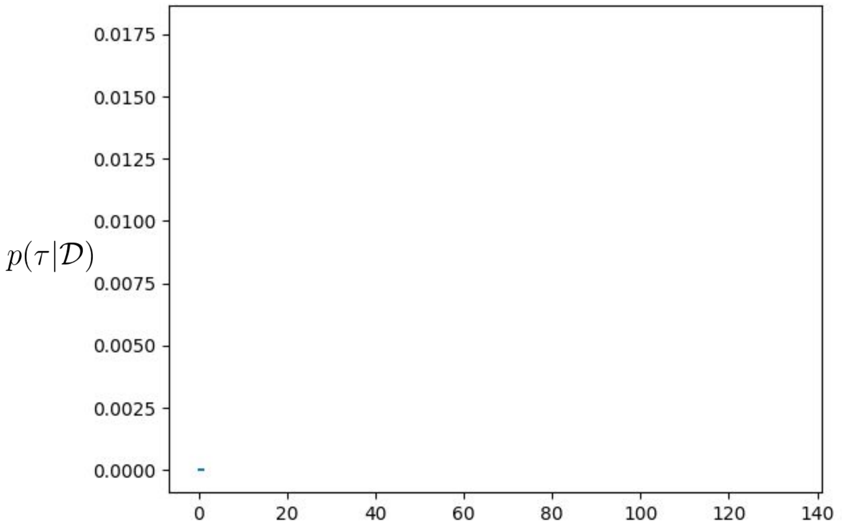


Posterior delay distribution for 3C120

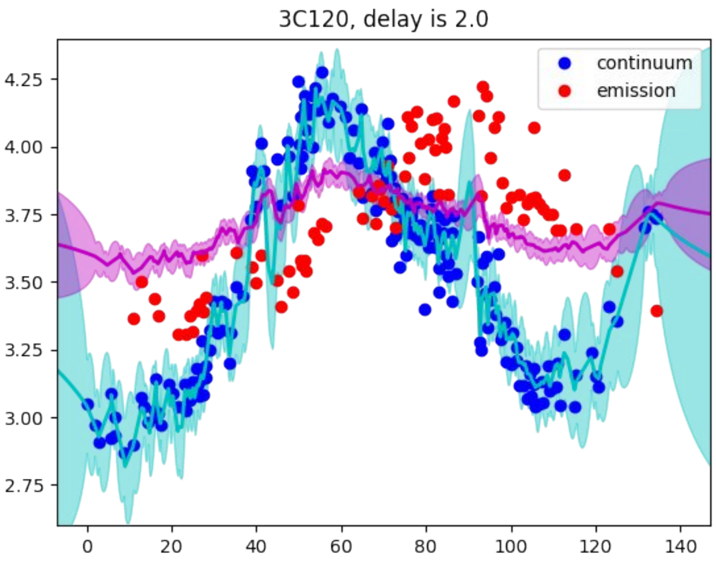


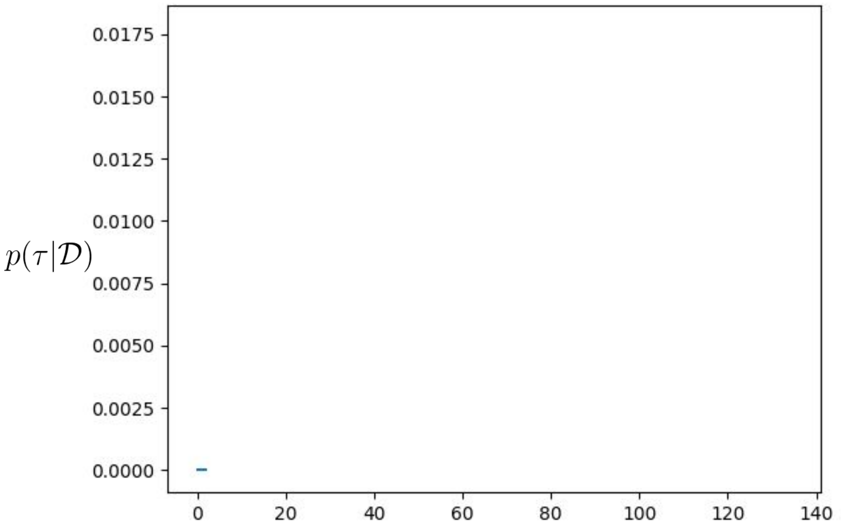
Posterior delay distribution for 3C120



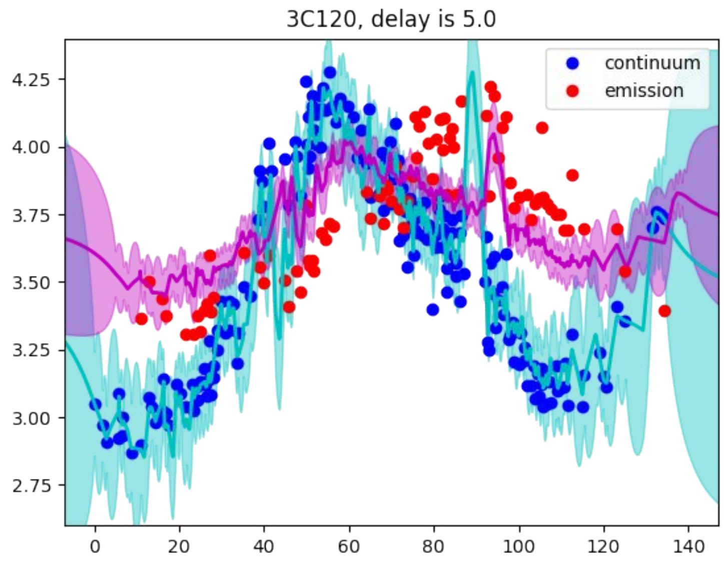


