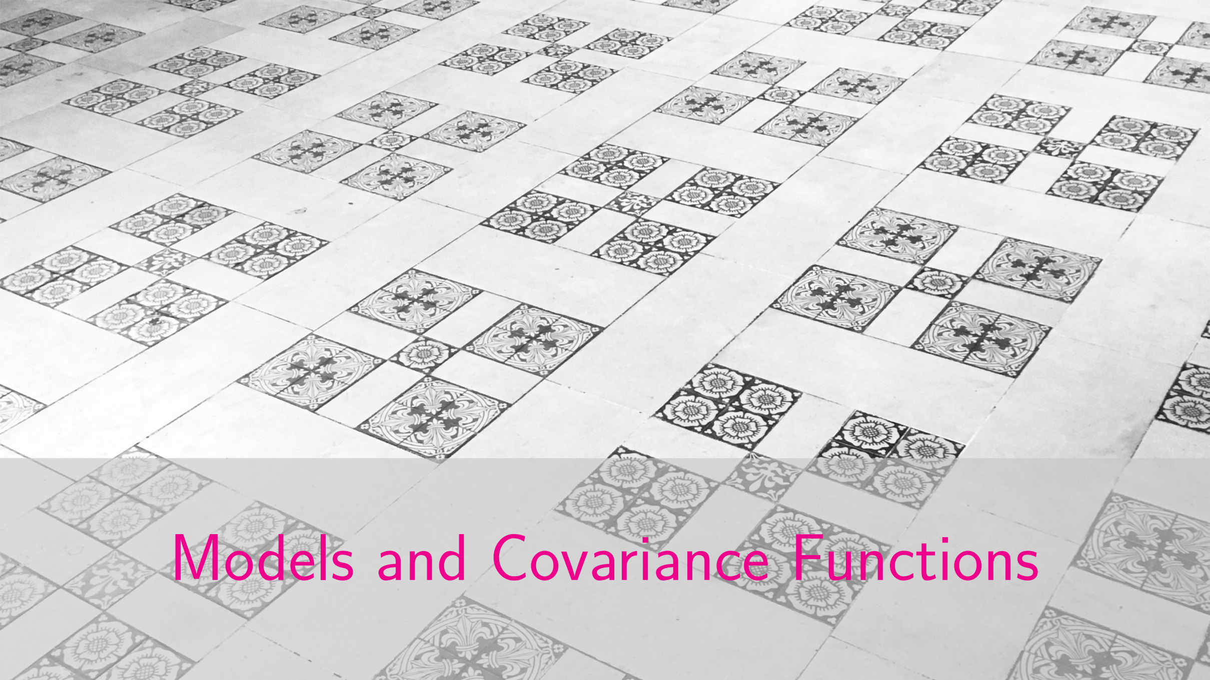
Using Gaussian Processes: Models, Applications, and Connections

Richard E. Turner
University of Cambridge

Outline of the tutorial

- **An Introduction to GPs**
  - Mathematical foundations
  - Hyper-parameter learning
  - Covariance functions
  - Multi-dimensional inputs
- **Using GPs: Models, Applications and Connections**
  - Models and more on covariance functions
  - Applications
  - Connections
- **GPs for large data and non-linear models**
  - Scaling through pseudo-data
  - Variational Inference
  - General Approximate inference



Models and Covariance Functions

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

3. Random sinusoid model

$$h(x) = a\cos(\omega t) + b\sin(\omega t)$$

$$a \sim \mathcal{N}(0, \sigma^2)$$
$$b \sim \mathcal{N}(0, \sigma^2)$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = \mathbb{E}_{f_1, f_2}[f(x)]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = \mathbb{E}_{f_1, f_2}[f(x)] = \mathbb{E}_{f_1, f_2}[f_1(x) + f_2(x)]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = \mathbb{E}_{f_1,f_2}[f(x)] = \mathbb{E}_{f_1,f_2}[f_1(x) + f_2(x)] = \mathbb{E}_{f_1}[f_1(x)] + \mathbb{E}_{f_2}[f_2(x)]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = \mathbb{E}_{f_1,f_2}[f(x)] = \mathbb{E}_{f_1,f_2}[f_1(x) + f_2(x)] = \mathbb{E}_{f_1}[f_1(x)] + \mathbb{E}_{f_2}[f_2(x)]$$
$$= 0 + 0$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = \mathbb{E}_{f_1,f_2}[f(x)] = \mathbb{E}_{f_1,f_2}[f_1(x) + f_2(x)] = \mathbb{E}_{f_1}[f_1(x)] + \mathbb{E}_{f_2}[f_2(x)]$$
$$= 0 + 0 = 0$$

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = 0$$

$$\Sigma(x, x') = \mathbb{E}_{f_1, f_2}[f(x) f(x')]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = 0$$

$$\Sigma(x, x') = \mathbb{E}_{f_1, f_2}[f(x)f(x')] = \mathbb{E}_{f_1, f_2}[(f_1(x) + f_2(x))(f_1(x') + f_2(x'))]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = 0$$

$$\Sigma(x, x') = \mathbb{E}_{f_1, f_2}[f(x)f(x')] = \mathbb{E}_{f_1, f_2}[(f_1(x) + f_2(x))(f_1(x') + f_2(x'))]$$
$$= \mathbb{E}_{f_1, f_2}[f_1(x)f_1(x') + f_1(x)f_2(x') + f_1(x')f_2(x) + f_2(x)f_2(x')]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = 0$$

$$\Sigma(x, x') = \mathbb{E}_{f_1,f_2}[f(x)f(x')] = \mathbb{E}_{f_1,f_2}[(f_1(x) + f_2(x))(f_1(x') + f_2(x'))]$$
$$= \mathbb{E}_{f_1,f_2}[f_1(x)f_1(x') + f_1(x)f_2(x') + f_1(x')f_2(x) + f_2(x)f_2(x')]$$
$$= \mathbb{E}_{f_1,f_2}[f_1(x)f_1(x')] + \mathbb{E}_{f_1,f_2}[f_2(x)f_2(x')]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = 0$$

$$\Sigma(x, x') = \mathbb{E}_{f_1, f_2}[f(x)f(x')] = \mathbb{E}_{f_1, f_2}[(f_1(x) + f_2(x))(f_1(x') + f_2(x'))]$$
$$= \mathbb{E}_{f_1, f_2}[f_1(x)f_1(x') + f_1(x)f_2(x') + f_1(x')f_2(x) + f_2(x)f_2(x')]$$
$$= \mathbb{E}_{f_1, f_2}[f_1(x)f_1(x')] + \mathbb{E}_{f_1, f_2}[f_2(x)f_2(x')] = \Sigma_1(x, x') + \Sigma_2(x, x')$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = 0$$

addition of functions <=> addition of mean and covariance

$$\Sigma(x, x') = \Sigma_1(x, x') + \Sigma_2(x, x')$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

1. Addition of two GPs

$$f(x) = f_1(x) + f_2(x)$$

$$f_1(x) \sim \mathcal{GP}(0, \Sigma_1(x, x'))$$
$$f_2(x) \sim \mathcal{GP}(0, \Sigma_2(x, x'))$$

Gaussians are closed under addition: so are GPs

$$m(x) = 0$$

addition of functions <=> addition of mean and covariance

$$\Sigma(x, x') = \Sigma_1(x, x') + \Sigma_2(x, x')$$

More generally: GPs closed under linear transformation / combination:

GP multiplied by a deterministic function = GP,
derivatives of GP = GP, integral of a GP = GP,
convolution of a GP by a deterministic function = GP

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

Gaussians are closed under linear transformations: so are GPs

Which are GPs? Compute the GPs mean and covariance functions.

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = \mathbb{E}_{m,c}[g(x)]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = \mathbb{E}_{m,c}[g(x)] = \mathbb{E}_{m,c}[mx + c]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = \mathbb{E}_{m,c}[g(x)] = \mathbb{E}_{m,c}[mx + c] = 0$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = \mathbb{E}_{m,c}[g(x)] = \mathbb{E}_{m,c}[mx + c] = 0$$

$$\Sigma(x, x') = \mathbb{E}_{m,c}[g(x)g(x')]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = \mathbb{E}_{m,c}[g(x)] = \mathbb{E}_{m,c}[mx + c] = 0$$

$$\Sigma(x, x') = \mathbb{E}_{m,c}[g(x)g(x')] = \mathbb{E}_{m,c}[(mx+c)(mx'+c)]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = \mathbb{E}_{m,c}[g(x)] = \mathbb{E}_{m,c}[mx + c] = 0$$

$$\Sigma(x, x') = \mathbb{E}_{m,c}[g(x)g(x')] = \mathbb{E}_{m,c}[(mx+c)(mx'+c)] = \mathbb{E}_m[m^2]x\, x' + \mathbb{E}_c[c^2]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = \mathbb{E}_{m,c}[g(x)] = \mathbb{E}_{m,c}[mx + c] = 0$$

$$\Sigma(x, x') = \mathbb{E}_{m,c}[g(x)g(x')] = \mathbb{E}_{m,c}[(mx+c)(mx'+c)] = \mathbb{E}_m[m^2]x\,x' + \mathbb{E}_c[c^2]$$
$$= \sigma_m^2 x\,x' + \sigma_c^2$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

2. Random linear model

$$g(x) = mx + c$$

$$m \sim \mathcal{N}(0, \sigma_m^2)$$
$$c \sim \mathcal{N}(0, \sigma_c^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = 0$$

$$\Sigma(x, x') = \sigma_m^2 x\, x' + \sigma_c^2$$

GPs encompass Bayesian linear regression
Not all GPs are non-parametric (infinite numbers of parameters)

Which are GPs? Compute the GPs mean and covariance functions.

3. Random sinusoid model

$$h(x) = a\cos(\omega t) + b\sin(\omega t)$$

$$a \sim \mathcal{N}(0, \sigma^2)$$
$$b \sim \mathcal{N}(0, \sigma^2)$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

3. Random sinusoid model

$$a \sim \mathcal{N}(0, \sigma^2)$$

$$h(x) = a\cos(\omega t) + b\sin(\omega t)$$

$$b \sim \mathcal{N}(0, \sigma^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = 0$$

$$\Sigma(x, x') = \mathbb{E}_{m,c}[h(x)h(x')]$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

3. Random sinusoid model

$$h(x) = a\cos(\omega t) + b\sin(\omega t)$$

$$a \sim \mathcal{N}(0, \sigma^2)$$
$$b \sim \mathcal{N}(0, \sigma^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = 0$$

$$\begin{aligned}\Sigma(x, x') &= \mathbb{E}_{m,c}[h(x)h(x')] \\ &= \mathbb{E}_{a,b}[(a\cos(\omega x)+b\sin(\omega x))(a\cos(\omega x')+b\sin(\omega x'))]\end{aligned}$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

3. Random sinusoid model

$$h(x) = a\cos(\omega t) + b\sin(\omega t)$$

$$a \sim \mathcal{N}(0, \sigma^2)$$
$$b \sim \mathcal{N}(0, \sigma^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = 0$$

$$\begin{aligned}\Sigma(x, x') &= \mathbb{E}_{m,c}[h(x)h(x')] \\ &= \mathbb{E}_{a,b}[(a\cos(\omega x)+b\sin(\omega x))(a\cos(\omega x')+b\sin(\omega x'))] \\ &= \mathbb{E}_a[a^2]\cos(\omega x)\cos(\omega x') + \mathbb{E}_b[b^2]\sin(\omega x)\sin(\omega x')\end{aligned}$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

3. Random sinusoid model

$$h(x) = a\cos(\omega t) + b\sin(\omega t)$$

$$a \sim \mathcal{N}(0, \sigma^2)$$
$$b \sim \mathcal{N}(0, \sigma^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = 0$$

$$\begin{aligned}
\Sigma(x, x') &= \mathbb{E}_{m,c}[h(x)h(x')] \\
&= \mathbb{E}_{a,b}[(a\cos(\omega x) + b\sin(\omega x))(a\cos(\omega x') + b\sin(\omega x'))] \\
&= \mathbb{E}_a[a^2]\cos(\omega x)\cos(\omega x') + \mathbb{E}_b[b^2]\sin(\omega x)\sin(\omega x') \\
&= \sigma^2\cos(\omega x)\cos(\omega x') + \sigma^2\sin(\omega x)\sin(\omega x')
\end{aligned}$$

Which are GPs? Compute the GPs mean and covariance functions.

