Spectral kernels

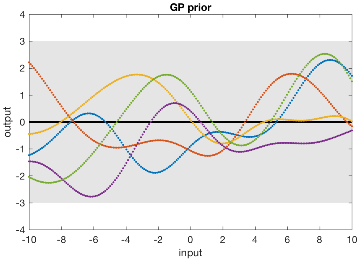
Markus Heinonen, Research fellow
Aalto University, Finland

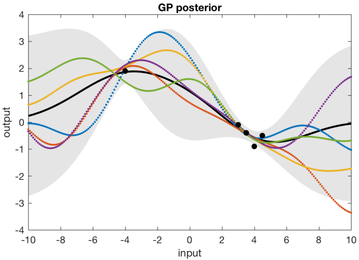
GPSS'21
13.9.2021

Gaussian processes

- Bayesian non-parametric kernel model
- Key idea: function prior $f(x) \sim \mathcal{GP}(m(x), K_\theta(x, x'))$ that encodes

$$p\begin{pmatrix} f(x_1) \\ \vdots \\ f(x_N) \end{pmatrix} = \mathcal{N}\left(\underbrace{\begin{bmatrix} f(x_1) \\ \vdots \\ f(x_N) \end{bmatrix}}_{\mathbf{f}} \middle| \underbrace{\begin{bmatrix} m(x_1) \\ \vdots \\ m(x_N) \end{bmatrix}}_{\mathbf{m}}, \underbrace{\begin{bmatrix} K_\theta(x_1, x_1) & \cdots & K_\theta(x_1, x_N) \\ \vdots & \ddots & \vdots \\ K_\theta(x_N, x_1) & \cdots & K_\theta(x_N, x_N) \end{bmatrix}}_{K_\theta} \right)$$





How to learn a kernel?

- Choose prior with maximum volume of data-matching functions

$$\underbrace{\log p(\mathbf{y}|\theta)}_{\text{marginal log likelihood}} = \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\theta)d\mathbf{f} \tag{1}$$

$$= -\frac{1}{2}\underbrace{\mathbf{y}^T(K_\theta + \sigma^2 I)^{-1}\mathbf{y}}_{\text{data fit}} - \frac{1}{2}\underbrace{\log|K_\theta + \sigma^2 I|}_{\text{model complexity}} - \frac{N}{2}\log 2\pi \tag{2}$$

- Relatively robust against overfitting
  - Determinant: volume of space spanned by kernel
  - Finds a simple basis for the data
  - Overfitting still possible, if $p(\mathbf{f})$ can be shaped to match $p(\mathbf{y}|\mathbf{f})$
- Powerful formalism to learn kernels
  - Replaces cross-validation
  - We can (auto)differentiate $\log p(\mathbf{y}|\theta)$ and optimise wrt θ
  - `GPflow`, `GPyTorch`, `Stan`, `GaussianProcesses.jl`, etc

How to learn a kernel?

- Choose prior with maximum volume of data-matching functions

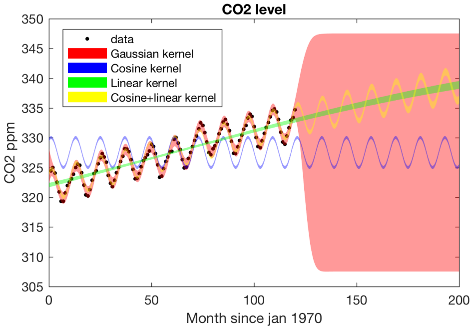
$$\underbrace{\log p(\mathbf{y}|\theta)}_{\text{marginal log likelihood}} = \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\theta)d\mathbf{f} \tag{1}$$

$$= -\frac{1}{2}\underbrace{\mathbf{y}^T(K_\theta + \sigma^2 I)^{-1}\mathbf{y}}_{\text{data fit}} - \frac{1}{2}\underbrace{\log|K_\theta + \sigma^2 I|}_{\text{model complexity}} - \frac{N}{2}\log 2\pi \tag{2}$$

- Relatively robust against overfitting
  - Determinant: volume of space spanned by kernel
  - Finds a simple basis for the data
  - Overfitting still possible, if $p(\mathbf{f})$ can be shaped to match $p(\mathbf{y}|\mathbf{f})$
- Powerful formalism to learn kernels
  - Replaces cross-validation
  - We can (auto)differentiate $\log p(\mathbf{y}|\theta)$ and optimise wrt θ
  - `GPflow`, `GPyTorch`, `Stan`, `GaussianProcesses.jl`, etc

How to choose kernel?

- Gaussian kernel $K_g(x, x') = \exp\left(-\frac{(x-x')^2}{2\ell^2}\right)$
- Periodic kernel $K_{cos}(x, x') = \exp\left(-\frac{2\sin^2(\pi|x-x'|/p}{\ell^2})\right)$
- Linear kernel $K_{lin}(x, x') = xx' + c$
- Composite kernel, eg $K(x, x') = K_g(x, x') + K_{lin}(x, x')$



- Our topic: Spectral kernels can learn arbitrary kernel function forms

Fourier transforms

- Fourier transform $S(\omega)$ of a function $f(x)$,

$$S(\omega) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \omega} dx \tag{3}$$

  where

  - i is the imaginary number with $i^2 = -1$ and $i^0 = 1$
  - ω is a frequency

- Inverse Fourier transform $f(x)$ of spectral density $S(\omega)$,

$$f(x) = \int_{-\infty}^{\infty} S(\omega) e^{2\pi i x \omega} d\omega \tag{4}$$

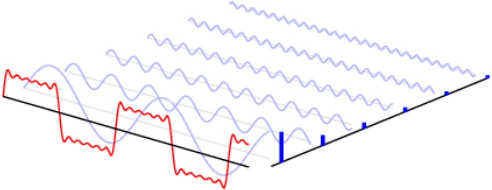
- Euler's identity helps compute Fouriers in practise

$$e^{ix} = \underbrace{\cos x}_{\text{real part}} + \underbrace{i \sin x}_{\text{imaginary part}} \tag{5}$$

  where the complex part often cancels out

- Hence,

$$e^{\pm 2\pi i x \omega} = \cos(2\pi x \omega) \pm i \sin(2\pi x \omega) \tag{6}$$



Fourier transforms

- Fourier transform $S(\omega)$ of a function $f(x)$,

$$S(\omega) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x\omega} dx \tag{3}$$

where

  - i is the imaginary number with $i^2 = -1$ and $i^0 = 1$
  - ω is a frequency

- Inverse Fourier transform $f(x)$ of spectral density $S(\omega)$,

$$f(x) = \int_{-\infty}^{\infty} S(\omega) e^{2\pi i x\omega} d\omega \tag{4}$$

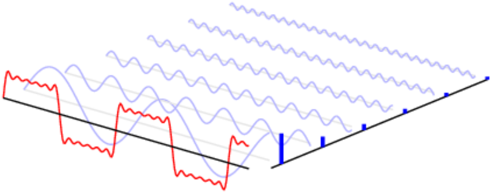
- Euler's identity helps compute Fouriers in practise

$$e^{ix} = \underbrace{\cos x}_{\text{real part}} + \underbrace{i \sin x}_{\text{imaginary part}} \tag{5}$$

where the complex part often cancels out

- Hence,

$$e^{\pm 2\pi i x\omega} = \cos(2\pi x\omega) \pm i \sin(2\pi x\omega) \tag{6}$$



Fourier duals for kernels

- Let's apply Fouriers to the kernel $K(\tau) := K(x, x')$, where $\tau = x - x'$ (instead of $f(x)$)

Theorem (Bochner)

Any stationary kernel $K : \mathbb{R}^D \mapsto \mathbb{R}$ and its spectral density $S : \mathbb{R}^D \mapsto \mathbb{R}$ are Fourier duals

$$K(\tau) = \int_{-\infty}^{\infty} S(\omega) e^{2\pi i \omega^T \tau} d\omega \qquad \text{(Inverse Fourier Transform)}$$

$$S(\omega) = \int_{-\infty}^{\infty} K(\tau) e^{-2\pi i \omega^T \tau} d\tau. \qquad \text{(Fourier Transform)}$$