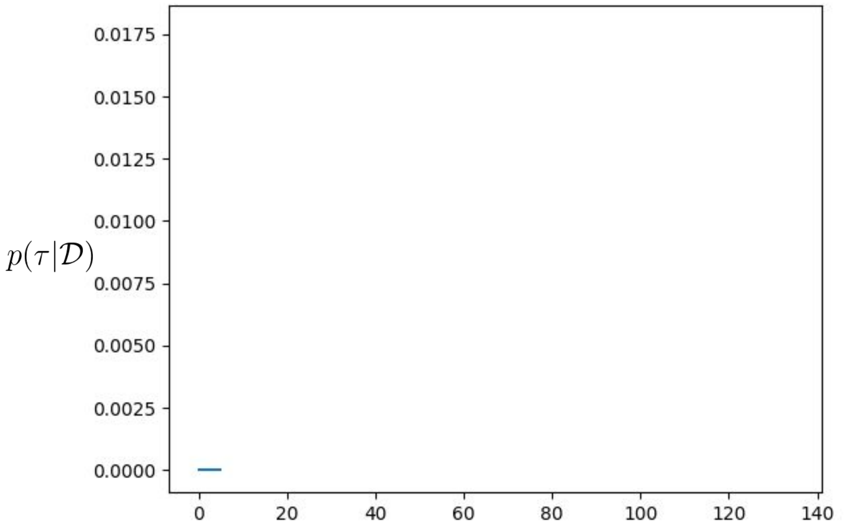
Posterior delay distribution for 3C120



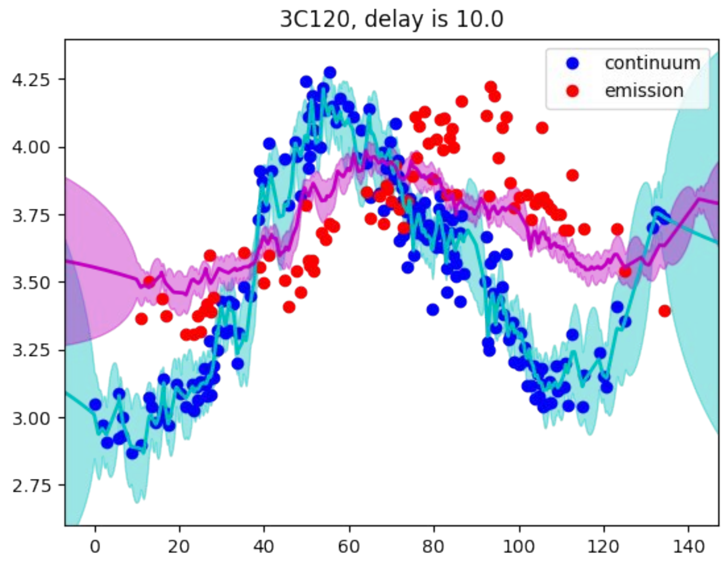


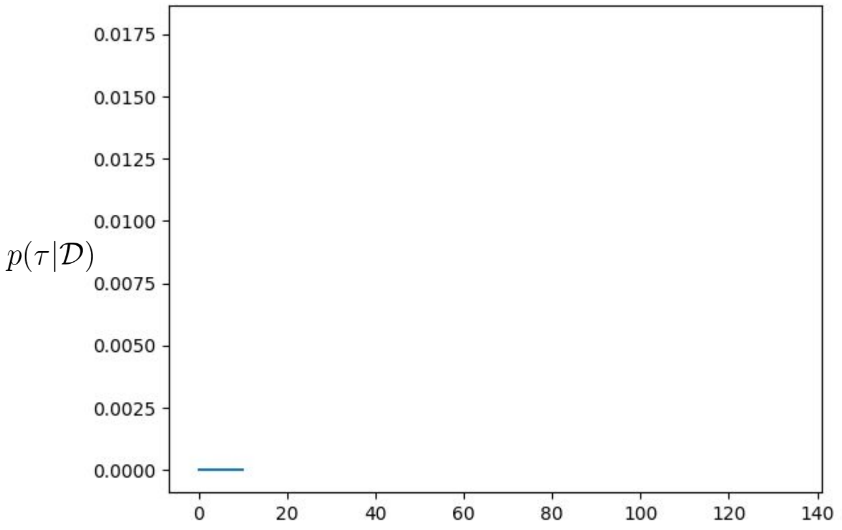
Posterior delay distribution for 3C120



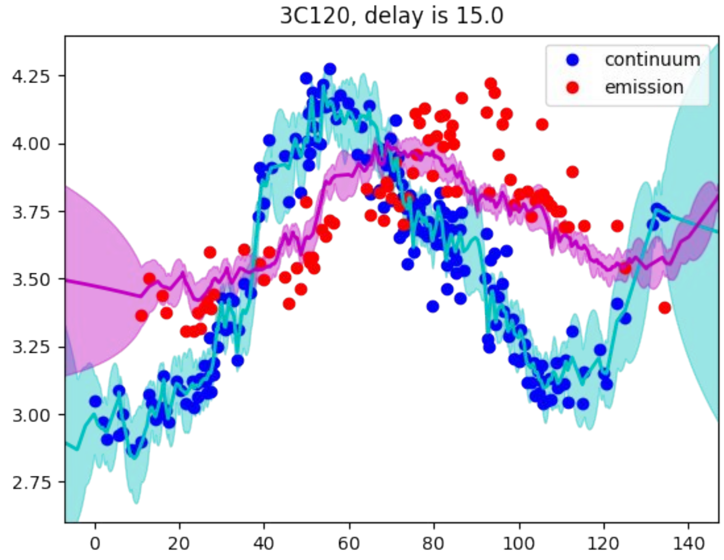