3. Random sinusoid model

$$h(x) = a\cos(\omega t) + b\sin(\omega t)$$

$$a \sim \mathcal{N}(0, \sigma^2)$$
$$b \sim \mathcal{N}(0, \sigma^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = 0$$

$$\begin{aligned}\Sigma(x, x') &= \mathbb{E}_{m,c}[h(x)h(x')] \\ &= \mathbb{E}_{a,b}[(a\cos(\omega x)+b\sin(\omega x))(a\cos(\omega x')+b\sin(\omega x'))] \\ &= \mathbb{E}_a[a^2]\cos(\omega x)\cos(\omega x') + \mathbb{E}_b[b^2]\sin(\omega x)\sin(\omega x') \\ &= \sigma^2\cos(\omega x)\cos(\omega x') + \sigma^2\sin(\omega x)\sin(\omega x') = \sigma^2\cos(\omega(x - x'))\end{aligned}$$

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

3. Random sinusoid model

$$h(x) = a\cos(\omega t) + b\sin(\omega t)$$

$$a \sim \mathcal{N}(0, \sigma^2)$$

$$b \sim \mathcal{N}(0, \sigma^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = 0$$

$$\Sigma(x, x') = \sigma^2 \cos(\omega(x - x'))$$

GPs can model periodic structure

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

3. Random sinusoid model

$$h(x) = a\cos(\omega t) + b\sin(\omega t)$$

$$a \sim \mathcal{N}(0, \sigma^2)$$

$$b \sim \mathcal{N}(0, \sigma^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = 0$$

$$\Sigma(x, x') = \sigma^2 \cos(\omega(x - x'))$$

GPs can model periodic structure

Sums of sinusoidal basis functions connects GPs to Fourier series and Fourier transforms

Big fat covariance function quiz

Which are GPs? Compute the GPs mean and covariance functions.

3. Random sinusoid model

$$h(x) = a\cos(\omega t) + b\sin(\omega t)$$

$$a \sim \mathcal{N}(0, \sigma^2)$$
$$b \sim \mathcal{N}(0, \sigma^2)$$

Gaussians are closed under linear transformations: so are GPs

$$m(x) = 0$$
$$\Sigma(x, x') = \sigma^2 \cos(\omega(x - x'))$$

GPs can model periodic structure

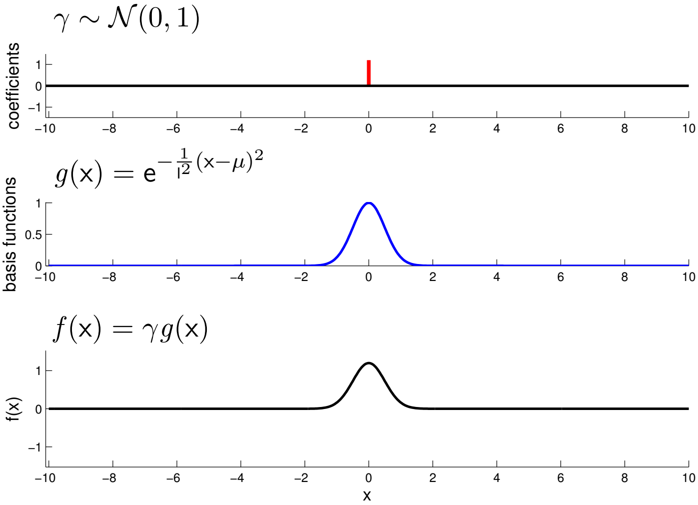
Sums of sinusoidal basis functions connects GPs to Fourier series and Fourier transforms

Bochner's theorem: Any stationary covariance function can be written as:

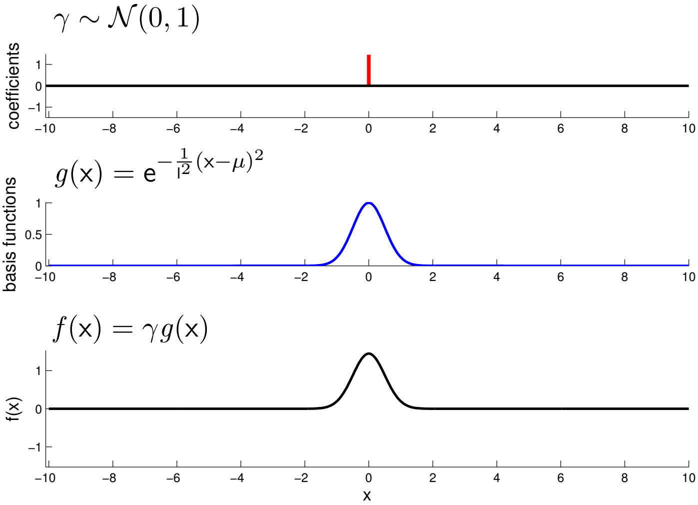
$$\Sigma(x - x') = \int \sigma^2(\omega) \cos(\omega(x - x'))\mathrm{d}\omega$$

roughly, the function comprises "an uncountably infinite sum of random sins and cosines"

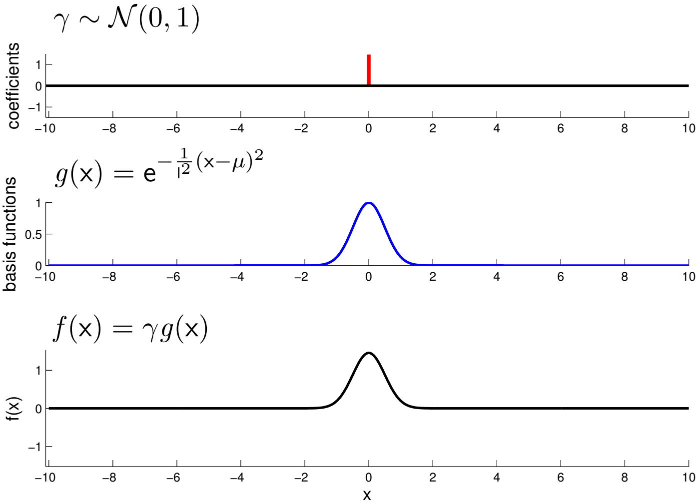
Basis function view of Gaussian processes



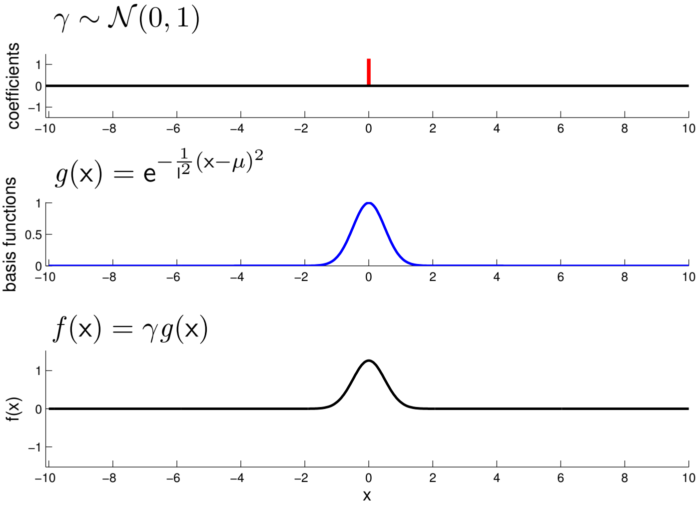
Basis function view of Gaussian processes



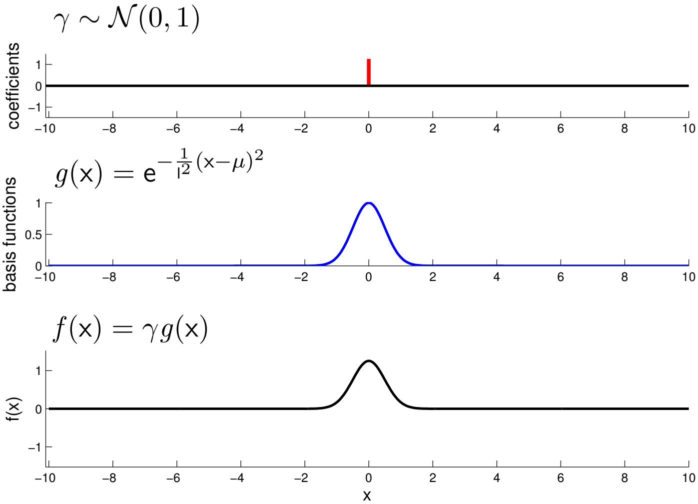
Basis function view of Gaussian processes



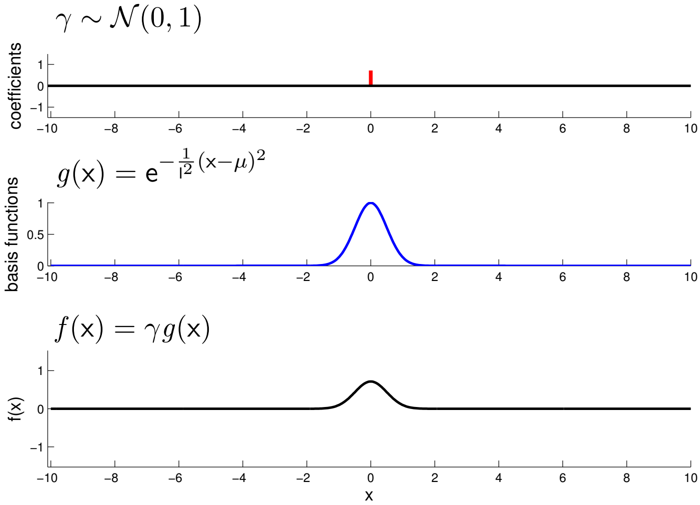
Basis function view of Gaussian processes



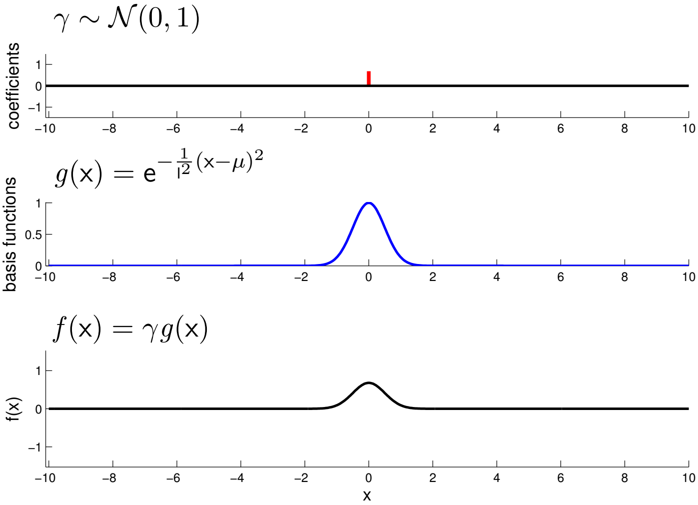
Basis function view of Gaussian processes



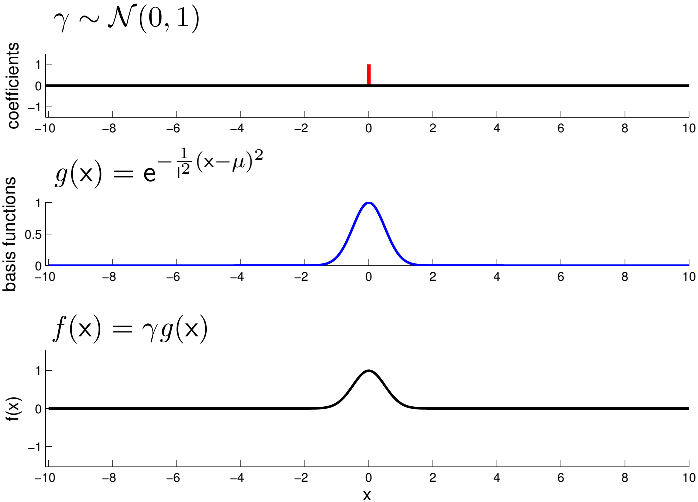
Basis function view of Gaussian processes