1. All stationary kernels have spectral density $S(\omega)$
   - If someone gives you a kernel $K(\tau)$, we can solve what frequencies it considers by solving the FT
   - Spectral features are of theoretical interest
2. All spectral densities define a covariance function $K(\tau)$
   - If someone gives you a spectral density $S(\omega)$, we can solve its similarity function (=kernel) by solving the IFT
   - If we change the spectral density, we get a new kernel
   - $\Rightarrow$ kernel learning

Fourier duals for kernels

- Let's apply Fouriers to the kernel $K(\tau) := K(x, x')$, where $\tau = x - x'$ (instead of $f(x)$)

Theorem (Bochner)

Any stationary kernel $K : \mathbb{R}^D \mapsto \mathbb{R}$ and its spectral density $S : \mathbb{R}^D \mapsto \mathbb{R}$ are Fourier duals

$$K(\tau) = \int_{-\infty}^{\infty} S(\omega) e^{2\pi i \omega^T \tau} d\omega \qquad \text{(Inverse Fourier Transform)}$$

$$S(\omega) = \int_{-\infty}^{\infty} K(\tau) e^{-2\pi i \omega^T \tau} d\tau. \qquad \text{(Fourier Transform)}$$

1. All stationary kernels have spectral density $S(\omega)$
   - If someone gives you a kernel $K(\tau)$, we can solve what frequencies it considers by solving the FT
   - Spectral features are of theoretical interest
2. All spectral densities define a covariance function $K(\tau)$
   - If someone gives you a spectral density $S(\omega)$, we can solve its similarity function (=kernel) by solving the IFT
   - If we change the spectral density, we get a new kernel
   - $\Rightarrow$ kernel learning

Fourier duals for kernels

- Let's apply Fouriers to the kernel $K(\tau) := K(x, x')$, where $\tau = x - x'$ (instead of $f(x)$)

Theorem (Bochner)

Any stationary kernel $K : \mathbb{R}^D \mapsto \mathbb{R}$ and its spectral density $S : \mathbb{R}^D \mapsto \mathbb{R}$ are Fourier duals

$$K(\tau) = \int_{-\infty}^{\infty} S(\omega) e^{2\pi i \omega^T \tau} d\omega \qquad \text{(Inverse Fourier Transform)}$$

$$S(\omega) = \int_{-\infty}^{\infty} K(\tau) e^{-2\pi i \omega^T \tau} d\tau. \qquad \text{(Fourier Transform)}$$

1. All stationary kernels have spectral density $S(\omega)$
   - If someone gives you a kernel $K(\tau)$, we can solve what frequencies it considers by solving the FT
   - Spectral features are of theoretical interest
2. All spectral densities define a covariance function $K(\tau)$
   - If someone gives you a spectral density $S(\omega)$, we can solve its similarity function (=kernel) by solving the IFT
   - If we change the spectral density, we get a new kernel
   - $\Rightarrow$ kernel learning

Fourier duals for kernels

- Let's apply Fouriers to the kernel $K(\tau) := K(x, x')$, where $\tau = x - x'$ (instead of $f(x)$)

Theorem (Bochner)

Any stationary kernel $K : \mathbb{R}^D \mapsto \mathbb{R}$ and its spectral density $S : \mathbb{R}^D \mapsto \mathbb{R}$ are Fourier duals

$$K(\tau) = \int_{-\infty}^{\infty} S(\omega) e^{2\pi i \omega^T \tau} d\omega \qquad \text{(Inverse Fourier Transform)}$$

$$S(\omega) = \int_{-\infty}^{\infty} K(\tau) e^{-2\pi i \omega^T \tau} d\tau. \qquad \text{(Fourier Transform)}$$

1. All stationary kernels have spectral density $S(\omega)$
   - If someone gives you a kernel $K(\tau)$, we can solve what frequencies it considers by solving the FT
   - Spectral features are of theoretical interest
2. All spectral densities define a covariance function $K(\tau)$
   - If someone gives you a spectral density $S(\omega)$, we can solve its similarity function (=kernel) by solving the IFT
   - If we change the spectral density, we get a new kernel
   - $\Rightarrow$ kernel learning

Kernel sinusoid representation

- Assume symmetric frequency distribution $S(\omega) = S(-\omega)$
- Euler's identity $e^{\pm ix} = \cos x \pm i \sin x$
- Sine identity $\sin(-x) = -\sin(x)$
- Then we can solve the inverse Fourier as

$$
\begin{aligned}
K(\tau) &= \int_{-\infty}^{\infty} S(\omega) e^{2\pi i \tau \omega} d\omega \\
&= \int_{-\infty}^{\infty} S(\omega) \cos(2\pi\tau\omega) d\omega + \int_{-\infty}^{\infty} i \cdot S(\omega) \sin(2\pi\tau\omega) d\omega \\
&= \mathbb{E}_{S(\omega)} \cos(2\pi\tau\omega) + \int_{-\infty}^{0} i \cdot S(\omega) \sin(2\pi\tau\omega) d\omega + \int_{0}^{\infty} i \cdot S(\omega) \sin(2\pi\tau\omega) d\omega \\
&= \mathbb{E}_{S(\omega)} \cos(2\pi\tau\omega) + \int_{0}^{\infty} iS(-\omega) \sin(2\pi\tau(-\omega)) d\omega + \int_{0}^{\infty} iS(\omega) \sin(2\pi\tau\omega) d\omega \\
&= \mathbb{E}_{S(\omega)} \cos(2\pi\tau\omega) + \int_{0}^{\infty} -iS(\omega) \sin(2\pi\tau\omega) d\omega + \int_{0}^{\infty} iS(\omega) \sin(2\pi\tau\omega) d\omega \\
&= \mathbb{E}_{S(\omega)} \cos(2\pi\tau\omega)
\end{aligned}
$$

- All stationary kernels are linear $S(\omega)$-combinations of $\cos(2\pi\tau\omega)$

Kernel sinusoid representation

- Assume symmetric frequency distribution $S(\omega) = S(-\omega)$
- Euler's identity $e^{\pm ix} = \cos x \pm i \sin x$
- Sine identity $\sin(-x) = -\sin(x)$
- Then we can solve the inverse Fourier as

$$
\begin{aligned}
K(\tau) &= \int_{-\infty}^{\infty} S(\omega) e^{2\pi i \tau \omega} d\omega \\
&= \int_{-\infty}^{\infty} S(\omega) \cos(2\pi\tau\omega) d\omega + \int_{-\infty}^{\infty} i \cdot S(\omega) \sin(2\pi\tau\omega) d\omega \\
&= \mathbb{E}_{S(\omega)} \cos(2\pi\tau\omega) + \int_{-\infty}^{0} i \cdot S(\omega) \sin(2\pi\tau\omega) d\omega + \int_{0}^{\infty} i \cdot S(\omega) \sin(2\pi\tau\omega) d\omega \\
&= \mathbb{E}_{S(\omega)} \cos(2\pi\tau\omega) + \int_{0}^{\infty} iS(-\omega) \sin(2\pi\tau(-\omega)) d\omega + \int_{0}^{\infty} iS(\omega) \sin(2\pi\tau\omega) d\omega \\
&= \mathbb{E}_{S(\omega)} \cos(2\pi\tau\omega) + \int_{0}^{\infty} -iS(\omega) \sin(2\pi\tau\omega) d\omega + \int_{0}^{\infty} iS(\omega) \sin(2\pi\tau\omega) d\omega \\
&= \mathbb{E}_{S(\omega)} \cos(2\pi\tau\omega)
\end{aligned}
$$

