

# CS-E4895 Gaussian Processes

## Lecture 11: State-space GPs

Arno Solin

Aalto University

Monday 3.4.2023

# Roadmap for today

- 1 Motivation: Temporal models
- 2 Three views into GPs
- 3 General likelihoods
- 4 Spatio-temporal GPs
- 5 Further extensions
- 6 Recap

# Section 1

## Motivation: Temporal models

# Motivation: Temporal models

## ⌚ One-dimensional problems

(the data has a natural ordering)

## ⌚ Spatio-temporal models

(something developing over time)

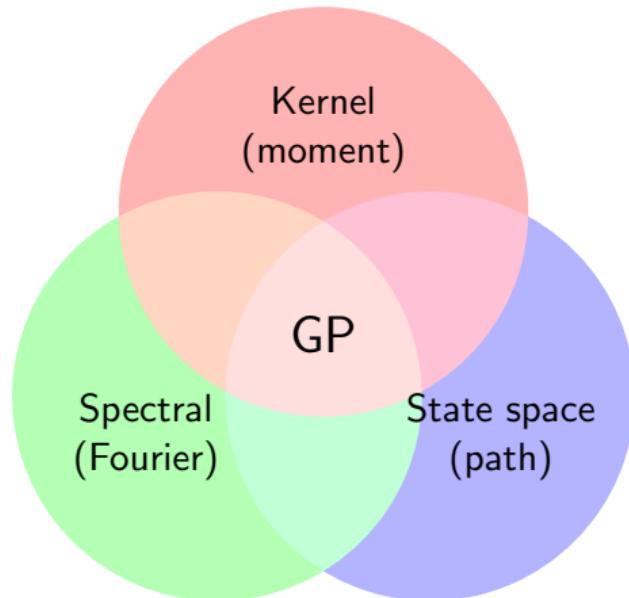
## ⌚ Long / unbounded data

(sensor data streams, daily observations, etc.)

## Section 2

### Three views into GPs

# Three views into (stationary) GPs



# Kernel (moment) representation

$$f(t) \sim \text{GP}(\mu(t), \kappa(t, t')) \quad \textit{GP prior}$$
$$\mathbf{y} \mid \mathbf{f} \sim \prod_i p(y_i \mid f(t_i)) \quad \textit{likelihood}$$

- Let's focus on the **GP prior** only.
- A **temporal** Gaussian process (GP) is a random function  $f(t)$ , such that joint distribution of  $f(t_1), \dots, f(t_n)$  is always Gaussian.
- Mean and covariance functions** have the form:

$$\mu(t) = \mathbb{E}[f(t)],$$
$$\kappa(t, t') = \mathbb{E}[(f(t) - \mu(t))(f(t') - \mu(t'))^\top].$$

- Convenient for **model specification**, but expanding the kernel to a **covariance matrix** can be **problematic** (the notorious  $\mathcal{O}(n^3)$  scaling).

## Spectral (Fourier) representation

- The Fourier transform of a function  $f(t) : \mathbb{R} \rightarrow \mathbb{R}$  is

$$\mathcal{F}[f](i\omega) = \int_{\mathbb{R}} f(t) \exp(-i\omega t) dt$$

- For a stationary GP, the covariance function can be written in terms of the difference between two inputs:

$$\kappa(t, t') \triangleq \kappa(t - t')$$

- Wiener–Kinchin: If  $f(t)$  is a stationary Gaussian process with covariance function  $\kappa(t)$  then its spectral density is  $S(\omega) = \mathcal{F}[\kappa]$ .
- Spectral representation of a GP in terms of spectral density function

$$S(\omega) = \mathbb{E}[\tilde{f}(i\omega) \tilde{f}^T(-i\omega)]$$

## State space (path) representation [1/3]

- Path or state space representation as solution to a linear time-invariant (LTI) **stochastic differential equation** (SDE):

$$d\mathbf{f} = \mathbf{F} \mathbf{f} dt + \mathbf{L} d\boldsymbol{\beta},$$

where  $\mathbf{f} = (f, df/dt, \dots)$  and  $\boldsymbol{\beta}(t)$  is a vector of Wiener processes.

- Equivalently, but more informally

$$\frac{d\mathbf{f}(t)}{dt} = \mathbf{F} \mathbf{f}(t) + \mathbf{L} \mathbf{w}(t),$$

where  $\mathbf{w}(t)$  is white noise.

- The model now consists of a **drift matrix**  $\mathbf{F} \in \mathbb{R}^{m \times m}$ , a **diffusion matrix**  $\mathbf{L} \in \mathbb{R}^{m \times s}$ , and the **spectral density matrix** of the white noise process  $\mathbf{Q}_c \in \mathbb{R}^{s \times s}$ .
- The scalar-valued GP can be recovered by  $f(t) = \mathbf{H} \mathbf{f}(t)$ .