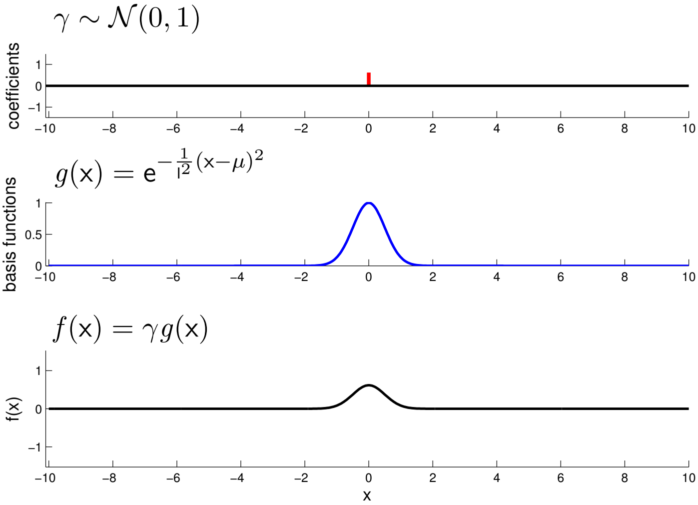
Basis function view of Gaussian processes



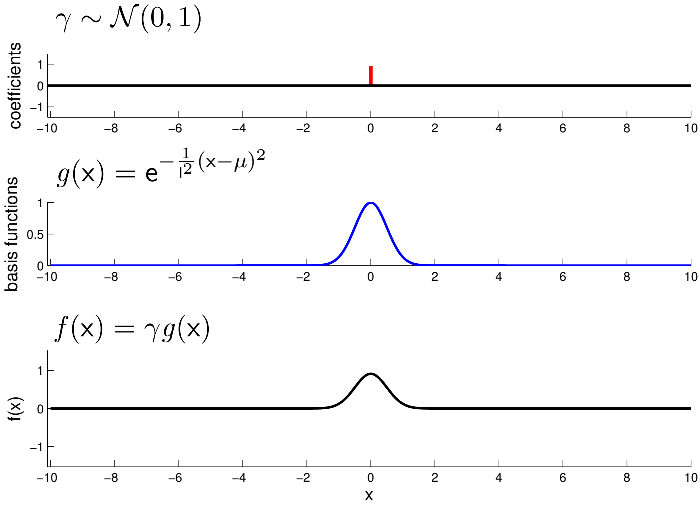
Basis function view of Gaussian processes



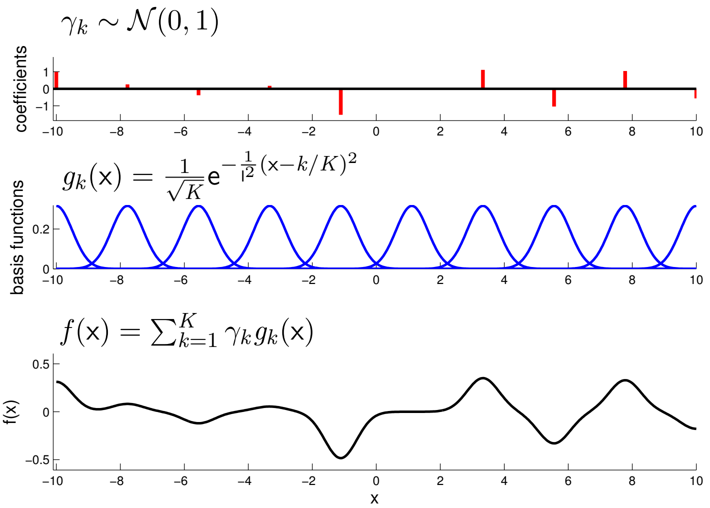
Basis function view of Gaussian processes



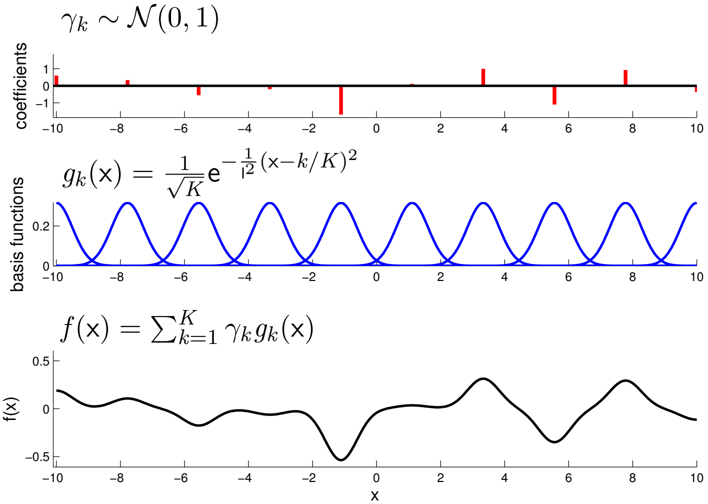
Basis function view of Gaussian processes



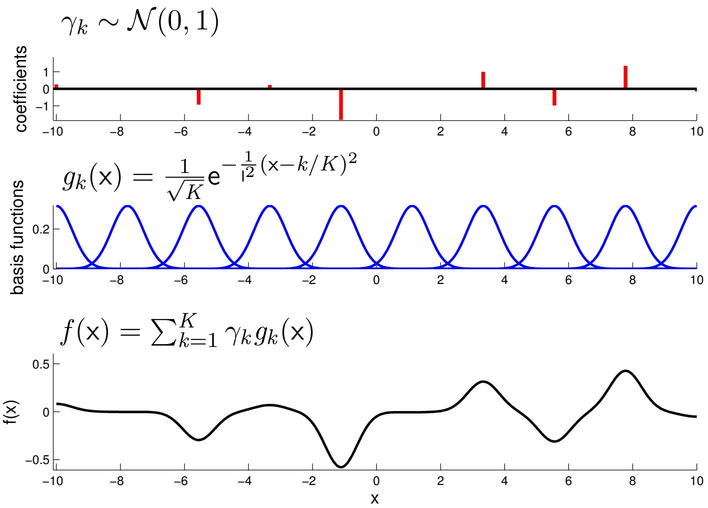
Basis function view of Gaussian processes



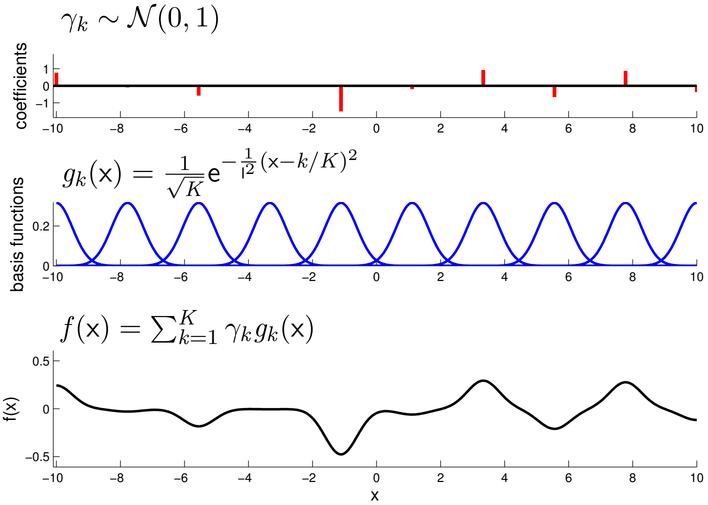
Basis function view of Gaussian processes



Basis function view of Gaussian processes

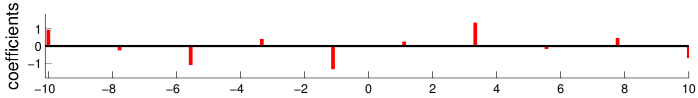


Basis function view of Gaussian processes

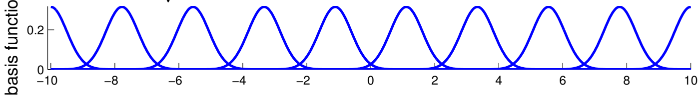


Basis function view of Gaussian processes

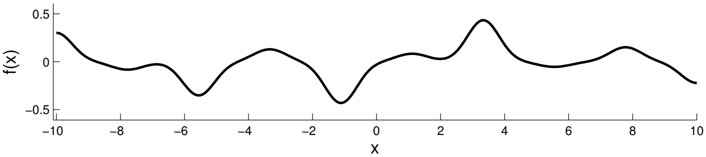
$$\gamma_k \sim \mathcal{N}(0,1)$$



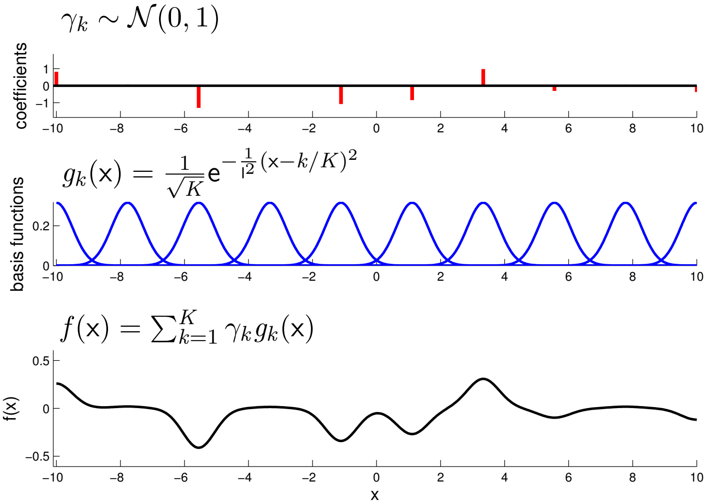
$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$



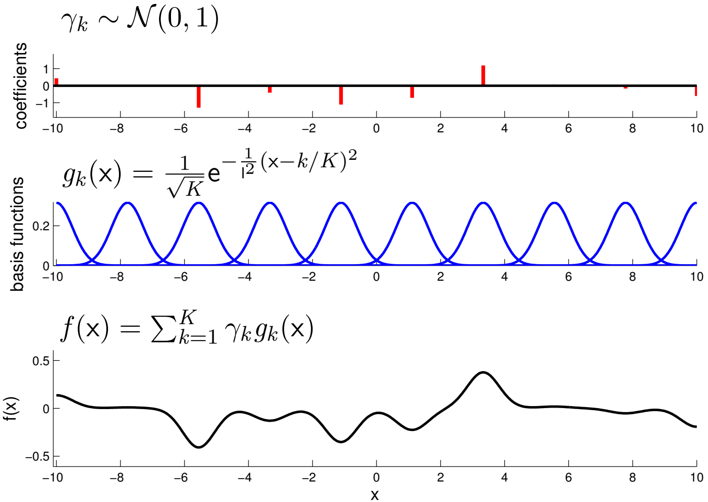
$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$



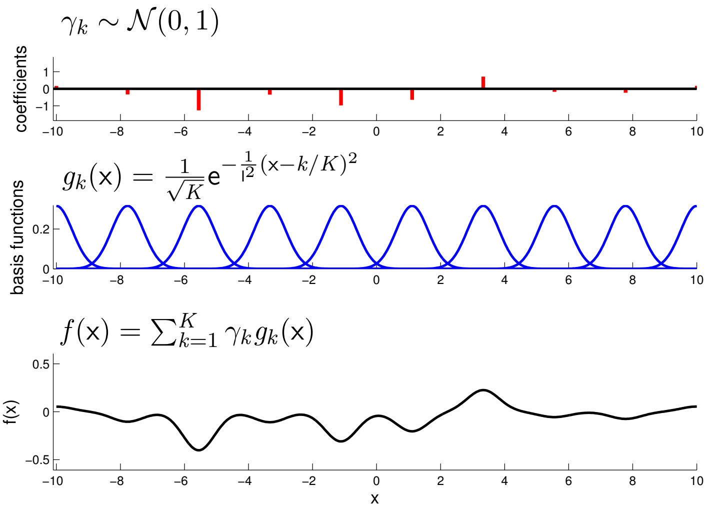
Basis function view of Gaussian processes