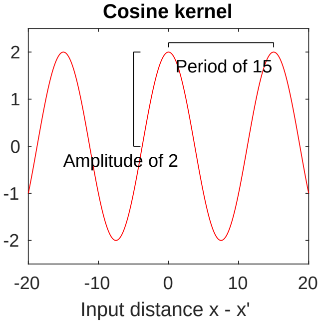
- All stationary kernels are linear $S(\omega)$-combinations of $\cos(2\pi\tau\omega)$

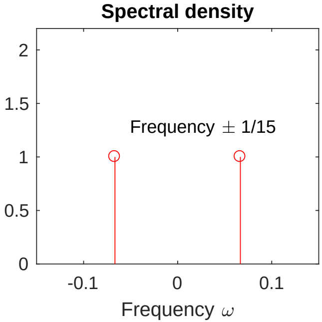
Kernel sinusoid representation

- Our new general kernel definition

$$K(\tau) = \mathbb{E}_{S(\omega)} \cos(2\pi\tau\omega) \tag{7}$$

- Frequency ω is inverse of period $1/\omega$
- Amplitude $S(\omega)$
- Frequencies are symmetric $S(\omega) = S(-\omega)$
- With $S(\omega) = \delta_{1/15}(\omega)$, the kernel becomes $K(\tau) = \cos(2\pi\tau\frac{1}{15})$





Gaussian kernel sinusoids

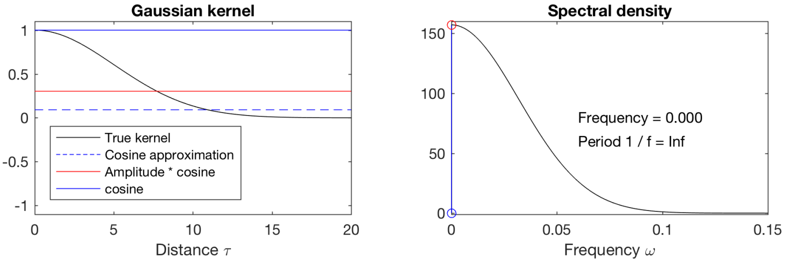
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{8}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{9}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{10}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{11}$$



Gaussian kernel sinusoids

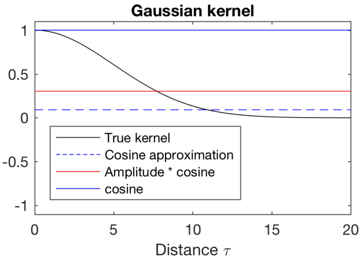
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

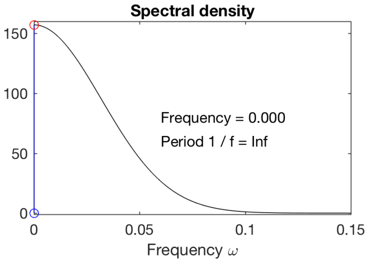
$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$





Gaussian kernel sinusoids

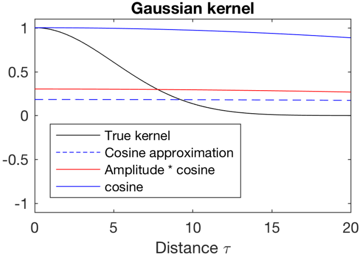
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

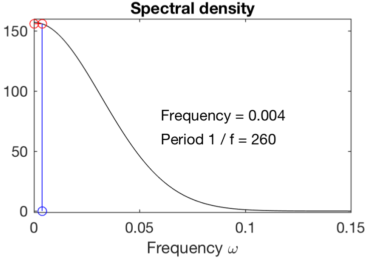
$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2 \ell^2 \omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$





Gaussian kernel sinusoids

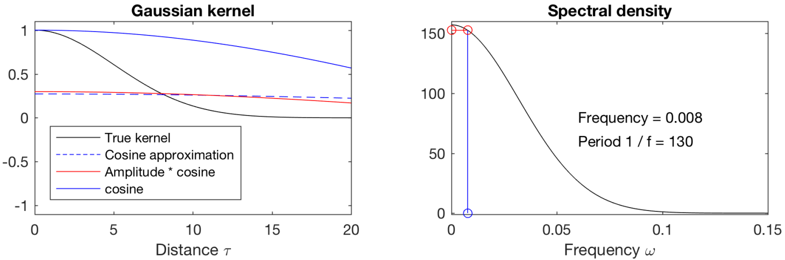
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2 \ell^2 \omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$



Gaussian kernel sinusoids

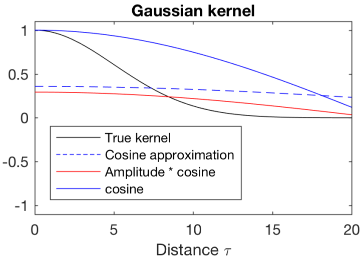
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

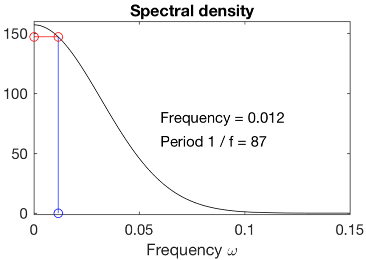
$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$





Gaussian kernel sinusoids

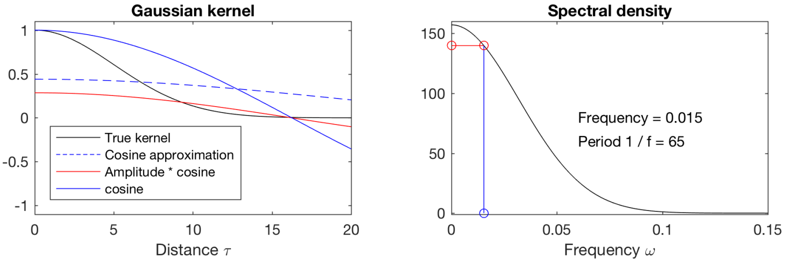
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$



Gaussian kernel sinusoids

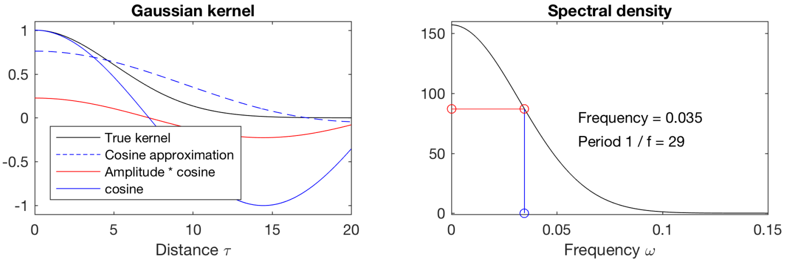
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$



Gaussian kernel sinusoids

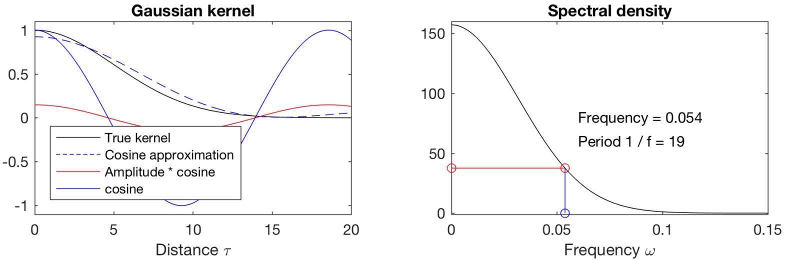
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$



Gaussian kernel sinusoids

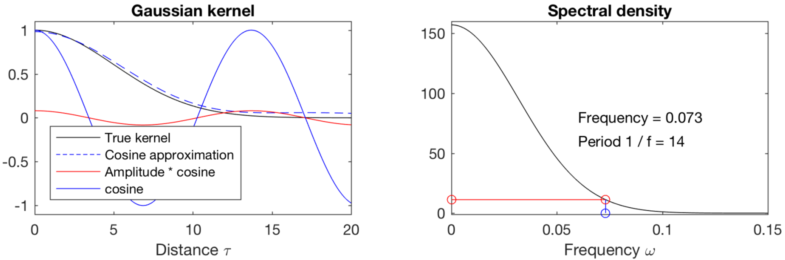
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$