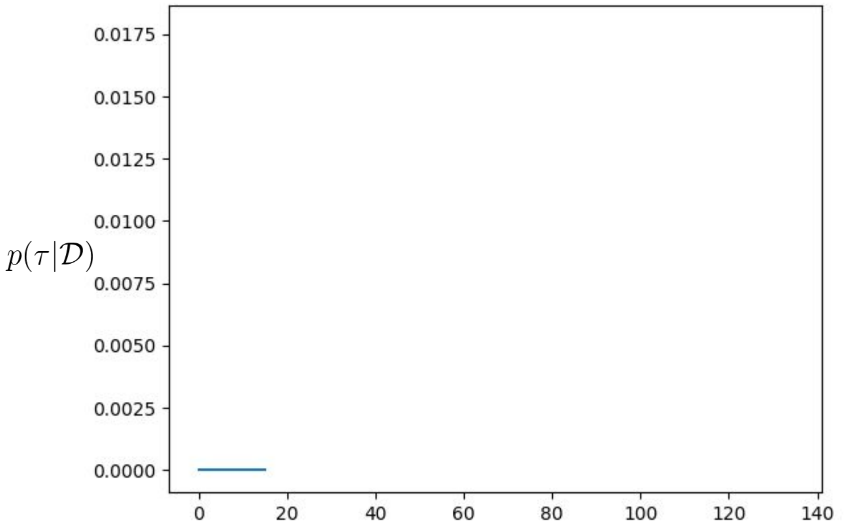
Posterior delay distribution for 3C120



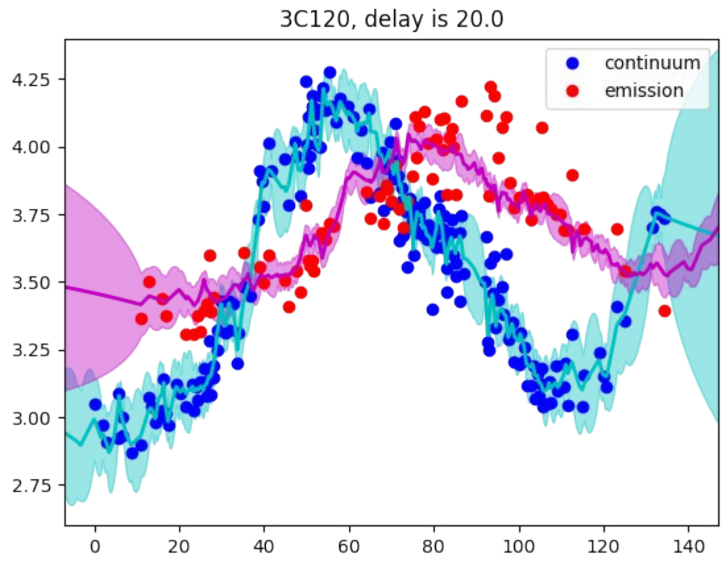


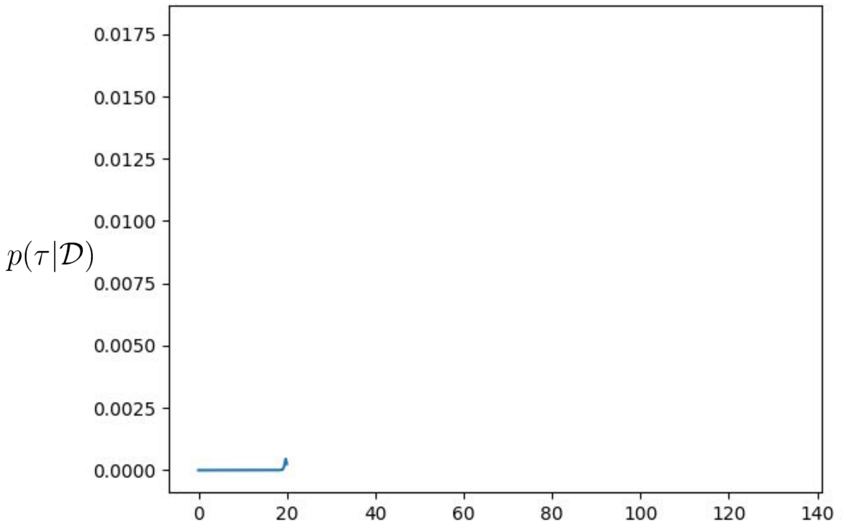
Posterior delay distribution for 3C120



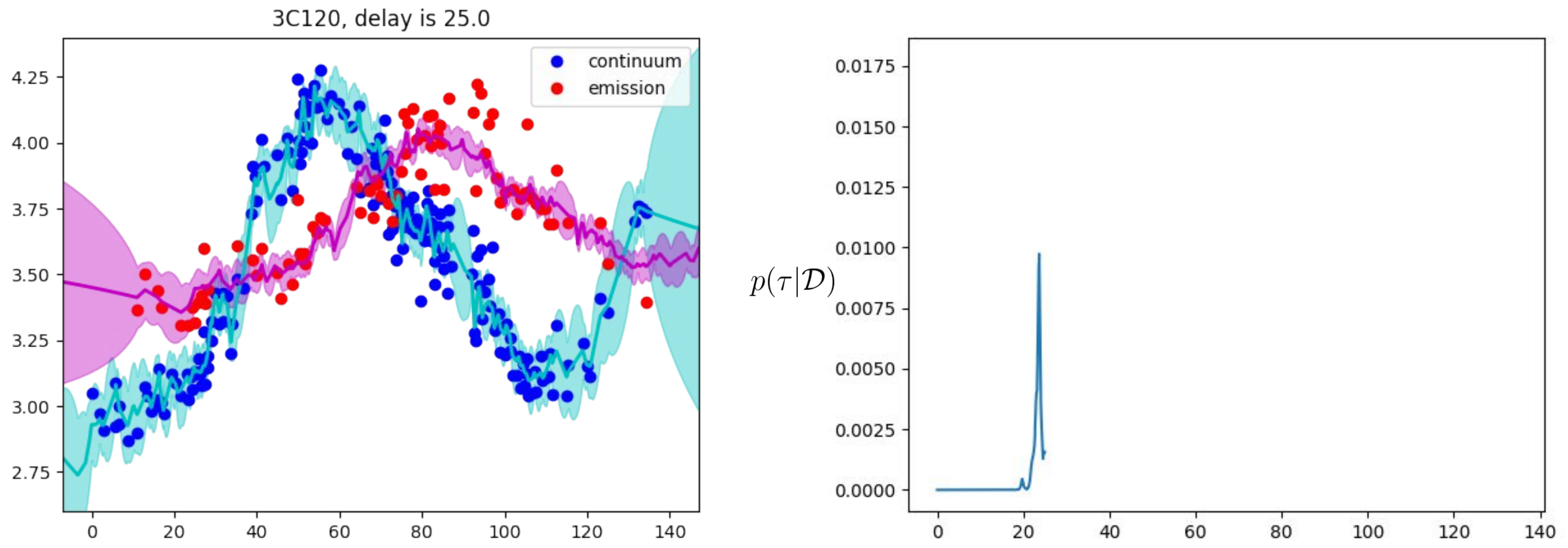