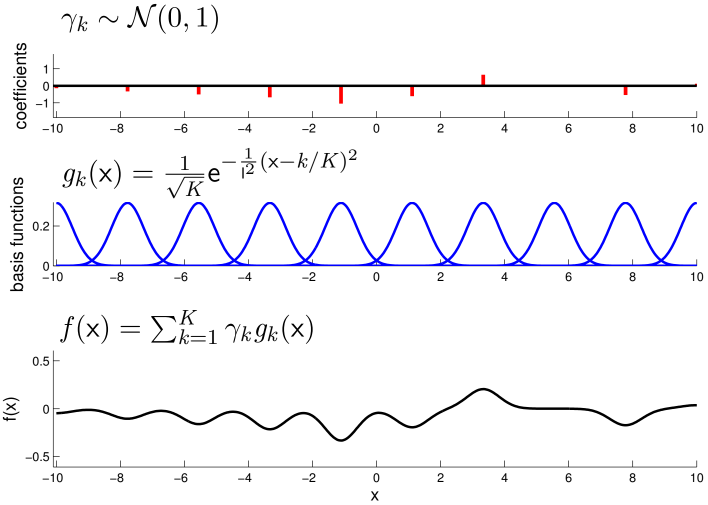
Basis function view of Gaussian processes



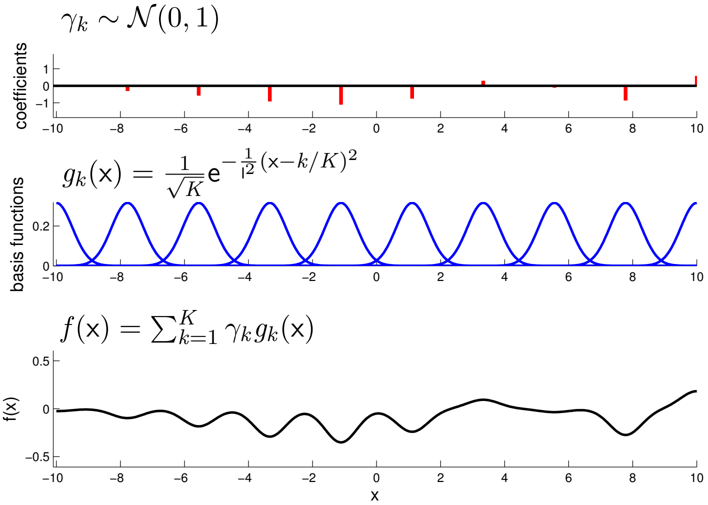
Basis function view of Gaussian processes



Basis function view of Gaussian processes



Basis function view of Gaussian processes

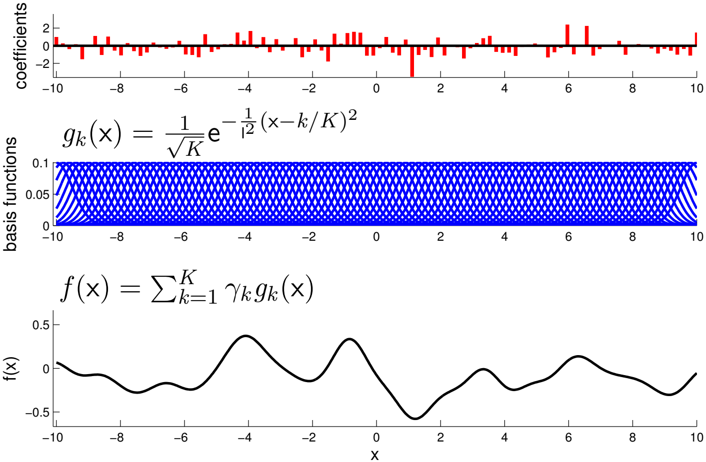


Basis function view of Gaussian processes

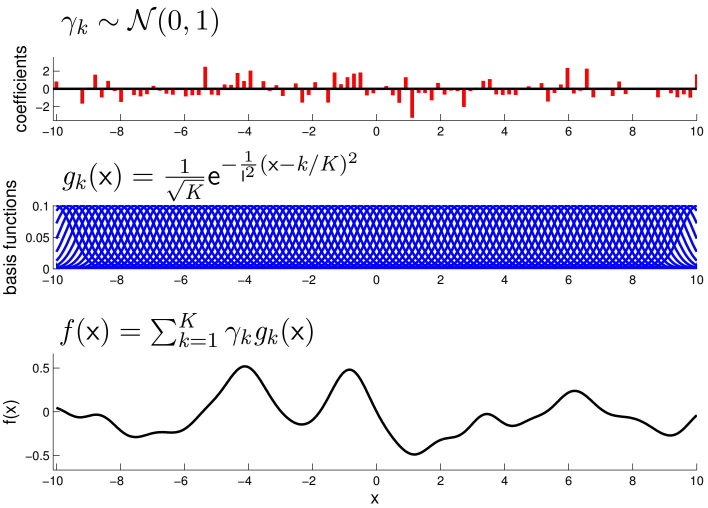
$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

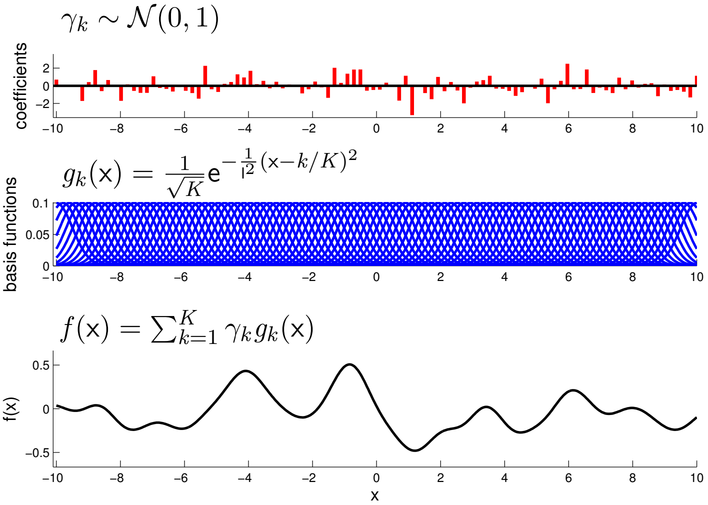
$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$



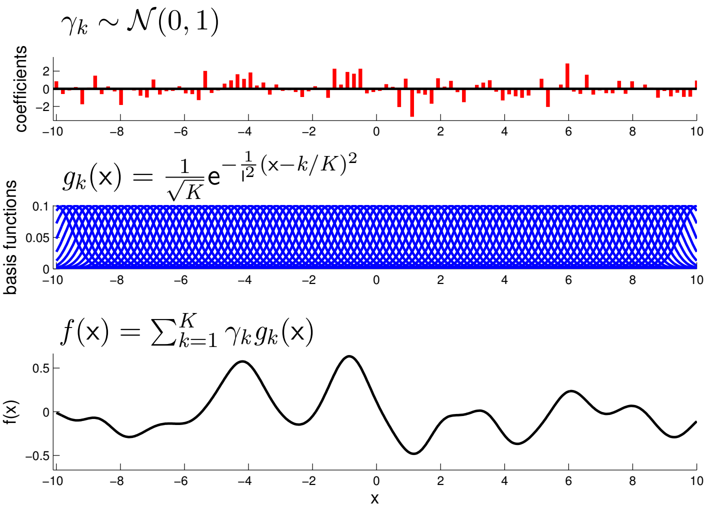
Basis function view of Gaussian processes



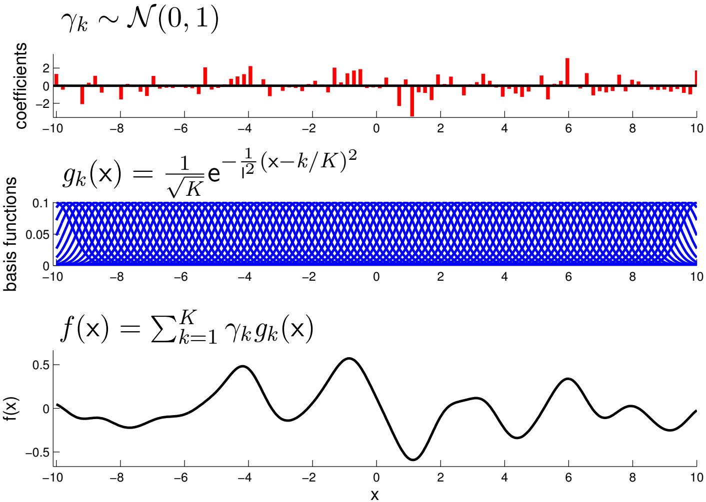
Basis function view of Gaussian processes



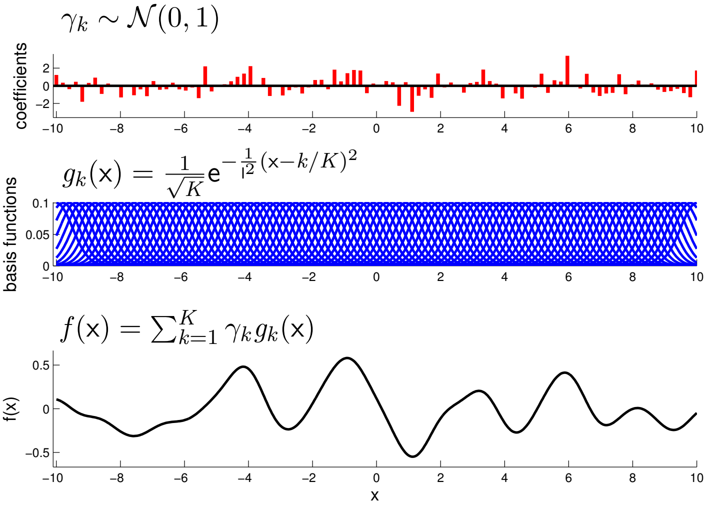
Basis function view of Gaussian processes



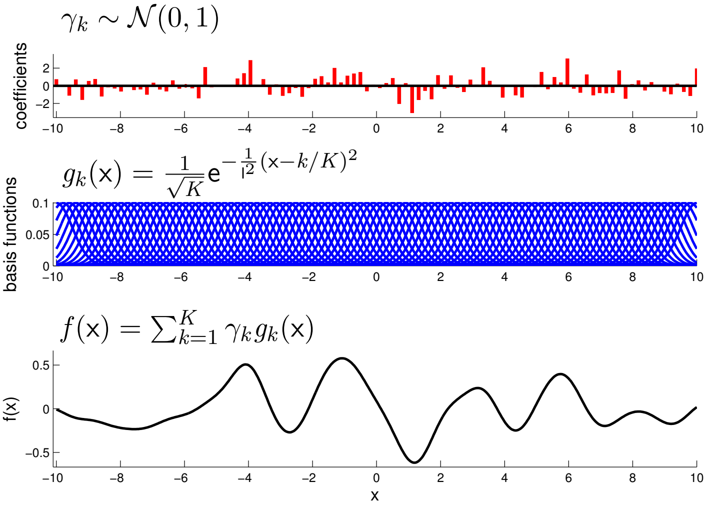
Basis function view of Gaussian processes



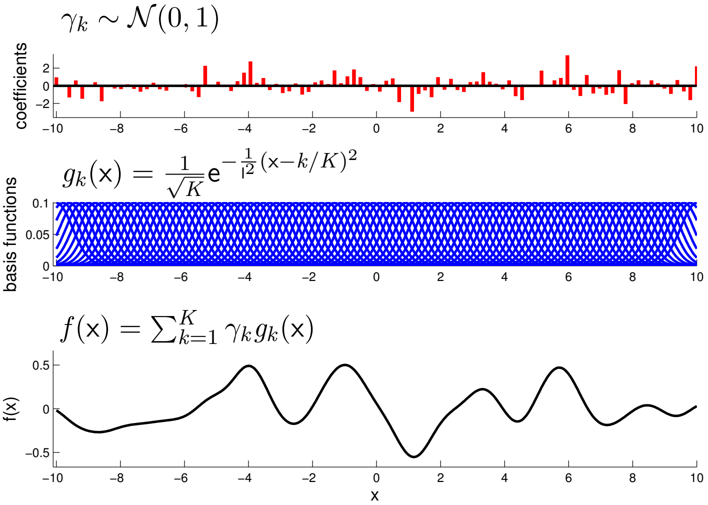
Basis function view of Gaussian processes