## State space (path) representation [2/3]

- The **initial state** is given by a stationary state  $\mathbf{f}(0) \sim N(\mathbf{0}, \mathbf{P}_\infty)$  which fulfills

$$\mathbf{F} \mathbf{P}_\infty + \mathbf{P}_\infty \mathbf{F}^T + \mathbf{L} \mathbf{Q}_c \mathbf{L}^T = \mathbf{0}$$

- The **covariance function** at the stationary state can be recovered by

$$\kappa(t, t') = \begin{cases} \mathbf{P}_\infty \exp((t' - t)\mathbf{F})^T, & t' \geq t \\ \exp((t' - t)\mathbf{F}) \mathbf{P}_\infty & t' < t \end{cases}$$

where  $\exp(\cdot)$  denotes the **matrix exponential** function.

- The **spectral density function** at the stationary state can be recovered by

$$S(\omega) = (\mathbf{F} + i\omega \mathbf{I})^{-1} \mathbf{L} \mathbf{Q}_c \mathbf{L}^T (\mathbf{F} - i\omega \mathbf{I})^{-T}$$

## State space (path) representation [3/3]

- Similarly as the kernel has to be evaluated into covariance matrix for computations, the SDE can be **solved** for discrete time points  $\{t_i\}_{i=1}^n$ .
- The resulting model is a **discrete state space model**:

$$\mathbf{f}_i = \mathbf{A}_{i-1} \mathbf{f}_{i-1} + \mathbf{q}_{i-1}, \quad \mathbf{q}_i \sim N(\mathbf{0}, \mathbf{Q}_i),$$

where  $\mathbf{f}_i = \mathbf{f}(t_i)$ .

- The **discrete-time model** matrices are given by:

$$\mathbf{A}_i = \exp(\mathbf{F} \Delta t_i),$$

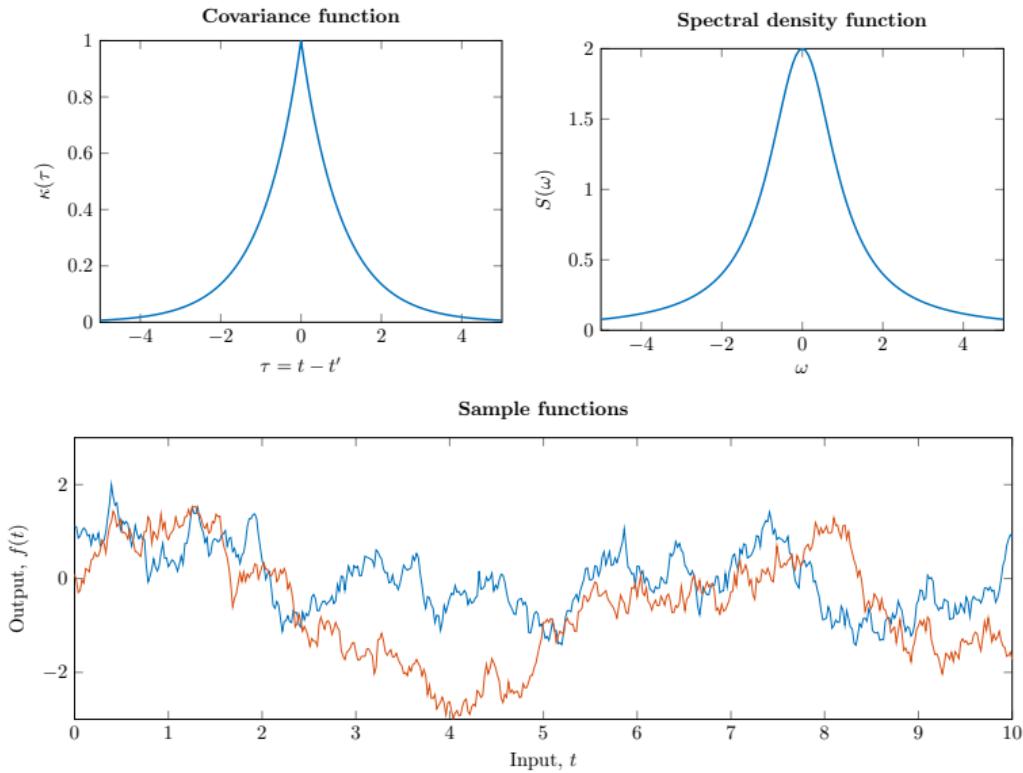
$$\mathbf{Q}_i = \int_0^{\Delta t_i} \exp(\mathbf{F} (\Delta t_i - \tau)) \mathbf{L} \mathbf{Q}_c \mathbf{L}^\top \exp(\mathbf{F} (\Delta t_i - \tau))^\top d\tau,$$

where  $\Delta t_i = t_{i+1} - t_i$

- If the model is stationary,  $\mathbf{Q}_i$  is given by

$$\mathbf{Q}_i = \mathbf{P}_\infty - \mathbf{A}_i \mathbf{P}_\infty \mathbf{A}_i^\top$$