Posterior delay distribution for 3C120

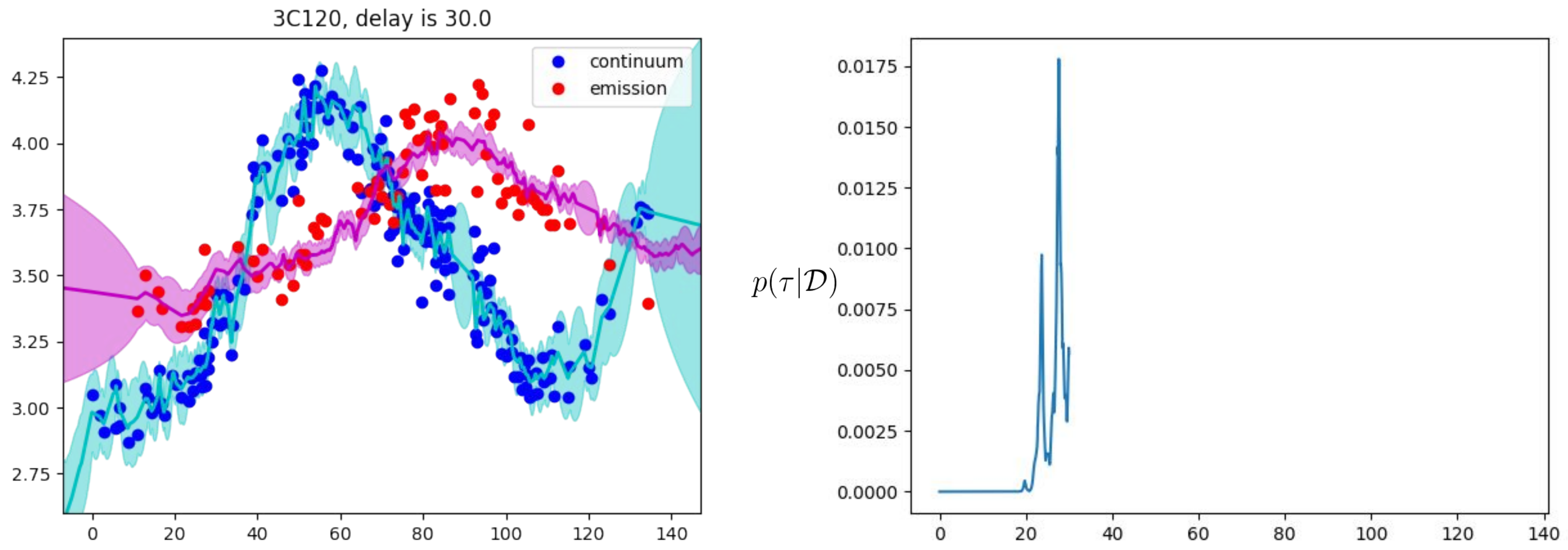




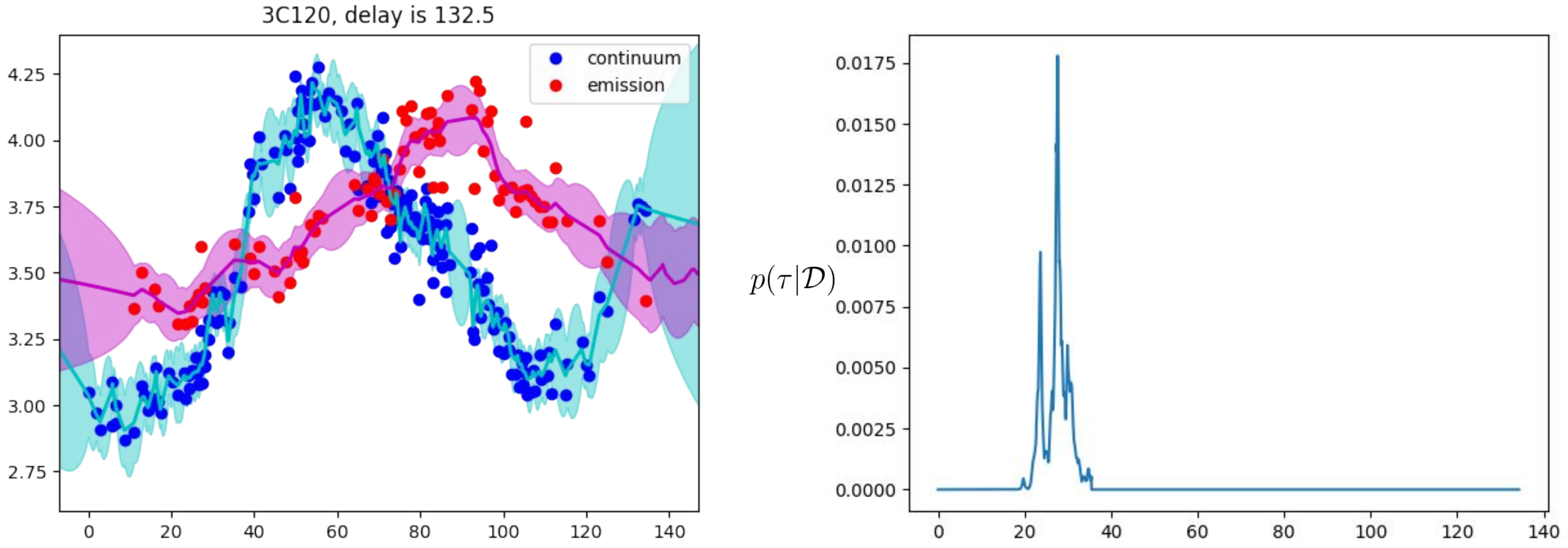
Posterior delay distribution for 3C120



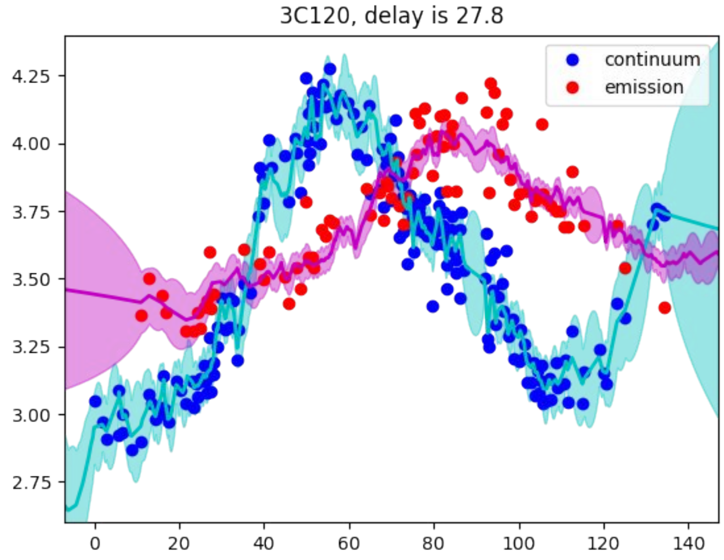
Posterior delay distribution for 3C120