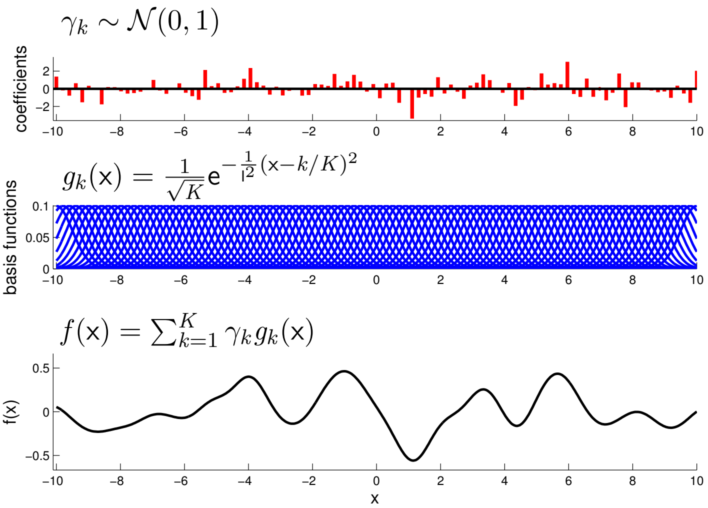
Basis function view of Gaussian processes



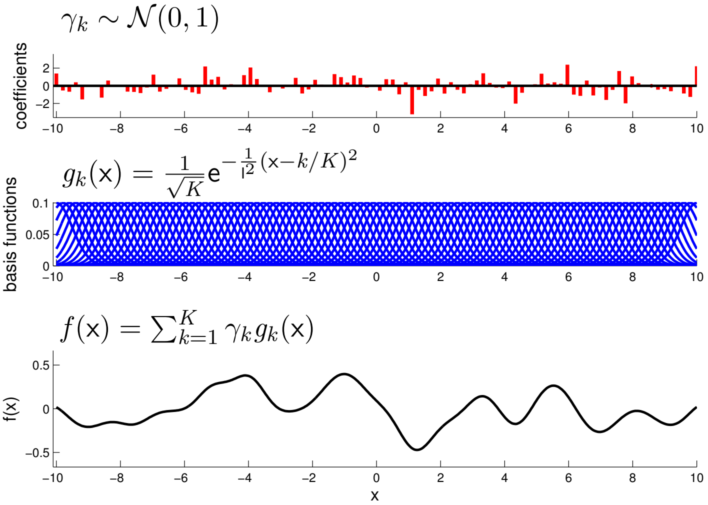
Basis function view of Gaussian processes



Basis function view of Gaussian processes



Basis function view of Gaussian processes



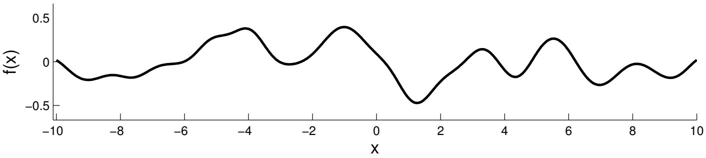
Basis function view of Gaussian processes

$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = \langle f(\mathsf{x}) \rangle$$



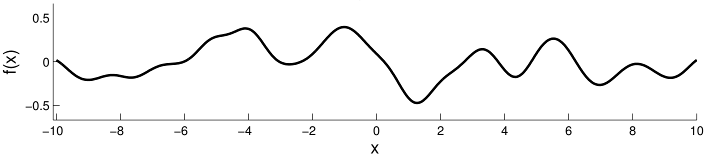
Basis function view of Gaussian processes

$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = \langle \textstyle\sum_k \gamma_k g_k(\mathsf{x}) \rangle$$



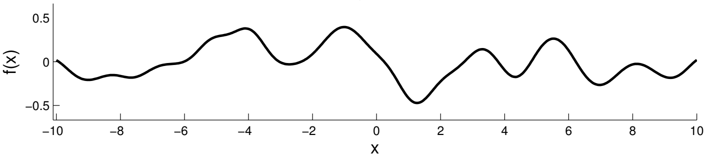
Basis function view of Gaussian processes

$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = \sum_k \langle \gamma_k \rangle g_k(\mathsf{x})$$



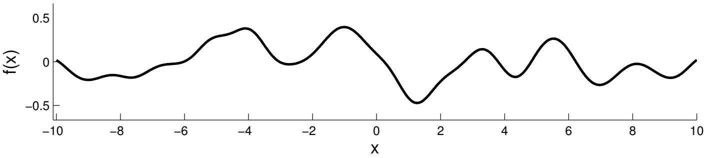
Basis function view of Gaussian processes

$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = 0$$



Basis function view of Gaussian processes

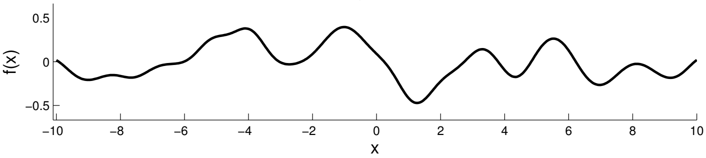
$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = 0$$

$$\mathsf{K}(\mathsf{x},\mathsf{x}') = \langle f(\mathsf{x}) f(\mathsf{x}') \rangle$$



Basis function view of Gaussian processes

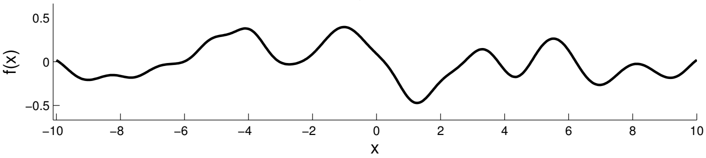
$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = 0$$

$$\mathsf{K}(\mathsf{x},\mathsf{x}') = \langle \sum_k \gamma_k g_k(\mathsf{x}) \sum_{k'} \gamma_{k'} g_{k'}(\mathsf{x}') \rangle$$



Basis function view of Gaussian processes

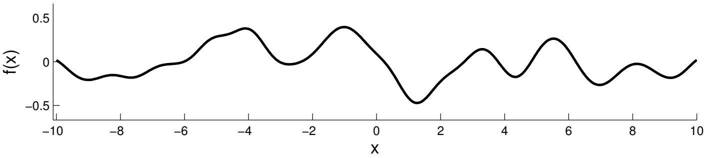
$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = 0$$

$$\mathsf{K}(\mathsf{x},\mathsf{x}') = \sum_{k,k'} \langle \gamma_k \gamma_{k'} \rangle g_k(\mathsf{x}) g_{k'}(\mathsf{x}')$$



Basis function view of Gaussian processes

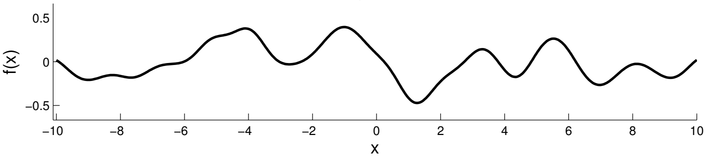
$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = 0$$

$$\mathsf{K}(\mathsf{x},\mathsf{x}') = \sum_{k,k'} \delta_{k,k'} g_k(\mathsf{x}) g_{k'}(\mathsf{x}')$$



Basis function view of Gaussian processes

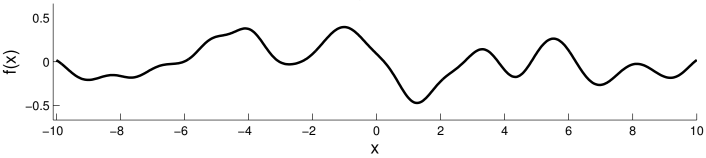
$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = 0$$

$$\mathsf{K}(\mathsf{x},\mathsf{x}') = \sum_k g_k(\mathsf{x}) g_k(\mathsf{x}')$$



Basis function view of Gaussian processes

$$\gamma_k \sim \mathcal{N}(0,1)$$

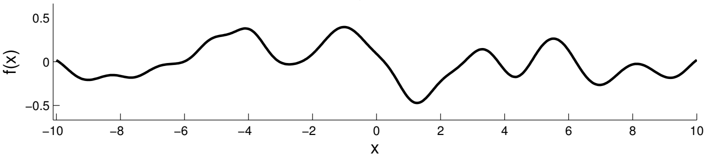
$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = 0$$

$$\mathsf{K}(\mathsf{x},\mathsf{x}') = \sum_k g_k(\mathsf{x}) g_k(\mathsf{x}')$$

$$= \frac{1}{K} \sum_k \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2 - \frac{1}{\mathsf{l}^2}(\mathsf{x}'-k/K)^2}$$



Basis function view of Gaussian processes

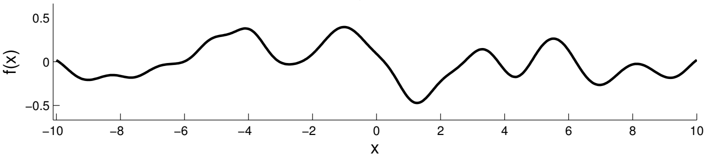
$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

$$m(\mathsf{x}) = 0$$

$$\begin{aligned} \mathsf{K}(\mathsf{x},\mathsf{x}') &= \textstyle\sum_k g_k(\mathsf{x}) g_k(\mathsf{x}') \\ &= \frac{1}{K} \textstyle\sum_k \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2 - \frac{1}{\mathsf{l}^2}(\mathsf{x}'-k/K)^2} \\ &\underset{K\to\infty}{\longrightarrow} \int du \;\; \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-u)^2 - \frac{1}{\mathsf{l}^2}(\mathsf{x}'-u)^2} \end{aligned}$$



Basis function view of Gaussian processes

$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

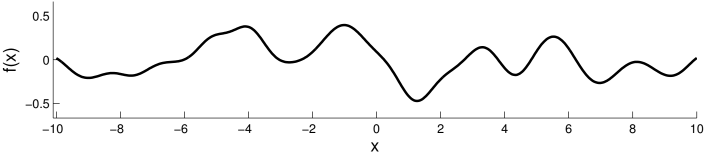
$$m(\mathsf{x}) = 0$$

$$\mathsf{K}(\mathsf{x},\mathsf{x}') = \sum_k g_k(\mathsf{x}) g_k(\mathsf{x}')$$

$$= \frac{1}{K} \sum_k \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2 - \frac{1}{\mathsf{l}^2}(\mathsf{x}'-k/K)^2}$$

$$\xrightarrow[K \to \infty]{} \int du \;\; \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-u)^2 - \frac{1}{\mathsf{l}^2}(\mathsf{x}'-u)^2}$$

$$\propto \mathsf{e}^{-\frac{1}{2\mathsf{l}^2}(\mathsf{x}-\mathsf{x}')^2}$$



Basis function view of Gaussian processes

$$\gamma_k \sim \mathcal{N}(0,1)$$

$$g_k(\mathsf{x}) = \frac{1}{\sqrt{K}} \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2}$$

$$f(\mathsf{x}) = \sum_{k=1}^{K} \gamma_k g_k(\mathsf{x})$$

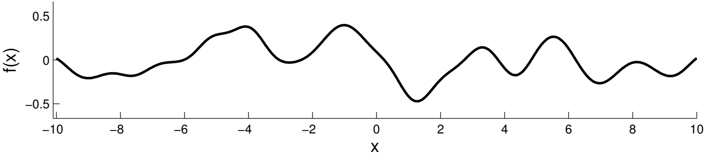
$$m(\mathsf{x}) = 0$$

$$\mathsf{K}(\mathsf{x},\mathsf{x}') = \sum_k g_k(\mathsf{x}) g_k(\mathsf{x}')$$

$$= \frac{1}{K}\sum_k \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-k/K)^2 - \frac{1}{\mathsf{l}^2}(\mathsf{x}'-k/K)^2}$$

$$\underset{K\to\infty}{\longrightarrow} \int du \;\; \mathsf{e}^{-\frac{1}{\mathsf{l}^2}(\mathsf{x}-u)^2 - \frac{1}{\mathsf{l}^2}(\mathsf{x}'-u)^2}$$

$$\propto \mathsf{e}^{-\frac{1}{2\mathsf{l}^2}(\mathsf{x}-\mathsf{x}')^2}$$



Gaussian processes $\equiv$ models with ∞ parameters

A selection of GP models

probabilistic model	linear mappings	neural network mappings	Gaussian Process mappings
	$f(x) = Wx$	$f(x) = \mathrm{NN}(x; W)$	$f(x) \sim \mathcal{GP}$