# Three views into GPs



## Example: Exponential covariance function

- Exponential covariance function (Ornstein-Uhlenbeck process):

$$\kappa(t, t') = \exp(-\lambda |t - t'|)$$

- Spectral density function:

$$S(\omega) = \frac{2}{\lambda + \omega^2/\lambda}$$

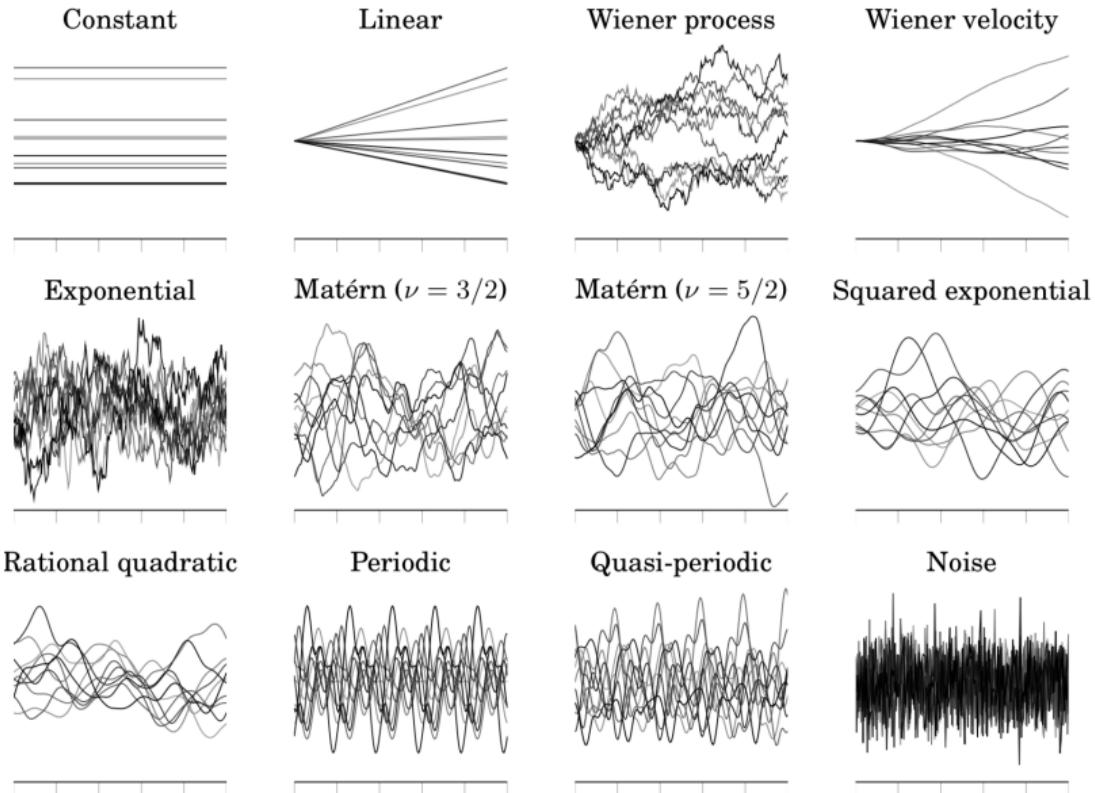
- Path representation: Stochastic differential equation (SDE)

$$\frac{df(t)}{dt} = -\lambda f(t) + w(t),$$

or using the notation from before:

$F = -\lambda$ ,  $L = 1$ ,  $Q_c = 2$ ,  $H = 1$ , and  $P_\infty = 1$ .

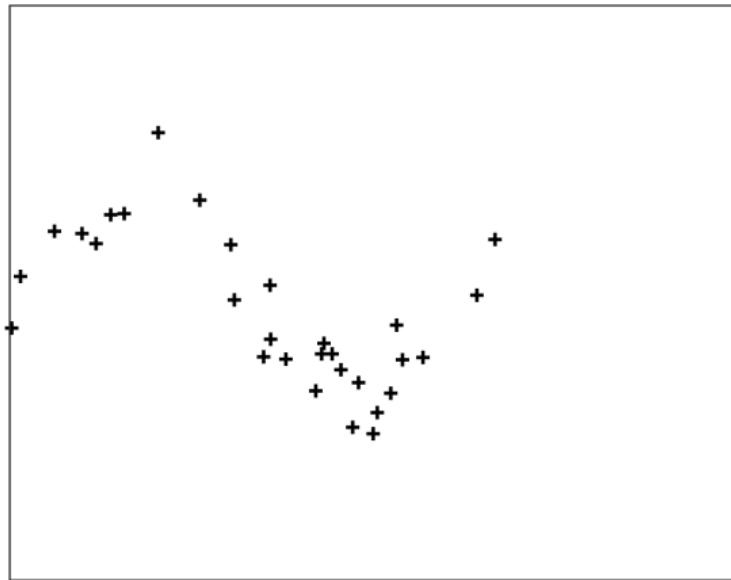
# Examples of applicable GP priors



## Applicable GP priors

- The covariance function needs to be **Markovian** (or approximated as such).
- Covers many common **stationary** and **non-stationary** models.
- **Sums of kernels:**  $\kappa(t, t') = \kappa_1(t, t') + \kappa_2(t, t')$ 
  - Stacking of the state spaces
  - State dimension:  $m = m_1 + m_2$
- **Product of kernels:**  $\kappa(t, t') = \kappa_1(t, t') \kappa_2(t, t')$ 
  - Kronecker sum of the models
  - State dimension:  $m = m_1 m_2$