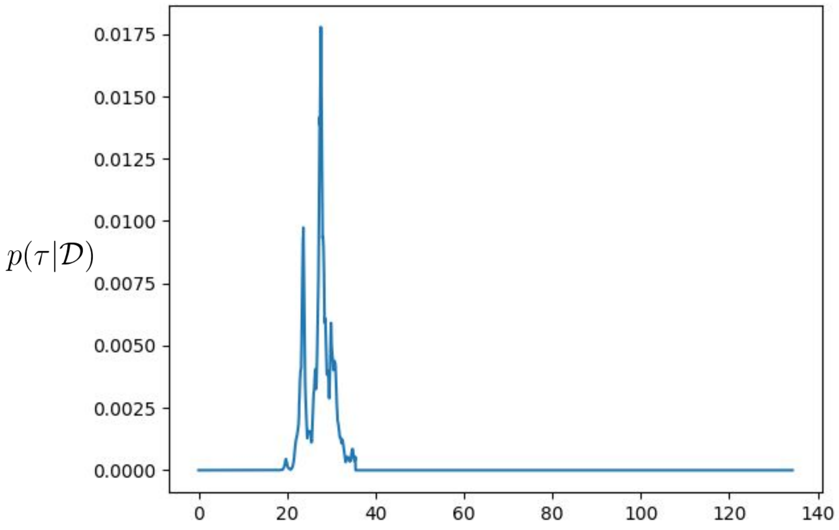
Posterior delay distribution for 3C120



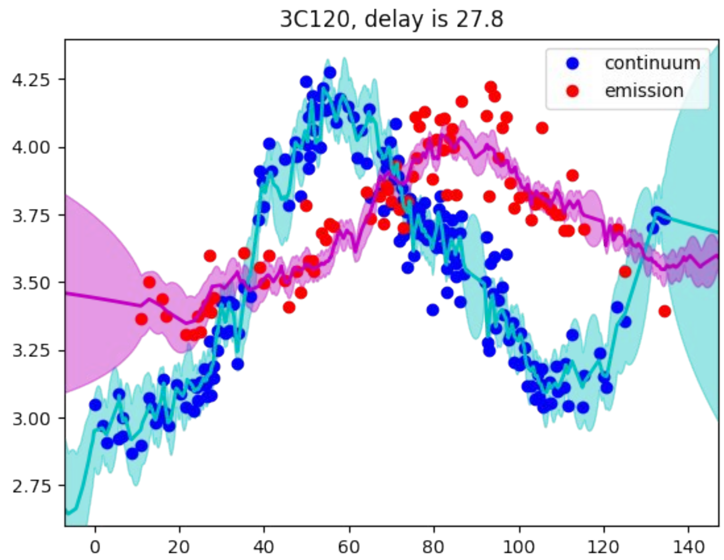
Posterior delay distribution for 3C120



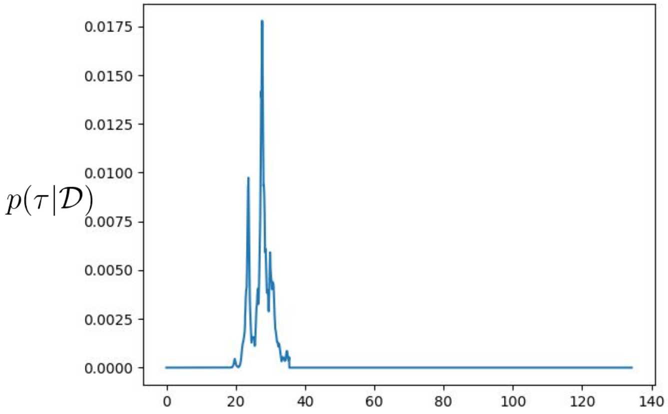
GP fit at posterior peak 27.8 days