A selection of GP models

probabilistic model	linear mappings	neural network mappings	Gaussian Process mappings
	$f(x) = Wx$	$f(x) = \mathrm{NN}(x; W)$	$f(x) \sim \mathcal{GP}$

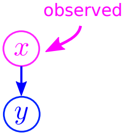
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y \,|\, x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

all edges imply a function



A selection of GP models

probabilistic model	linear mappings	neural network mappings	Gaussian Process mappings
	$f(x) = Wx$	$f(x) = \mathrm{NN}(x; W)$	$f(x) \sim \mathcal{GP}$
	linear regression		

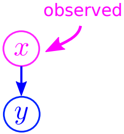
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y \,|\, x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

all edges imply a function



A selection of GP models

probabilistic model		linear mappings	neural network mappings	Gaussian Process mappings
		$f(x) = Wx$	$f(x) = \mathrm{NN}(x; W)$	$f(x) \sim \mathcal{GP}$

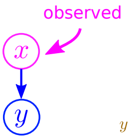
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y \mid x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

all edges imply a function



linear regression

(logistic regression)

$y|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$

A selection of GP models

probabilistic model

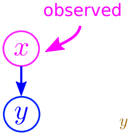
roots

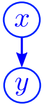
$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y \,|\, x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

all edges imply a function





linear mappings

$f(x) = Wx$

linear regression

(logistic regression)

$y|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$

neural network mappings

$f(x) = \mathrm{NN}(x; W)$

Gaussian Process mappings

$f(x) \sim \mathcal{GP}$

A selection of GP models

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$

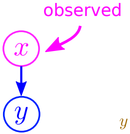
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y \mid x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

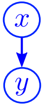
all edges imply a function



linear regression

(logistic regression)

$y|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$



PCA or factor analysis

A selection of GP models

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$

roots

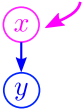
$$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$$

other nodes

$$y \,|\, x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$$

all edges imply a function

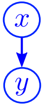
observed



linear regression

(logistic regression)

$y|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$



PCA or factor analysis

A selection of GP models

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$

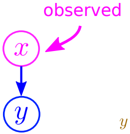
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y \,|\, x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

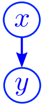
all edges imply a function



linear regression

(logistic regression)

$y|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$



PCA or factor analysis

Gaussian auto-regressive (or Markov) model

A selection of GP models

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$

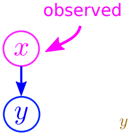
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

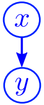
$y|x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

all edges imply a function



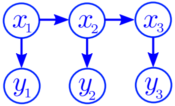
linear regression

(logistic regression)
$y|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$



PCA or factor analysis

Gaussian auto-regressive (or Markov) model



A selection of GP models

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$

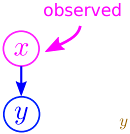
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y|x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

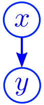
all edges imply a function



linear regression

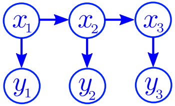
(logistic regression)

$y|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$



PCA or factor analysis

Gaussian auto-regressive (or Markov) model



linear Gaussian state space model (LGSSM)

A selection of GP models

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$

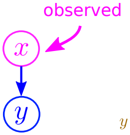
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y|x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

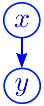
all edges imply a function



linear regression

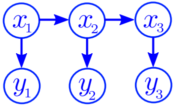
(logistic regression)

$y|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$



PCA or factor analysis

Gaussian auto-regressive (or Markov) model



linear Gaussian state space model (LGSSM)

A selection of GP models

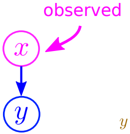
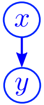
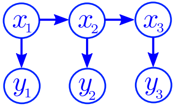
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y|x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

all edges imply a function

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$
	linear regression (logistic regression) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$	neural network regression (NN classifcation) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$	
	PCA or factor analysis	variational auto-encoder (VAE) (deep generative model, DGM)	
	Gaussian auto-regressive (or Markov) model	neural auto-regressive density estimation (NADE)	
	linear Gaussian state space model (LGSSM)	recurrent neural latent variable model	

A selection of GP models

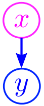
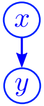
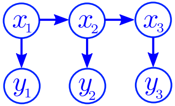
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y \,|\, x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

all edges imply a function

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$
	linear regression (logistic regression) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$	neural network regression (NN classifcation) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$	bad term for a model
	PCA or factor analysis	variational auto-encoder (VAE) (deep generative model, DGM)	
	Gaussian auto-regressive (or Markov) model	neural auto-regressive density estimation (NADE)	
	linear Gaussian state space model (LGSSM)	recurrent neural latent variable model	

A selection of GP models

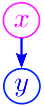
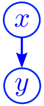
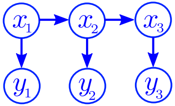
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y \,|\, x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

all edges imply a function

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$
observed 	linear regression (logistic regression) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$	neural network regression (NN classifcation) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$	
	PCA or factor analysis	variational auto-encoder (VAE) (deep generative model, DGM) bad term for a model	
	Gaussian auto-regressive (or Markov) model	neural auto-regressive density estimation (NADE)	
	linear Gaussian state space model (LGSSM)	recurrent neural latent variable model	

A selection of GP models

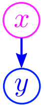
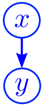
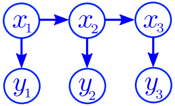
roots

$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y|x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

all edges imply a function

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$
observed 	linear regression (logistic regression) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$	neural network regression (NN classifcation) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$	Gaussian Process regression (GP classifcation) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$
	PCA or factor analysis	variational auto-encoder (VAE) (deep generative model, DGM) — bad term for a model	Gaussian Process latent variable model
	Gaussian auto-regressive (or Markov) model	neural auto-regressive density estimation (NADE)	Gaussian process auto-regressive model (GPAR)
	linear Gaussian state space model (LGSSM)	recurrent neural latent variable model	Gaussian process state-space model (GP-SSM)

A selection of GP models

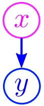
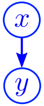
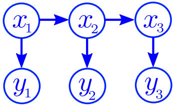
roots

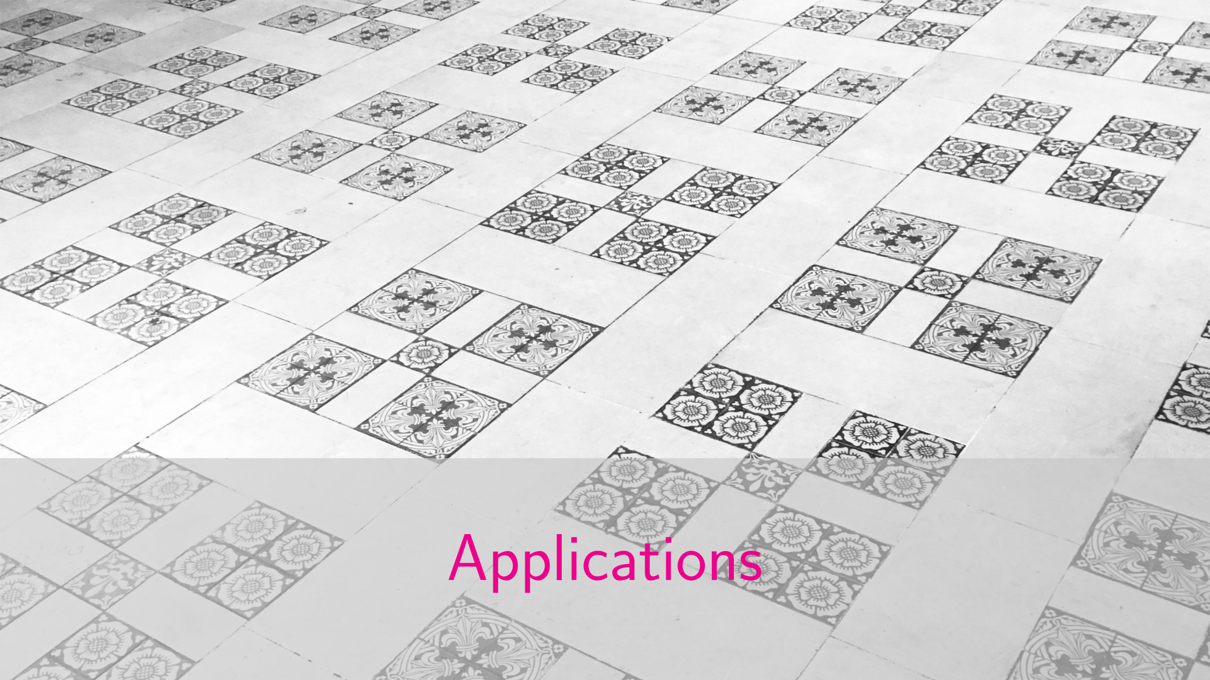
$x \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$

other nodes

$y|x \sim \mathcal{N}(f(x), \sigma_y^2 \mathrm{I})$

all edges imply a function

probabilistic model	linear mappings $f(x) = Wx$	neural network mappings $f(x) = \mathrm{NN}(x; W)$	Gaussian Process mappings $f(x) \sim \mathcal{GP}$
	linear regression (logistic regression) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$	neural network regression (NN classifcation) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$	Gaussian Process regression (GP classifcation) $y\|x = \mathrm{Bern}(\mathrm{softmax}[f(x)])$
	PCA or factor analysis	variational auto-encoder (VAE) (deep generative model, DGM) bad term for a model	Gaussian Process latent variable model used inference networks in 2005
	Gaussian auto-regressive (or Markov) model	neural auto-regressive density estimation (NADE)	Gaussian process auto-regressive model (GPAR)
	linear Gaussian state space model (LGSSM)	recurrent neural latent variable model	Gaussian process state-space model (GP-SSM)



Applications

What are Gaussian Processes good for?

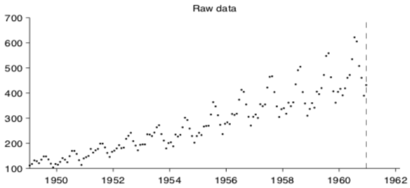
Strengths

- **interpretable** machine learning (covariance functions specify easy-to-explain high-level properties of functions)
- **data-efficient** machine learning (non-parametric + Bayesian $\implies$ lots of flexibility + avoid overfitting)
- **decision making** (well-calibrated uncertainties: knows when it does not know)
- **automated machine learning** including **probabilistic numerics** (regression and classification are rock-solid)

Weaknesses

- **Large numbers of datapoints** ($N \leq 10^5$ unless there is special structure, due to covariance matrix inversion & storage)
- **High-dimensional inputs spaces** ($D \leq 10^2$ unless there is special structure, due to need to compute pair-wise elements of covariance function)

Interpretable auto-ML: the automatic statistician



Interpretable auto-ML: the automatic statistician

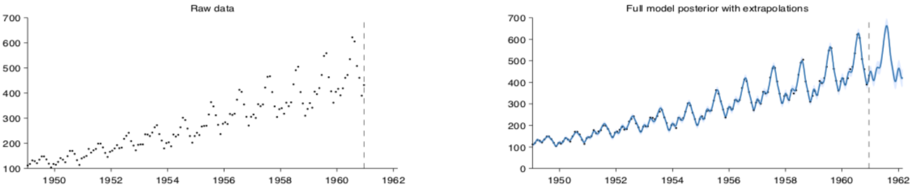


Figure 1: Raw data (left) and model posterior with extrapolation (right)

The structure search algorithm has identified four additive components in the data.