## Example: GP regression, $\mathcal{O}(n^3)$



## Example: GP regression, $\mathcal{O}(n^3)$

- Consider the GP regression problem with input–output training pairs  $\{(t_i, y_i)\}_{i=1}^n$ :

$$f(t) \sim \text{GP}(0, \kappa(t, t')),$$

$$y_i = f(t_i) + \varepsilon_i, \quad \varepsilon_i \sim N(0, \sigma_n^2)$$

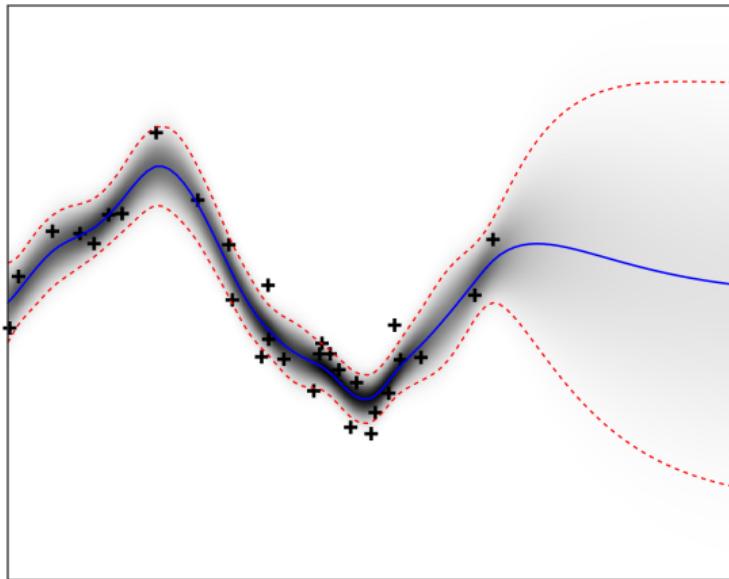
- The posterior mean and variance for an unseen test input  $t_*$  is given by (see previous lectures):

$$\mathbb{E}[f_*] = \mathbf{k}_* (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y},$$

$$\mathbb{V}[f_*] = \kappa(t_*, t_*) - \mathbf{k}_* (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{k}_*^\top$$

- Note the inversion of the  $n \times n$  matrix.

## Example: GP regression, $\mathcal{O}(n^3)$



## Example: GP regression, $\mathcal{O}(n)$

- The **sequential solution** (goes under the name '**Kalman filter**') considers one data point at a time, hence the linear time-scaling.
- Start from  $\mathbf{m}_0 = \mathbf{0}$  and  $\mathbf{P}_0 = \mathbf{P}_\infty$  and for each data point iterate the following steps.
- **Kalman prediction:**

$$\begin{aligned}\mathbf{m}_{i|i-1} &= \mathbf{A}_{i-1} \mathbf{m}_{i-1|i-1}, \\ \mathbf{P}_{i|i-1} &= \mathbf{A}_{i-1} \mathbf{P}_{i-1|i-1} \mathbf{A}_{i-1}^\top + \mathbf{Q}_{i-1}.\end{aligned}$$

- **Kalman update:**

$$\begin{aligned}\mathbf{v}_i &= y_i - \mathbf{H} \mathbf{m}_{i|i-1}, \\ \mathbf{S}_i &= \mathbf{H}_i \mathbf{P}_{i|i-1} \mathbf{H}^\top + \sigma_n^2, \\ \mathbf{K}_i &= \mathbf{P}_{i|i-1} \mathbf{H}^\top \mathbf{S}_i^{-1}, \\ \mathbf{m}_{i|i} &= \mathbf{m}_{i|i-1} + \mathbf{K}_i \mathbf{v}_i, \\ \mathbf{P}_{i|i} &= \mathbf{P}_{i|i-1} - \mathbf{K}_i \mathbf{S}_i \mathbf{K}_i^\top.\end{aligned}$$

## Example: GP regression, $\mathcal{O}(n)$

- To condition all time-marginals on all data, run a backward sweep (**Rauch–Tung–Striebel smoother**):

$$\mathbf{m}_{i+1|i} = \mathbf{A}_i \mathbf{m}_{i|i},$$

$$\mathbf{P}_{i+1|i} = \mathbf{A}_i \mathbf{P}_{i|i} \mathbf{A}_i^\top + \mathbf{Q}_i,$$

$$\mathbf{G}_i = \mathbf{P}_{i|i} \mathbf{A}_i^\top \mathbf{P}_{i+1|i}^{-1},$$

$$\mathbf{m}_{i|n} = \mathbf{m}_{i|i} + \mathbf{G}_i (\mathbf{m}_{i+1|n} - \mathbf{m}_{i+1|i}),$$

$$\mathbf{P}_{i|n} = \mathbf{P}_{i|i} + \mathbf{G}_i (\mathbf{P}_{i+1|n} - \mathbf{P}_{i+1|i}) \mathbf{G}_i^\top,$$

- The marginal mean and variance can be recovered by:

$$\mathbb{E}[f_i] = \mathbf{H} \mathbf{m}_{i|n},$$

$$\mathbb{V}[f_i] = \mathbf{H} \mathbf{P}_{i|n} \mathbf{H}^\top$$

- The log **marginal likelihood** can be evaluated as a by-product of the Kalman update:

$$\log p(\mathbf{y}) = -\frac{1}{2} \sum_{i=1}^n \log |2\pi \mathbf{S}_i| + \mathbf{v}_i^\top \mathbf{S}_i^{-1} \mathbf{v}_i$$