Gaussian kernel sinusoids

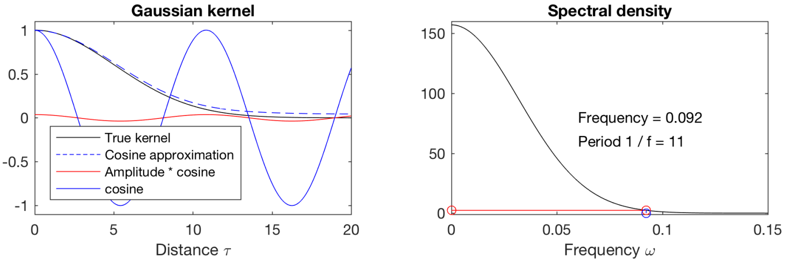
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$



Gaussian kernel sinusoids

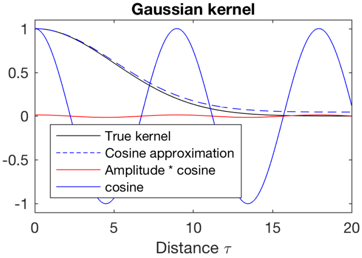
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

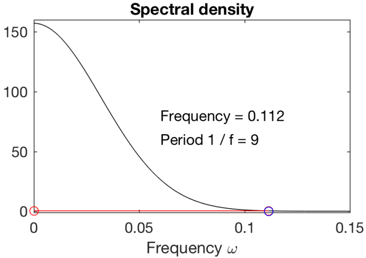
$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$





Gaussian kernel sinusoids

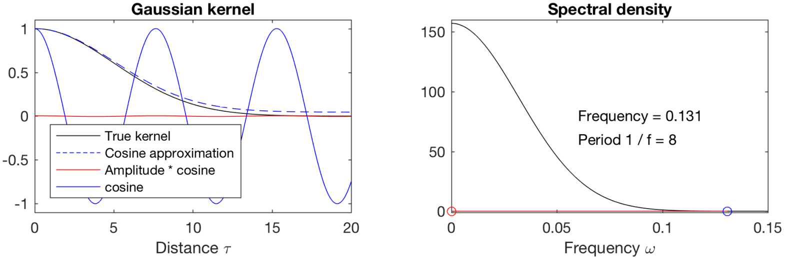
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$



Gaussian kernel sinusoids

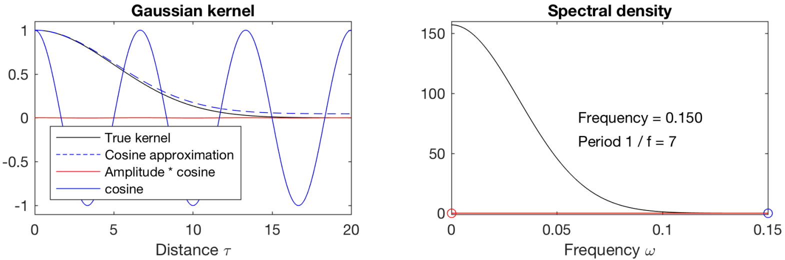
- Gaussian kernel $K_{SE}(\tau) = \exp(-\tau^2/\ell^2)$ fourier representation

$$S_{SE}(\omega) = \int_{-\infty}^{\infty} K_{SE}(\tau) e^{-2\pi i \omega^T \tau} d\tau \tag{12}$$

$$= 2\pi\ell^2 \exp(-2\pi^2\ell^2\omega^2) \tag{13}$$

$$K_{SE}(\tau) = \int_0^{\infty} \underbrace{S_{SE}(\omega)}_{\text{amplitudes}} \cdot \underbrace{\cos(2\pi\tau\omega)}_{\text{sinusoids}} d\omega \tag{14}$$

$$\approx \sum_{\omega} S_{SE}(\omega) \cdot \cos(2\pi\tau\omega) \tag{15}$$

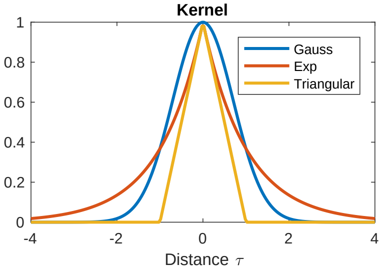


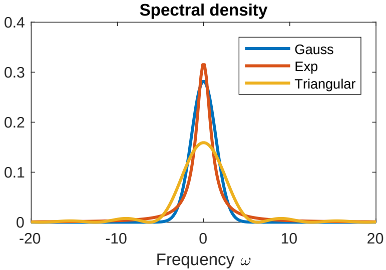
Some spectral densities

$$K_{gauss}(\tau) = \exp(-\frac{\tau^2}{\ell^2}) \qquad S_{gauss}(\omega) = \frac{\sqrt{\ell}}{2\sqrt{\pi}} \exp(-\ell\omega^2/4) \tag{16}$$

$$K_{exp}(\tau) = \exp(-|\tau|/\ell) \qquad S_{exp}(\omega) = 1/(\pi/\ell + \pi\ell\omega^2) \tag{17}$$

$$K_{tri}(\tau) = 0.5(1 - |\tau|)_+ \qquad S_{tri}(\omega) = (1 - \cos\omega)/(\pi\omega^2) \tag{18}$$





- Can we construct new kernels from custom spectral densities?

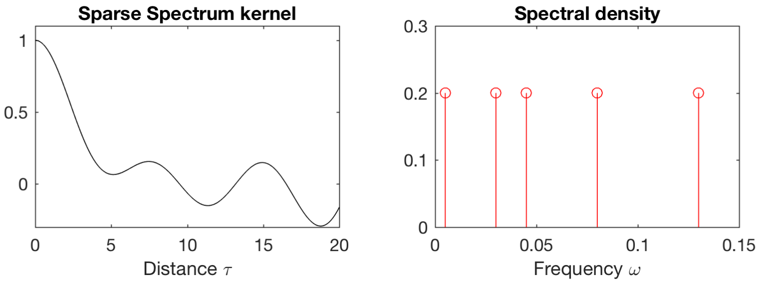
Sparse Spectrum (SS) kernel[1]

- Define Q real frequencies $(\omega_1, \ldots, \omega_Q)^T \in \mathbb{R}^Q$ with Fourier dual

$$S(\omega) := \frac{1}{Q} \sum_{i=1}^{Q} \delta(\omega = \omega_i) \tag{19}$$

$$\xrightarrow{\text{IFT}} K(\tau) = \frac{1}{Q} \sum_{i=1}^{Q} \cos(2\pi\tau\omega_i) \tag{20}$$

- No decay, prone to overfitting



[1] Lazaro-Gredilla et al (JMLR 2010) Sparse spectrum gaussian process regression

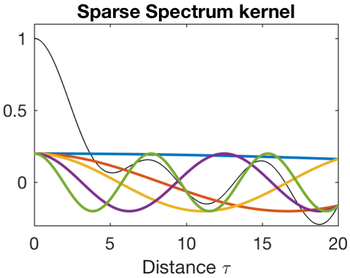
Sparse Spectrum (SS) kernel[1]

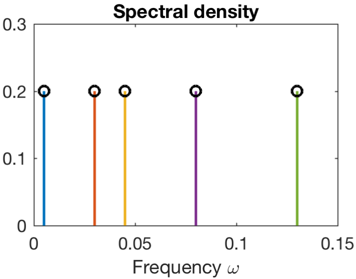
- Define Q real frequencies $(\omega_1, \ldots, \omega_Q)^T \in \mathbb{R}^Q$ with Fourier dual

$$S(\omega) := \frac{1}{Q} \sum_{i=1}^{Q} \delta(\omega = \omega_i) \tag{19}$$

$$\xrightarrow{\text{IFT}} K(\tau) = \frac{1}{Q} \sum_{i=1}^{Q} \cos(2\pi\tau\omega_i) \tag{20}$$