- A linearly increasing function.
- An approximately periodic function with a period of 1.0 years and with approximately linearly increasing amplitude.
- A smooth function.
- Uncorrelated noise with linearly increasing standard deviation.

Interpretable auto-ML: the automatic statistician

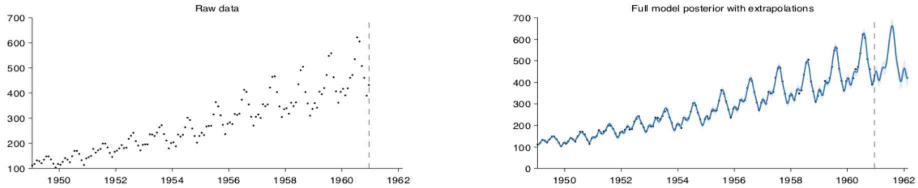


Figure 1: Raw data (left) and model posterior with extrapolation (right)

$$\Sigma(t,t') = \Sigma_1(t,t') + \Sigma_2(t,t') + \Sigma_3(t,t') + \Sigma_4(t,t') + \Sigma_5(t,t')$$

The structure search algorithm has identified four additive components in the data.

- A linearly increasing function.
- An approximately periodic function with a period of 1.0 years and with approximately linearly increasing amplitude.
- A smooth function.
- Uncorrelated noise with linearly increasing standard deviation.

Interpretable auto-ML: the automatic statistician

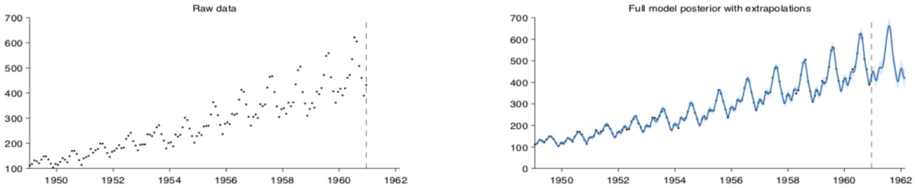


Figure 1: Raw data (left) and model posterior with extrapolation (right)

$$\Sigma(t,t') = \Sigma_1(t,t') + \Sigma_2(t,t') + \Sigma_3(t,t') + \Sigma_4(t,t') + \Sigma_5(t,t')$$

The structure search algorithm has identified four additive components in the data.

- A linearly increasing function. $\Sigma_1(t,t') = \sigma_m^2 t\, t' + \sigma_c^2$
- An approximately periodic function with a period of 1.0 years and with approximately linearly increasing amplitude.
- A smooth function.
- Uncorrelated noise with linearly increasing standard deviation.

Interpretable auto-ML: the automatic statistician

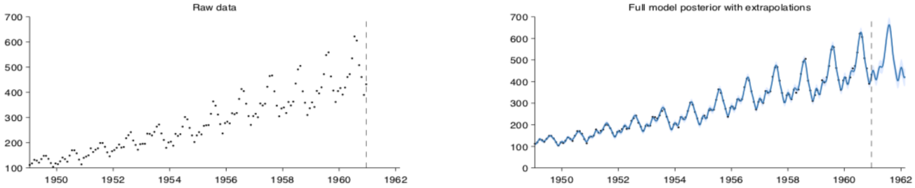


Figure 1: Raw data (left) and model posterior with extrapolation (right)

$$\Sigma(t,t') = \Sigma_1(t,t') + \Sigma_2(t,t') + \Sigma_3(t,t') + \Sigma_4(t,t') + \Sigma_5(t,t')$$

The structure search algorithm has identified four additive components in the data.

- A linearly increasing function.

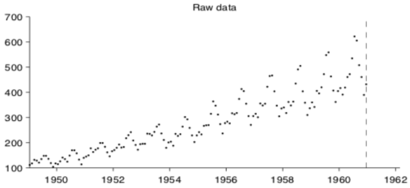
$$\Sigma_1(t,t') = \sigma_m^2 t\, t' + \sigma_c^2$$

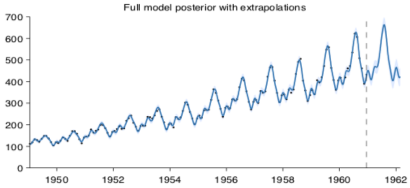
- An approximately periodic function with a period of 1.0 years and with approximately linearly increasing amplitude.

$$\Sigma_2(t,t') = \mathrm{SE}(t,t') \exp(k \cos(\omega(t-t')))$$

- A smooth function.
- Uncorrelated noise with linearly increasing standard deviation.

Interpretable auto-ML: the automatic statistician





$\Sigma_3(t,t') = \mathrm{SE}(t,t')$

Raw data (left) and model posterior with extrapolation (right)

$\Sigma(t,t') = \Sigma_1(t,t') + \Sigma_2(t,t') + \Sigma_3(t,t') + \Sigma_4(t,t') + \Sigma_5(t,t')$

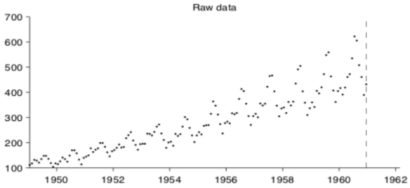
The structure search algorithm has identified four additive components in the data.

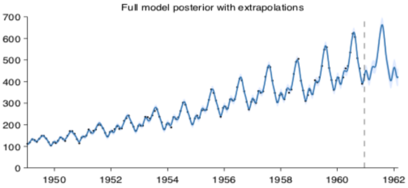
- A linearly increasing function.
- An approximately periodic function with a period of 1.0 years and with approximately linearly increasing amplitude.
- A smooth function.
- Uncorrelated noise with linearly increasing standard deviation.

$\Sigma_1(t,t') = \sigma_m^2 t\, t' + \sigma_c^2$

$\Sigma_2(t,t') = \mathrm{SE}(t,t')\exp(k\cos(\omega(t-t')))$

Interpretable auto-ML: the automatic statistician





$\Sigma_3(t, t') = \text{SE}(t, t')$

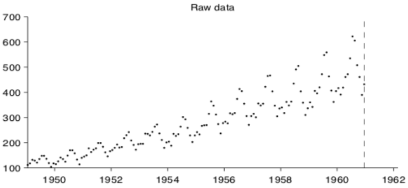
Raw data (left) and model posterior with extrapolation (right)

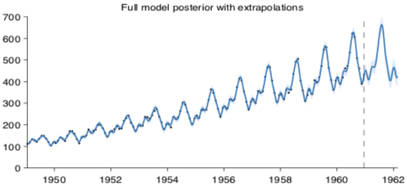
$\Sigma(t, t') = \Sigma_1(t, t') + \Sigma_2(t, t') + \Sigma_3(t, t') + \Sigma_4(t, t') + \Sigma_5(t, t')$

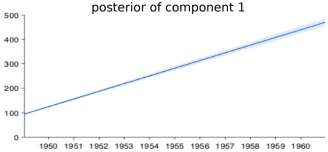
The structure search algorithm has identified four additive components in the data.

- A linearly increasing function. ← $\Sigma_1(t, t') = \sigma_m^2 t\, t' + \sigma_c^2$
- An approximately periodic function with a period of 1.0 years and with approximately linearly increasing amplitude. ← $\Sigma_2(t, t') = \text{SE}(t, t') \exp(k \cos(\omega(t - t')))$
- A smooth function. ← $\Sigma_3(t, t')$
- Uncorrelated noise with linearly increasing standard deviation. ← $\Sigma_4(t, t') = \sigma_y^2 t \delta(t - t')$

Interpretable auto-ML: the automatic statistician



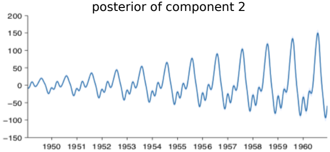


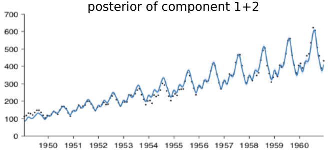


$$\Sigma_1(t,t') = \sigma_m^2 t\, t' + \sigma_c^2$$

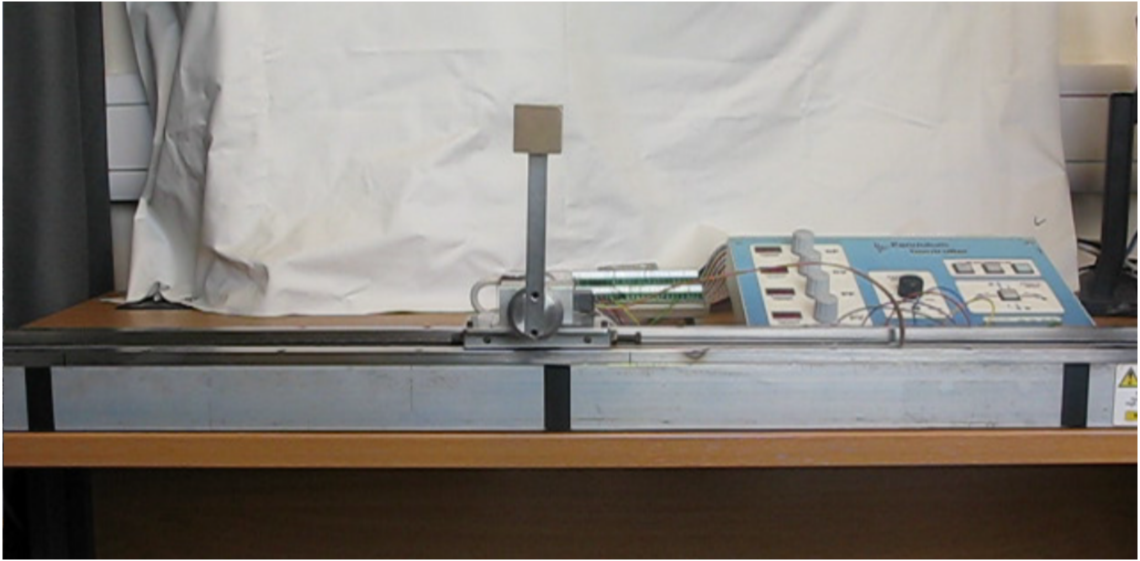
$$\Sigma_2(t,t') = \mathrm{SE}(t,t') \exp(k \cos(\omega(t-t')))$$