## Example: GP regression, $\mathcal{O}(n)$

## Example

- Number of births in the US
- Daily data between 1969–1988 ( $n = 7305$ )
- GP regression with a prior covariance function:

$$\begin{aligned}\kappa(t, t') &= \kappa_{\text{Mat.}}^{\nu=5/2}(t, t') + \kappa_{\text{Mat.}}^{\nu=3/2}(t, t') \\ &\quad + \kappa_{\text{Per.}}^{\text{year}}(t, t') \kappa_{\text{Mat.}}^{\nu=3/2}(t, t') + \kappa_{\text{Per.}}^{\text{week}}(t, t') \kappa_{\text{Mat.}}^{\nu=3/2}(t, t')\end{aligned}$$

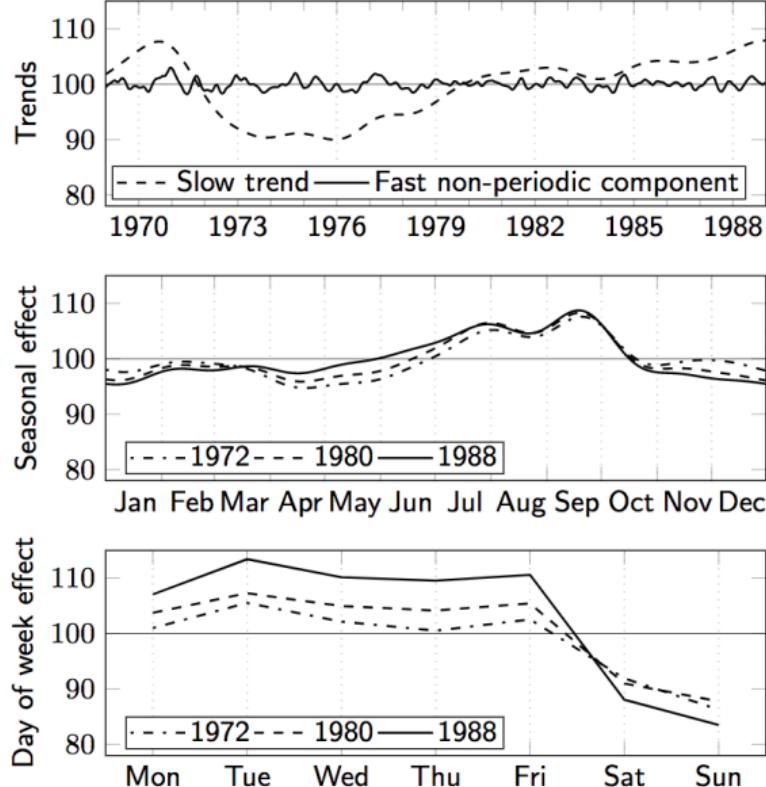
- Learn hyperparameters by optimizing the marginal likelihood

## Example

- Number of births
- Daily data between
- GP regression with

$\kappa(t,$

- Learn hyperparam



Explaining changes in number of births in the US

# Section 3

## General likelihoods

# Non-Gaussian likelihoods

- The observation model might not be Gaussian

$$f(t) \sim \text{GP}(0, \kappa(t, t'))$$

$$\mathbf{y} \mid \mathbf{f} \sim \prod_i p(y_i \mid f(t_i))$$

- There exists a multitude of great methods to tackle general likelihoods with approximations of the form

$$\mathbb{Q}(\mathbf{f} \mid \mathcal{D}) = \mathcal{N}(\mathbf{f} \mid \mathbf{m} + \mathbf{K}\boldsymbol{\alpha}, (\mathbf{K}^{-1} + \mathbf{W})^{-1})$$

- Use those methods, but deal with the latent using state space models

# Inference

- Laplace approximation
  - (both inner-loop and outer-loop)
- Variational inference
- Direct KL minimization
- Assumed denisty filtering / Single-sweep EP
  - (only requires one-pass through the data)
- Can be evaluated in terms of a (Kalman) filter forward and backward pass, or by iterating them