- No decay, prone to overfitting





[1] Lazaro-Gredilla et al (JMLR 2010) Sparse spectrum gaussian process regression

Spectral Mixture (SM) kernel[2]

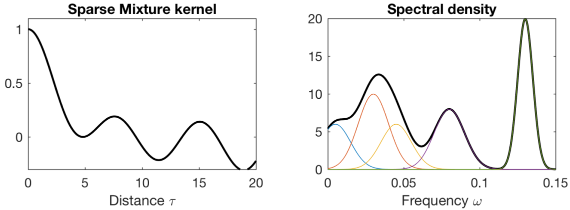
- Define mixture of Q Gaussians $\{a_i \mathcal{N}(\mu_i, \sigma_i^2)\}_{i=1}^{Q}$

$$S(\omega) := \sum_{i=1}^{Q} a_i \mathcal{N}(\omega|\mu_i, \sigma_i^2) \tag{21}$$

$$\xRightarrow{\text{IFT}} K(\tau) = \int_{-\infty}^{\infty} S(\omega) \cos(2\pi\tau\omega) d\omega \tag{22}$$

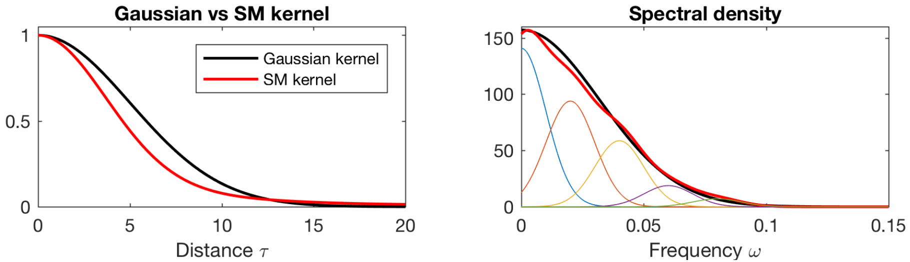
$$= \sum_{i=1}^{Q} a_i \underbrace{\exp(-2\pi^2 \sigma_i^2 \tau^2)}_{\text{smooth decay}} \underbrace{\cos(2\pi\tau\mu_i)}_{\text{periodic}} \tag{23}$$

- Dense in the set of stationary kernels $\Rightarrow$ can generate any stationary kernel



[2]Wilson, Adams (ICML 2013) Gaussian process kernels for pattern discovery and extrapolation

Spectral Mixture (SM) kernel

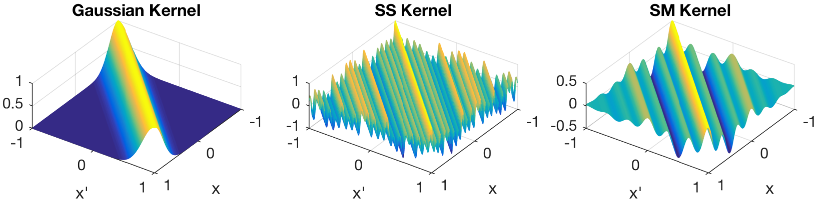


- Approximate gaussian kernel with SM kernel with $Q = 5$ components, i.e.

$$\sum_{i=1}^{Q} a_i \exp(-2\pi^2 \sigma_i^2 \tau^2) \cos(2\pi\tau\mu_i) \approx \exp\left(-\frac{(x - x')^2}{2\ell^2}\right)$$

  for appropriate $\{a_i, \mu_i, \sigma_i\}$
- (Doable with $Q = 1$ as well)

Spectral kernels



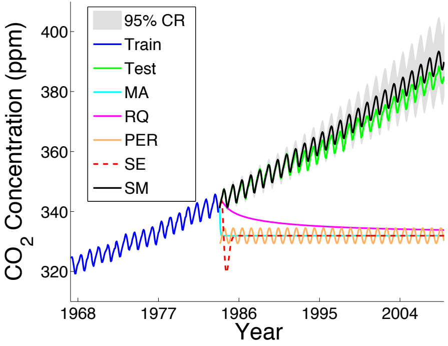
- Image from Remes, Heinonen, Kaski: Non-stationary spectral kernels, NIPS’17

SM kernel inference

- Optimize $3Q$ hyperparameters $\theta = \{a_i, \mu_i, \sigma_i\}_{i=1}^{Q}$ of kernel $K_\theta(x - x') = \sum_{i=1}^{Q} a_i \exp(-2\pi^2 {\sigma_i}^2 \tau^2) \cos(2\pi\tau\mu_i)$ by maximizing

$$\log p(\mathbf{y}|\theta) = -\frac{1}{2} \underbrace{\mathbf{y}^T (K_\theta + \sigma^2 I)^{-1} \mathbf{y}}_{\text{data fit}} - \frac{1}{2} \underbrace{\log |K_\theta + \sigma^2 I|}_{\text{model complexity}} - \frac{N}{2} \log 2\pi$$

- After kernel is fixed, predictions have closed form



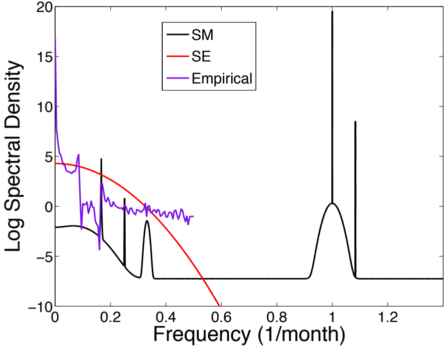
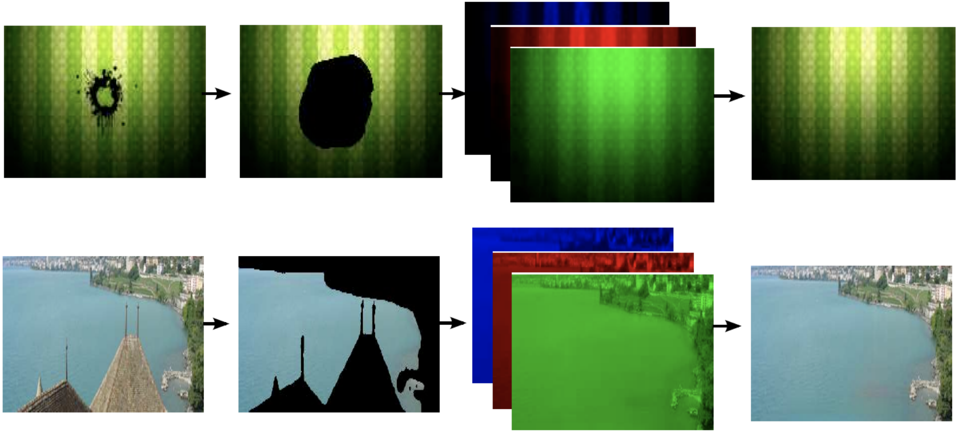
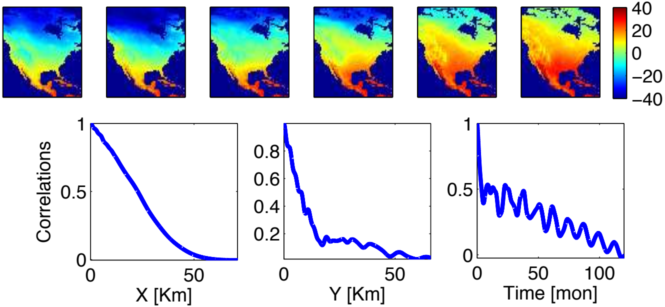