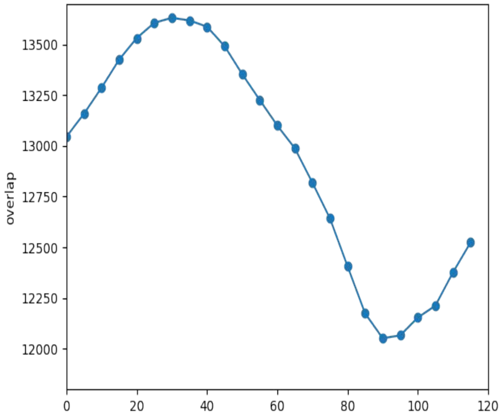


Posterior delay distribution for 3C120



GP fit at posterior peak 27.8 days





ICCF

Summary

We reformulated ICCF in a probabilistic manner:

- Lightcurves related as $$y_e(t) = \alpha_e f(t - \tau) + b_e$$ $$y_c(t) = \alpha_c f(t) + b_c$$
- Model f(t) as a GP
- Since GP closed under affine transformation, lightcurves also governed by GP
- We work out a posterior distribution for the delay

Our current model relies on simplification that one lightcurve is a **delayed**, **scaled** and **offsetted** version of the other.

In future work we want to be more physically realistic and use models of the type

$$y(t) = \int \mathcal{G}(\tau) f(t + \tau) d\tau$$

where a transfer function G encodes properties of the physical system