## Example

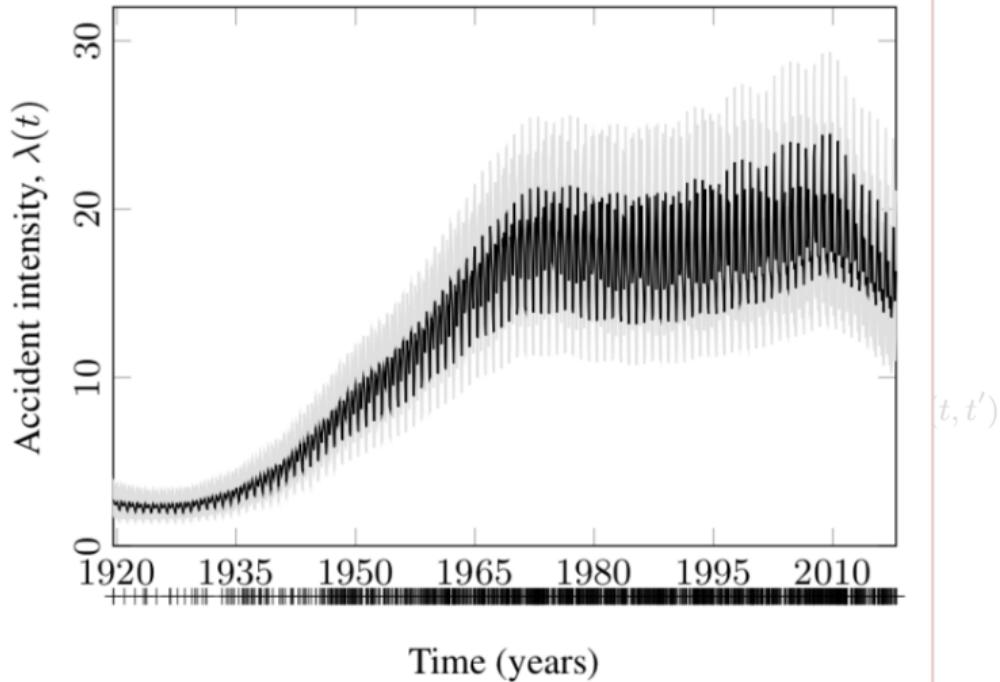
- Commercial aircraft accidents 1919–2017
- Log-Gaussian Cox process (Poisson likelihood) by ADF/EP
- Daily binning,  $n = 35,959$
- GP prior with a covariance function:

$$\kappa(t, t') = \kappa_{\text{Mat.}}^{\nu=3/2}(t, t') + \kappa_{\text{Per.}}^{\text{year}}(t, t') \kappa_{\text{Mat.}}^{\nu=3/2}(t, t') + \kappa_{\text{Per.}}^{\text{week}}(t, t') \kappa_{\text{Mat.}}^{\nu=3/2}(t, t')$$

- Learn hyperparameters by optimizing the marginal likelihood

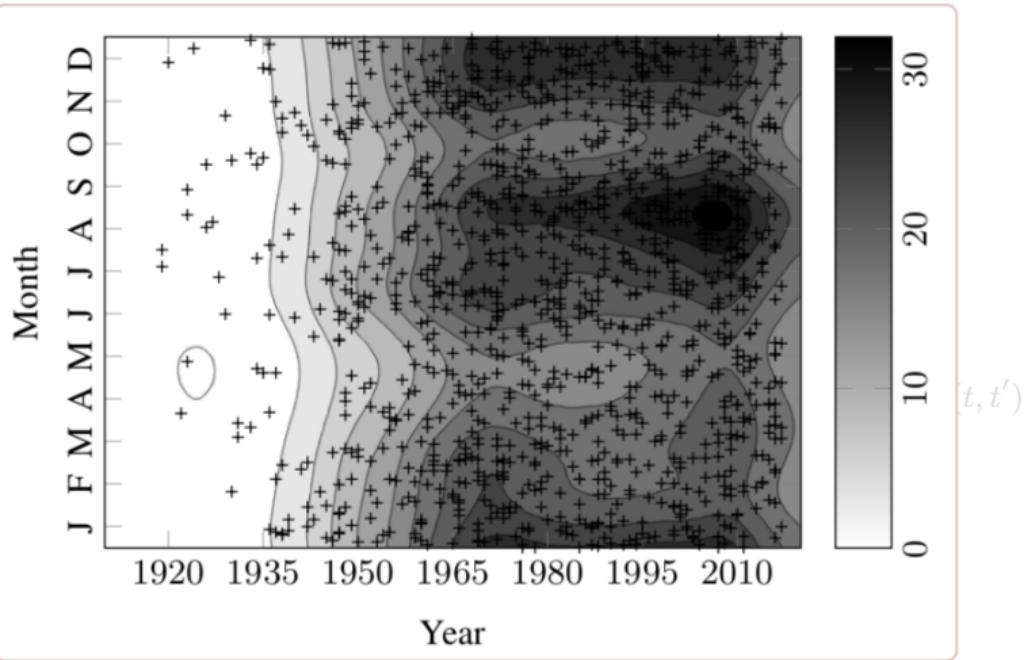
## Example

- Commercial a
- Log-Gaussian
- Daily binning
- GP prior with
- Learn hyperp



## Example

- Commercial air traffic
- Log-Gaussian
- Daily binning
- GP prior with
- Learn hyperparameters



## Section 4

### Spatio-temporal GPs

# Spatio-temporal GPs

$$f(\mathbf{x}) \sim \text{GP}(0, \kappa(\mathbf{x}, \mathbf{x}'))$$

$$\mathbf{y} \mid \mathbf{f} \sim \prod_i p(y_i \mid f(\mathbf{x}_i))$$

$$f(\mathbf{r}, t) \sim \text{GP}(0, \kappa(\mathbf{r}, t; \mathbf{r}', t'))$$

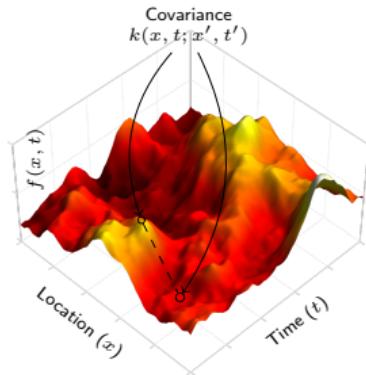
$$\mathbf{y} \mid \mathbf{f} \sim \prod_i p(y_i \mid f(\mathbf{r}_i, t_i))$$