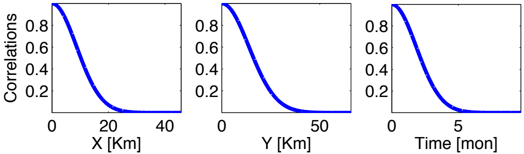
Image inpainting use case



Spatio-temporal temperatures



(a) Learned GPatt Kernel for Temperatures



(b) Learned GP-SE Kernel for Temperatures

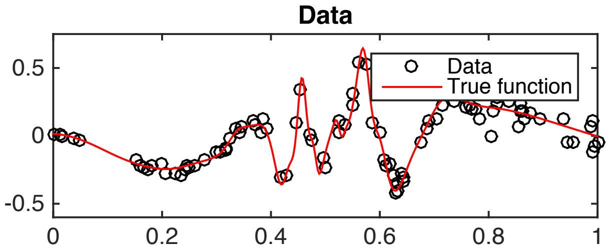
- SM kernel induces only stationary covariances, but temperatures are non-stationary

Stationary kernels

- Stationary kernels are translation-invariant:

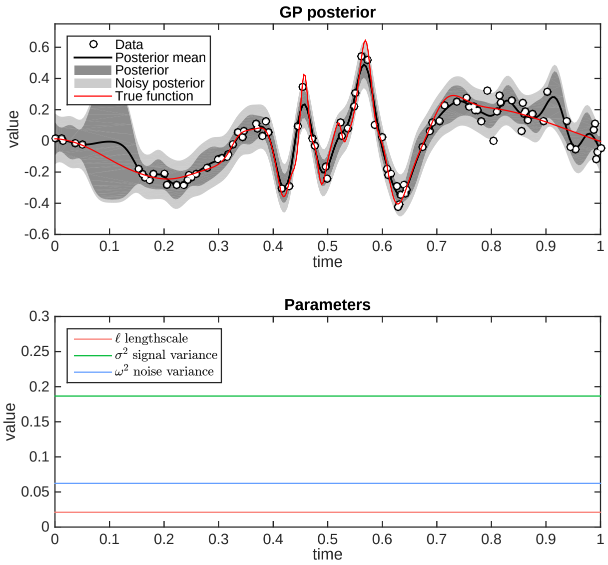
$$K(x, x') = K(x + a, x' + a) \tag{24}$$
$$K(x, x') = K(x - x') \tag{25}$$

  for any a
  - Stationary kernels are function of vector distance $x - x'$
  - For instance if input variable is ‘age’ in years, then a stationary kernel has property $K(1, 2) = K(80, 81)$
  - Strange to assume that 1 and 2 year olds are as similar to each other as 80 and 81 year olds
- Non-stationary kernel is not translation invariant, i.e. we can have $K(1, 2) \neq K(80, 81)$
- Simplest non-stationary kernel is the dot product, $K(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T\mathbf{x}$ since
  - $\mathbf{x} = [1, 1]^T$, $\mathbf{x}' = [2, 2]$, $K(\mathbf{x}, \mathbf{x}') = 1 \cdot 2 + 1 \cdot 2 = 4$
  - $\mathbf{x} = [10, 10]^T$, $\mathbf{x}' = [11, 11]$, $K(\mathbf{x}, \mathbf{x}') = 10 \cdot 11 + 10 \cdot 11 = 120$



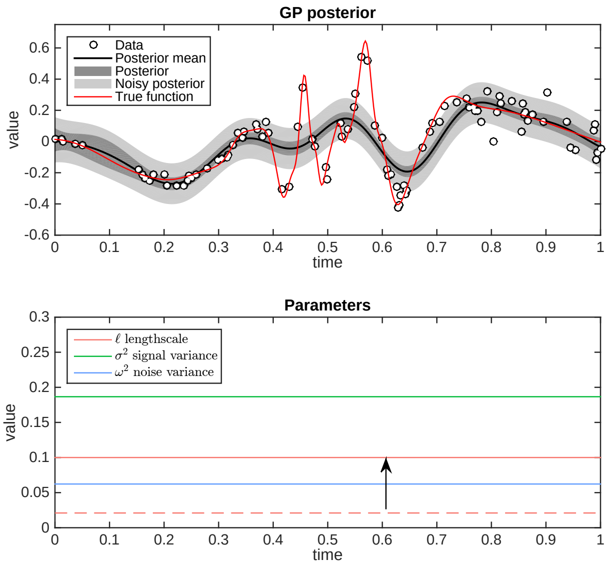
- Simple dataset

Problem with stationary functions



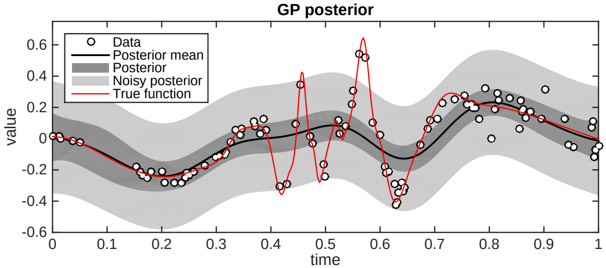
- Optimal Gaussian process fit
- Bad fit in the beginning

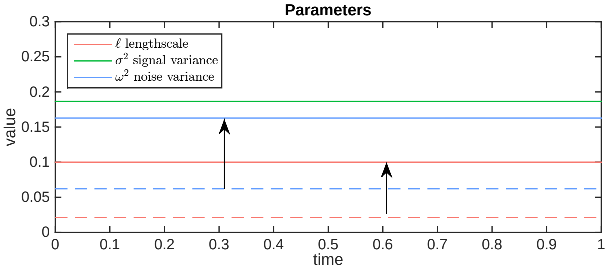
Problem with stationary functions



- Let's increase lengthscale to get smoother model
- Initial fit fixed, now ill fit in the middle

Problem with stationary functions





- Let's increase noise level to to match data
- $\Rightarrow$ We need input-dependent parameters

Non-stationary Gaussian process[3]

- The Gaussian kernel has a fixed, global lengthscale

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{||\mathbf{x} - \mathbf{x}'||^2}{2\ell^2}\right) \tag{26}$$

  - Equally smooth functions everywhere
- The non-stationary Gaussian kernel ('Gibbs kernel') admits a lengthscale function $\ell(x)$

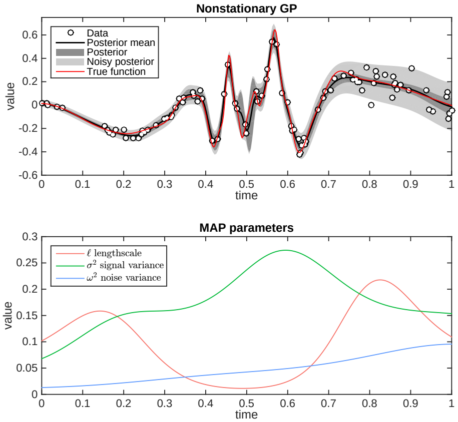
$$K(x, x') = \underbrace{\sqrt{\frac{2\ell(x)\ell(x')}{\ell(x)^2 + \ell(x')^2}}}_{\text{normalizer}} \exp\left(-\frac{(x - x')^2}{\ell(x)^2 + \ell(x')^2}\right) \tag{27}$$

- The multivariate Gibbs kernel, where $\Sigma_i := \Sigma(\mathbf{x}_i) \in \mathbb{R}^{D \times D}$

$$K(\mathbf{x}_i, \mathbf{x}_j) = |\Sigma_i|^{1/4}|\Sigma_j|^{1/4}|(\Sigma_i + \Sigma_j)/2|^{-1/2} \exp\left(-(\mathbf{x}_i - \mathbf{x}_j)^T((\Sigma_i + \Sigma_j)/2)^{-1}(\mathbf{x}_i - \mathbf{x}_j)\right) \tag{28}$$

[3] Paciorek, Schervish (NIPS 2004): Nonstationary Covariance Functions for Gaussian Process Regression

Non-stationary solution[4]



- Function process

$$y(x) = f(x) + \varepsilon(x) \tag{29}$$

$$f(x) \sim \mathcal{GP}(0, \sigma(x)\sigma(x')K_{\ell(\cdot)}(x, x')) \tag{30}$$

$$\varepsilon(x) \sim \mathcal{N}(0, \omega(x)^2) \tag{31}$$

- Parameter processes

$$\ell(x) \sim \mathcal{GP}(\mu_\ell, K_\ell(x, x')) \tag{32}$$

$$\sigma(x) \sim \mathcal{GP}(\mu_\sigma, K_\sigma(x, x')) \tag{33}$$

$$\omega(x) \sim \mathcal{GP}(\mu_\omega, K_\omega(x, x')) \tag{34}$$

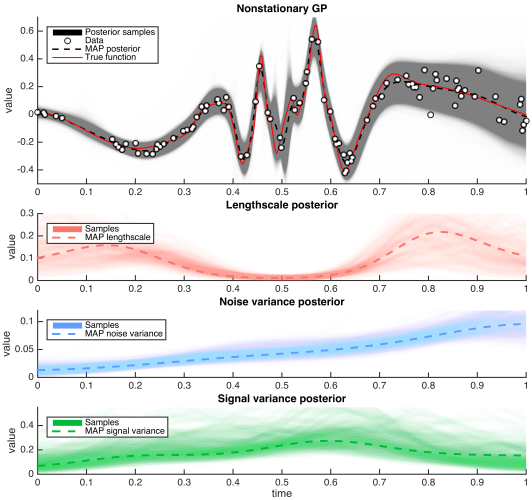
- Kernel

$$K(x, x') = \sqrt{\frac{2\ell(x)\ell(x')}{\ell(x)^2 + \ell(x')^2}} \exp\left(-\frac{(x - x')^2}{\ell(x)^2 + \ell(x')^2}\right) \tag{35}$$

- Explicit function representation through smoothness, scale and noise functions

[4] Heinonen, Mannerström, Rousu, Kaski, Lähdesmäki (AISTATS 2016): Non-stationary Gaussian process regression with Hamiltonian Monte Carlo

MCMC inference



- Sample exact posterior with HMC[5]

$$p(\mathbf{f}, \boldsymbol{\ell}, \boldsymbol{\sigma}, \boldsymbol{\omega}; \mathbf{y})$$

[5] Heinonen et al. Non-stationary Gaussian process regression with Hamiltonian Monte Carlo. AISTATS 2016

Generalised Spectral Mixture (GSM) kernel[6,7]

- Non-stationary spectral kernel:

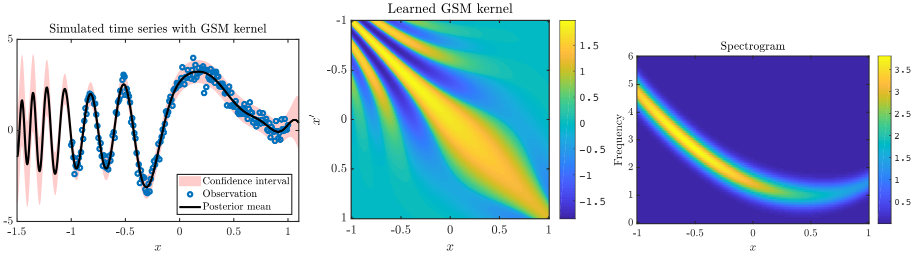
$$K_{\mathbf{w},\boldsymbol{\mu},\boldsymbol{\sigma}}(x,x') \propto \sum_{i=1}^{Q} {\color{red} w_i(x) w_i(x')} \underbrace{\exp\left(-\frac{(x-x')^2}{{\color{green}\ell_i(x)}^2 + {\color{green}\ell_i(x')}^2)}\right)}_{\text{Exponential kernel}} \underbrace{\cos(2\pi({\color{blue}\mu_i(x)}x - {\color{blue}\mu_i(x')}x'))}_{\text{periodic}}$$

with

$$\log {\color{red} w_i(x)} \sim \mathcal{GP}(0, K_w) \tag{36}$$

$$\log {\color{blue} \mu_i(x)} \sim \mathcal{GP}(0, K_\mu) \tag{37}$$

$$\log {\color{green} \ell_i(x)} \sim \mathcal{GP}(0, K_\sigma) \tag{38}$$



[6] Remes, Heinonen, Kaski (NIPS 2017): Non-stationary spectral kernels
[7] Shen, Heinonen, Kaski (AISTATS 2019): Harmonizable mixture kernels with variational Fourier features

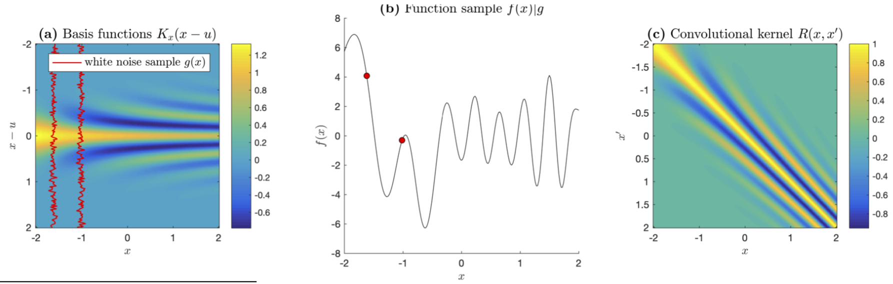
- A Gaussian process can be represented as a convolution over $\mathbf{x}, \mathbf{u} \in \mathcal{X}$,

$$f(\mathbf{x}_i) = \int K_{\mathbf{x}_i}(\mathbf{u}) g(\mathbf{u}) d\mathbf{u} \tag{39}$$

  - Feature map $K_{\mathbf{x}_i}$
  - White noise process $g(\mathbf{u}) \sim \mathcal{GP}(0, \delta_{\mathbf{x}=\mathbf{x}'})$
- The kernel becomes[8]

$$C(\mathbf{x}_i, \mathbf{x}_j) = \int K_{\mathbf{x}_i}(\mathbf{u}) \overline{K}_{\mathbf{x}_j}(\mathbf{u}) d\mathbf{u} \tag{40}$$

where $\overline{K}$ is complex conjugate



[8] Shen, Heinonen, Kaski (AISTATS 2020): Learning spectrograms with convolutional spectral kernels

Convolutional kernel family[9]

- Gaussian kernel

$$K_{\mathbf{x}_i}(\mathbf{u}) \propto \mathcal{N}(\mathbf{u}|\mathbf{x}_i, \boldsymbol{\Sigma}) \tag{41}$$

- A non-stationary Gaussian kernel

$$K_{\mathbf{x}_i}(\mathbf{u}) \propto \mathcal{N}(\mathbf{u}|\mathbf{x}_i, \boldsymbol{\Sigma}(\mathbf{x}_i)) \tag{42}$$

- Spectral mixture kernel

$$K_{\mathbf{x}_i}(\mathbf{u}) \propto \mathcal{N}(\mathbf{u}|\mathbf{x}_i + i\boldsymbol{\mu}, \boldsymbol{\Sigma}) \tag{43}$$

- Non-stationary spectral mixture kernel

$$K_{\mathbf{x}_i}(\mathbf{u}) \propto \mathcal{N}(\mathbf{u}|\mathbf{x}_i + i\boldsymbol{\mu}(\mathbf{x}_i), \boldsymbol{\Sigma}(\mathbf{x}_i)) \tag{44}$$

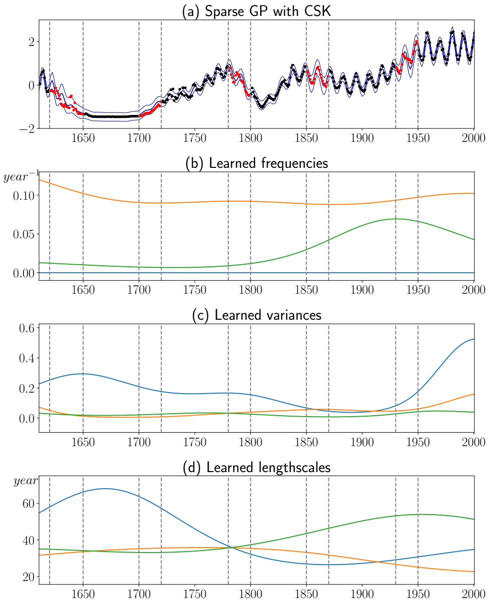
  - Input-dependent frequencies $\boldsymbol{\mu}_i$
  - Input-dependent Gaussian covariance $\boldsymbol{\Sigma}_i$