# Spatio-temporal Gaussian processes

## GPs under the kernel formalism

$$f(\mathbf{x}, t) \sim \text{GP}(0, \kappa(\mathbf{x}, t; \mathbf{x}', t'))$$

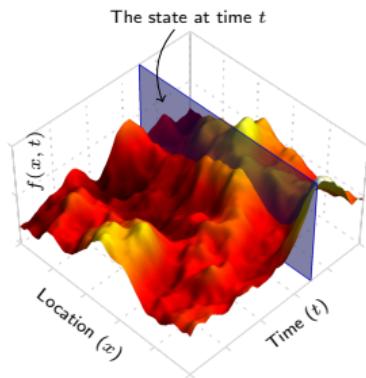
$$y_i = f(\mathbf{x}_i, t_i) + \varepsilon_i$$



## Stochastic partial differential equation formalism

$$\frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial t} = \mathcal{F} \mathbf{f}(\mathbf{x}, t) + \mathcal{L} w(\mathbf{x}, t)$$

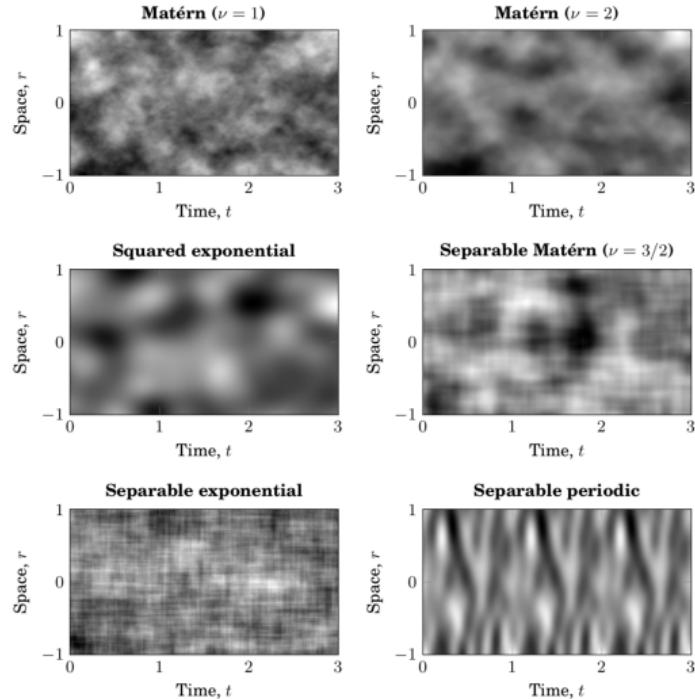
$$y_i = \mathcal{H}_i \mathbf{f}(\mathbf{x}, t) + \varepsilon_i$$