[9] Shen, Heinonen, Kaski (AISTATS 2020): Learning spectrograms with convolutional spectral kernels

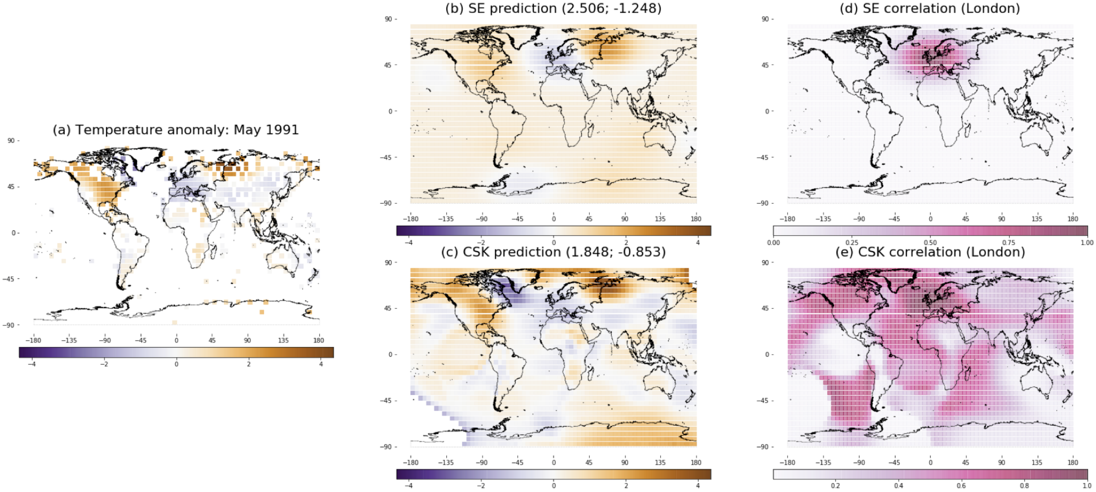
Convolutional spectral kernel

- We parameterise frequencies with smooth GP's and learn point estimates,

$$
\begin{aligned}
w^q(\cdot) &\sim \mathcal{GP}(c_w, k_{SE}(\cdot,\cdot)) \\
\mathrm{logit}\boldsymbol{\mu}^q(\cdot) &\sim \mathcal{GP}(c_\mu, k_{SE}(\cdot,\cdot)) \\
(\boldsymbol{\Lambda}^q)^{1/2}(\cdot) &\sim \mathcal{GP}(c_\lambda, k_{SE}(\cdot,\cdot))
\end{aligned}
$$



Spatial interpolation with non-stationary spectral kernel



- The non-stationary spectral kernel family is extremely flexible

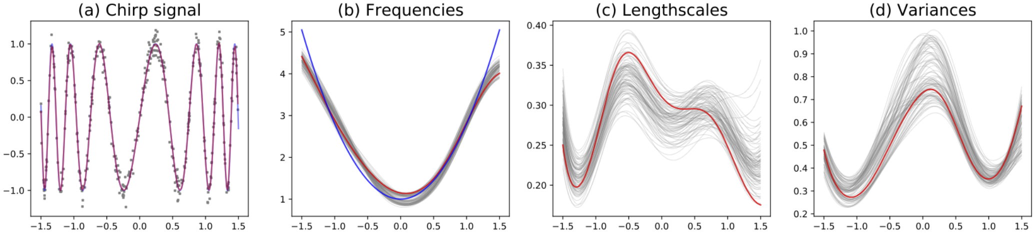
How to learn an overly flexible kernel?

- Marginal log likelihood matches likelihood $p(\mathbf{y}|\mathbf{f})$ and prior $p(\mathbf{f}|\theta)$

$$\log p(\mathbf{y}|\theta) = \log \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\theta)d\mathbf{f} \quad (45)$$

$$= \log \mathbb{E}_{p(\mathbf{f})} p(\mathbf{y}|\mathbf{f}) \quad (46)$$

- Risk overfitting if $p(\mathbf{f}) \approx p(\mathbf{y}|\mathbf{f})$ is possible
    - Consider function space that only contains training-data like functions
- Remedies
    - MCMC sampling of $p(\mathbf{y}|\theta)$
    - Hyperpriors $p(\theta)$



Summary

- The kernel choice defines how well the GP performs
- Gaussian kernel is a convenient 'default' kernel that can interpolate well
  - Advantage: simple, efficient, easy-to-learn, universal
  - Disadvantage: cannot fit periodic, "long-range" or non-stationary signals
- Spectral kernels can extrapolate repeating patterns
  - Advantage: can learn arbitrary periodic or non-periodic stationary patterns
  - Disadvantage: slow, possibility to overfit
- Non-stationary spectral kernels can learn adaptive interpolations
  - Advantage: can learn evolving frequencies
  - Disadvantage: slow, more possibilities to overfit

Thanks!

Appendix

Convolutional spectral kernel

- Convolutional spectral kernel (CSK) with Gaussian/periodic feature map

$$K_{\mathbf{x}_i}(\mathbf{u}) = \sum_{q=1}^{Q} w_i^q \exp(-2\pi^2 S_i^q + 2\pi i \theta_i^q) \sim \sum_{q=1}^{Q} \mathcal{N}(i\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \tag{47}$$

  where $S_i^q = (\mathbf{x}_i - \mathbf{u})^T \boldsymbol{\Lambda}_i^q (\mathbf{x}_i - \mathbf{u})$ and $\theta_i^q = \langle \boldsymbol{\mu}_i^q, \mathbf{x}_i - \mathbf{u} \rangle$
- The kernel can be solved with $\boldsymbol{\Sigma}_i = \boldsymbol{\Lambda}_i^{-1}$

$$k(\mathbf{x}_i, \mathbf{x}_j) = \int_{\mathbb{R}^D} K_{\mathbf{x}_i}(\mathbf{u}) K_{\mathbf{x}_j}(\mathbf{u}) d\mathbf{u} \tag{48}$$

$$= \sum_{q,p=1}^{Q} \frac{w_i^p \overline{w}_j^q}{(2\pi)^{D/2} |\boldsymbol{\Lambda}_i^p + \boldsymbol{\Lambda}_j^q|^{1/2}} \exp(-\pi^2 S_{ij}^{pq} + 2\pi i \theta_{ij}^{pq} - R_{ij}^{pq}) \tag{49}$$

  where

$$R_{ij}^{pq} = (\boldsymbol{\mu}_i^p - \boldsymbol{\mu}_j^q)^T ((\boldsymbol{\Lambda}_i^p + \boldsymbol{\Lambda}_j^q)/2)^{-1} (\boldsymbol{\mu}_i^p - \boldsymbol{\mu}_j^q) \tag{50}$$

$$S_{ij}^{pq} = (\mathbf{x}_i - \mathbf{x}_j)^T ((\boldsymbol{\Sigma}_i^p + \boldsymbol{\Sigma}_j^q)/2)^{-1} (\mathbf{x}_i - \mathbf{x}_j) \tag{51}$$

$$\theta_{ij}^{pq} = \langle \boldsymbol{\Lambda}_i^p (\boldsymbol{\Lambda}_i^p + \boldsymbol{\Lambda}_j^q)^{-1} \boldsymbol{\mu}_i^p + \boldsymbol{\Lambda}_j^q (\boldsymbol{\Lambda}_i^p + \boldsymbol{\Lambda}_j^q)^{-1} \boldsymbol{\mu}_j^q, \mathbf{x}_i - \mathbf{x}_j \rangle \tag{52}$$