# Spatio-temporal GP regression

# Spatio-temporal GP regression

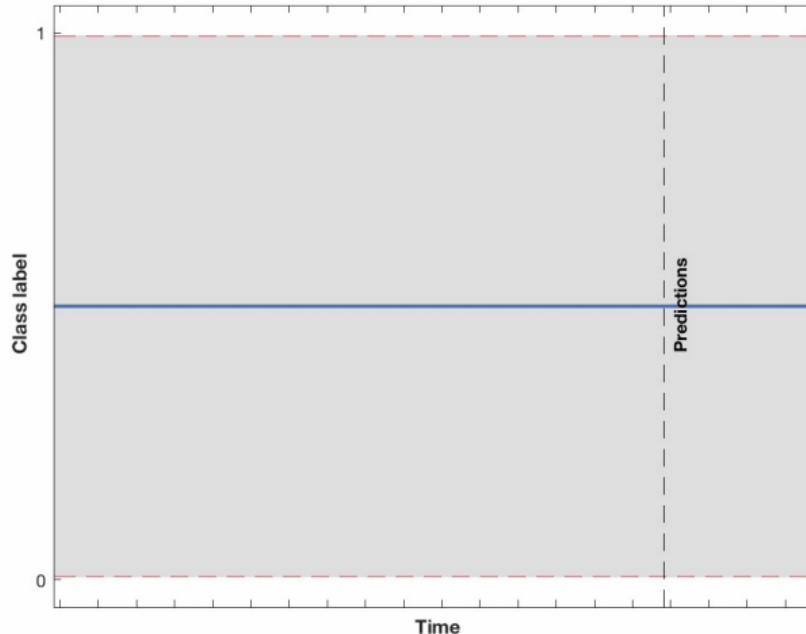
# Spatio-temporal GP priors



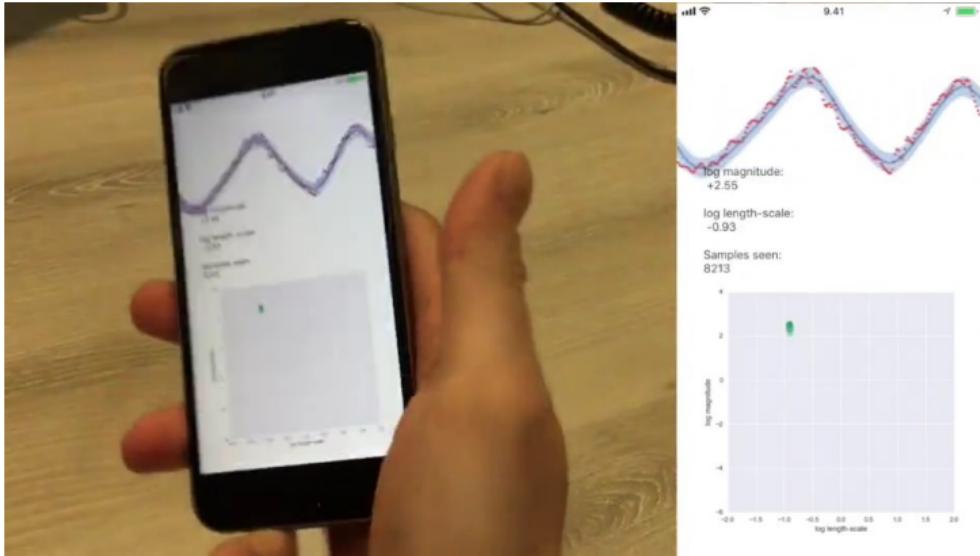
# Section 5

## Further extensions

# What if the data really is infinite?



# Infinite time-horizon with adapting the hyperparameters online



<https://youtu.be/myCvUT3XGPc>

# Section 6

## Recap

# Gaussian processes ❤️ SDEs

GPs under the kernel formalism

$$f(t) \sim \text{GP}(0, \kappa(t, t'))$$

$$\mathbf{y} | \mathbf{f} \sim \prod_i p(y_i | f(t_i))$$

Flexible model specification

Stochastic differential equations

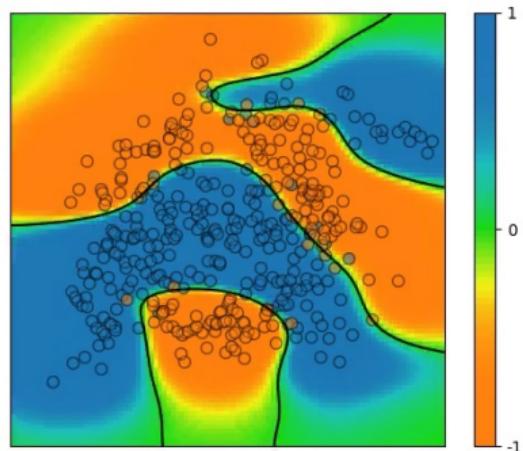
$$d\mathbf{f}(t) = \mathbf{F} \mathbf{f}(t) dt + \mathbf{L} d\boldsymbol{\beta}(t)$$

$$y_i \sim p(y_i | \mathbf{h}^\top \mathbf{f}(t_i))$$

Inference /  
First-principles

## Recap

- Gaussian processes have different representations:
  - Covariance function
  - Spectral density
  - State space
- Temporal (single-input) Gaussian processes  
 $\iff$  stochastic differential equations (SDEs)
- Conversions between the representations can make model building easier
- (Exact) inference of the latent functions, can be done in  $\mathcal{O}(n)$  time and memory complexity by Kalman filtering



# Bibliography

The examples and methods presented on this lecture are presented in greater detail in the following works:

- Särkkä, S., Solin, A., and Hartikainen, J. (2013). *Spatio-temporal learning via infinite-dimensional Bayesian filtering and smoothing*. *IEEE Signal Processing Magazine*, 30(4):51–61.
- Särkkä, S. (2013). *Bayesian Filtering and Smoothing*. Cambridge University Press. Cambridge, UK.
- Solin, A. (2016). *Stochastic Differential Equation Methods for Spatio-Temporal Gaussian Process Regression*. Doctoral dissertation, Aalto University.
- Solin, A., Hensman, J., and Turner, R.E. (2018). *Infinite-horizon Gaussian processes*. *Advances in Neural Information Processing Systems (NeurIPS)*, pages 3490–3499. Montréal, Canada.
- Särkkä, S., and Solin, A. (2019). *Applied Stochastic Differential Equations*. Cambridge University Press. Cambridge, UK.

# ICML 2020 tutorial on Machine Learning with Signal Processing

## Machine Learning with Signal Processing

ICML 2020 TUTORIAL

Arno Solin



Part I

Tools and  
discrete-time  
models



Part II

SDEs  
(continuous-time  
models)



Part III

Gaussian  
processes



Part IV

Application  
examples

@arnosolin

arno.solin.fi

[https://youtu.be/vTRD03\\_yReI](https://youtu.be/vTRD03_yReI)

## What next?

- Last lecture on the course tomorrow
- Last set of exercises to be published this week  
(no Thursday exercise session this week due to Spring Break over Easter,  
but there will be a video with help & hints)
- Course feedback (Webropol) opening soon  
(you should receive a personal link over email)