$$\Sigma(t,t') = \Sigma_1(t,t') + \Sigma_2(t,t')$$

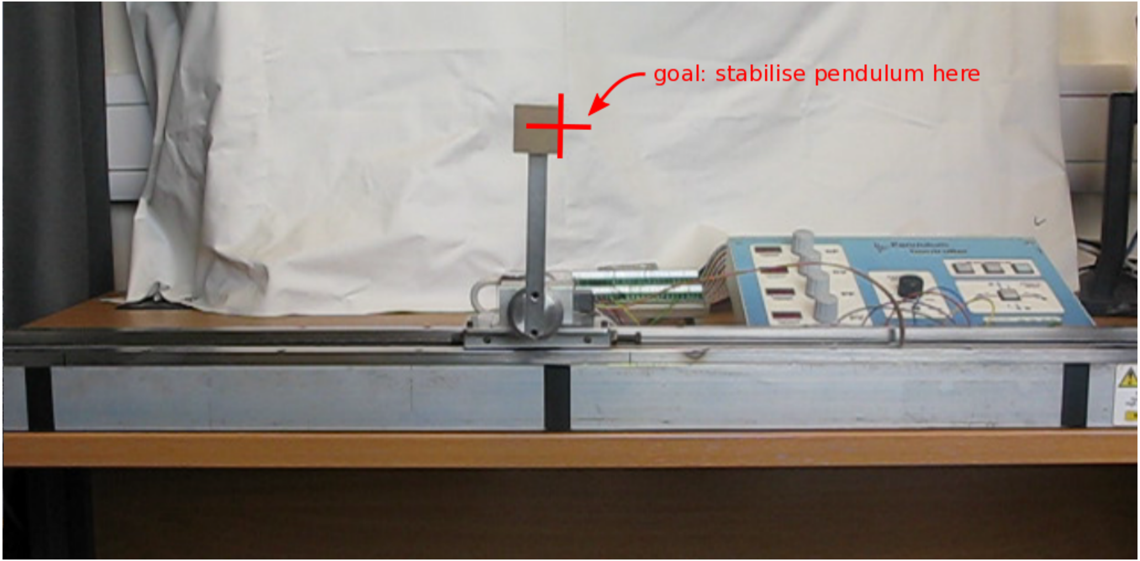




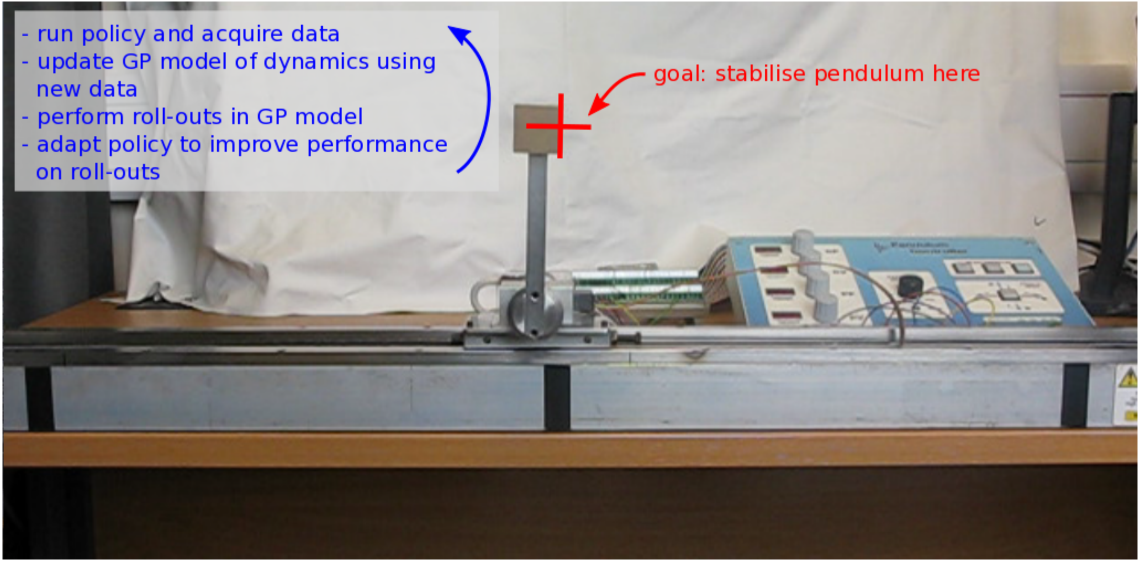
Data-efficient reinforcement learning: PILCO



Data-efficient reinforcement learning: PILCO



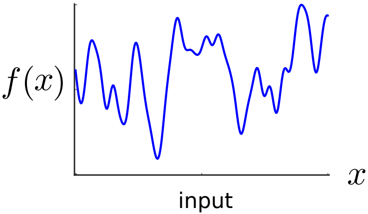
Data-efficient reinforcement learning: PILCO



Deep Gaussian Processes

$$y(x) = f(x) + \sigma_{\mathrm{y}}\epsilon$$

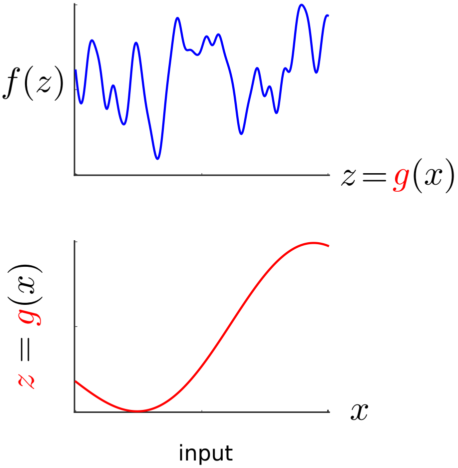
$$f(x) = \mathcal{GP}(0, K_f(x, x'))$$



Deep Gaussian Processes

$$y(x) = f(g(x)) + \sigma_{\mathrm{y}}\epsilon$$

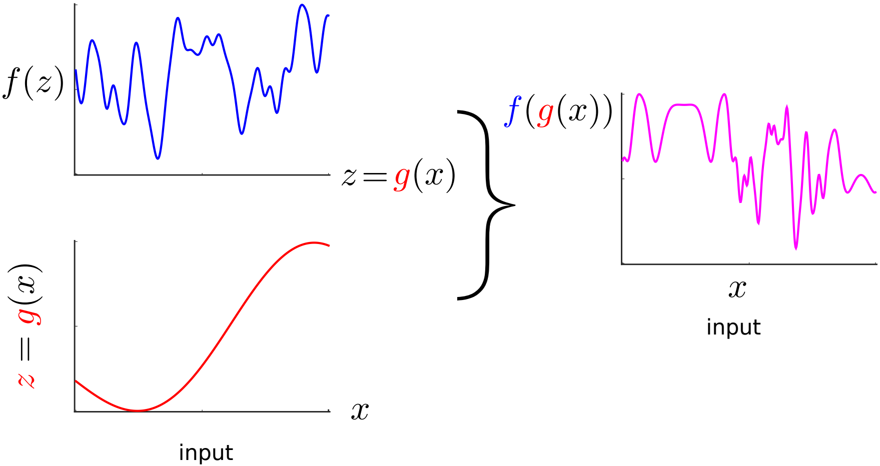
$$f(x) = \mathcal{GP}(0, K_f(x, x'))$$
$$g(x) = \mathcal{GP}(0, K_g(x, x'))$$



Deep Gaussian Processes

$$y(x) = f(g(x)) + \sigma_{\mathrm{y}}\epsilon$$

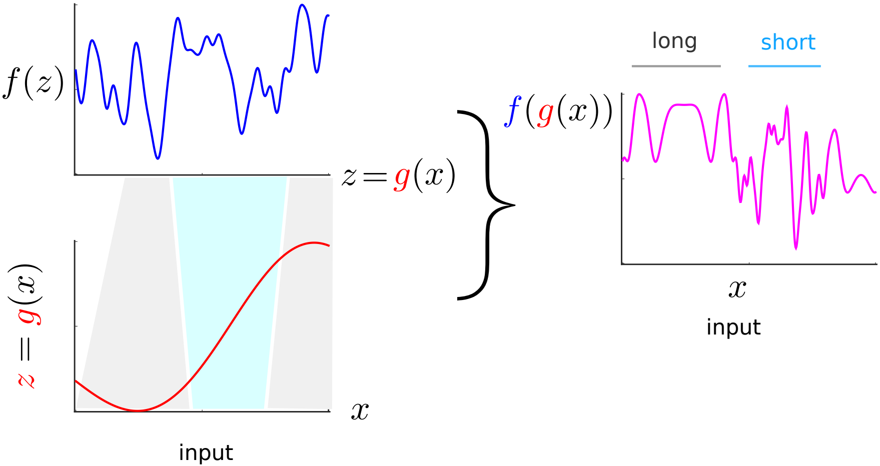
$$f(x) = \mathcal{GP}(0, K_f(x, x'))$$
$$g(x) = \mathcal{GP}(0, K_g(x, x'))$$



Deep Gaussian Processes

$$y(x) = f(g(x)) + \sigma_{\mathrm{y}}\epsilon$$

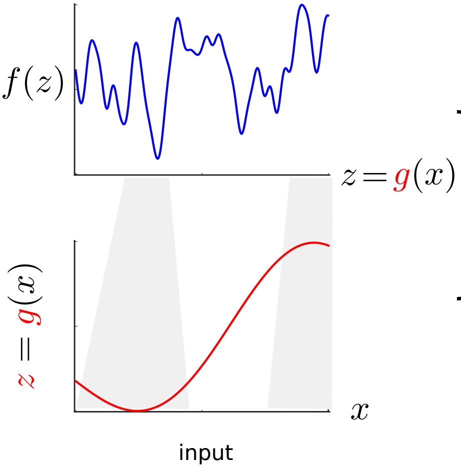
$$f(x) = \mathcal{GP}(0, K_f(x, x'))$$
$$g(x) = \mathcal{GP}(0, K_g(x, x'))$$

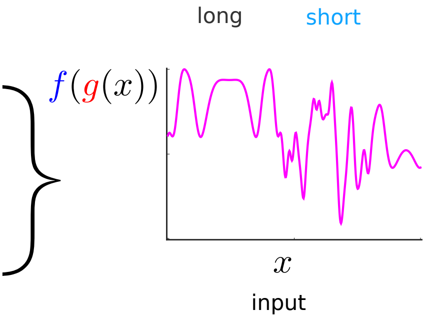


Deep Gaussian Processes

$$y(x) = f(g(x)) + \sigma_y \epsilon$$

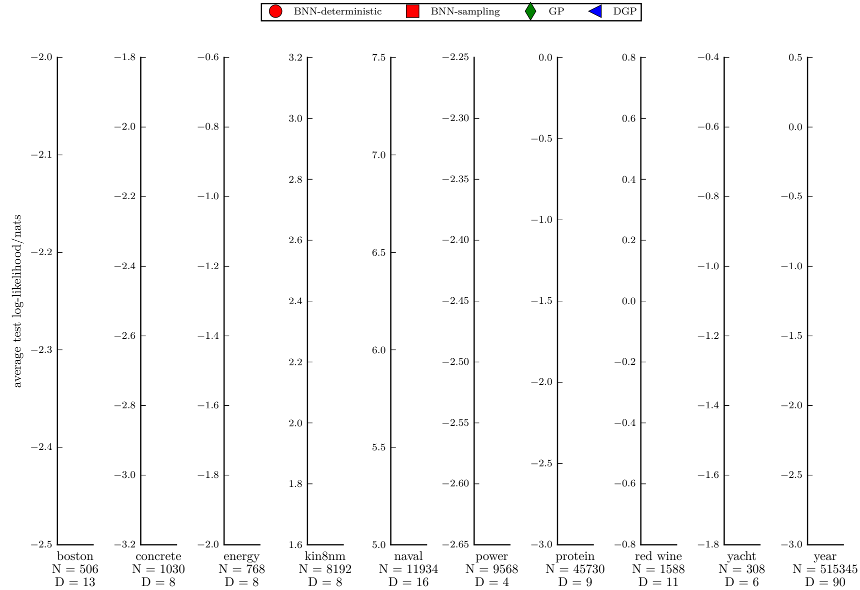
$$f(x) = \mathcal{GP}(0, K_f(x, x'))$$
$$g(x) = \mathcal{GP}(0, K_g(x, x'))$$



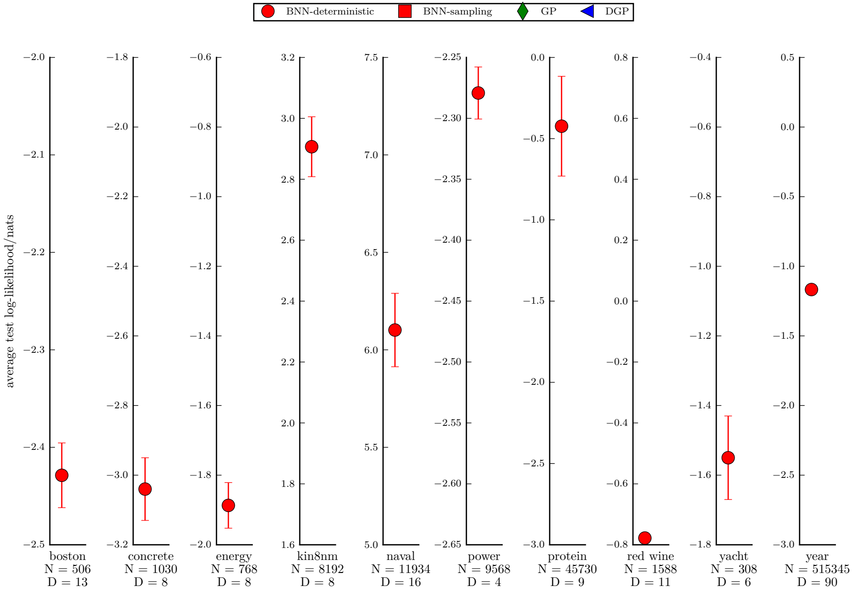


Deep GP perform
automatic kernel design

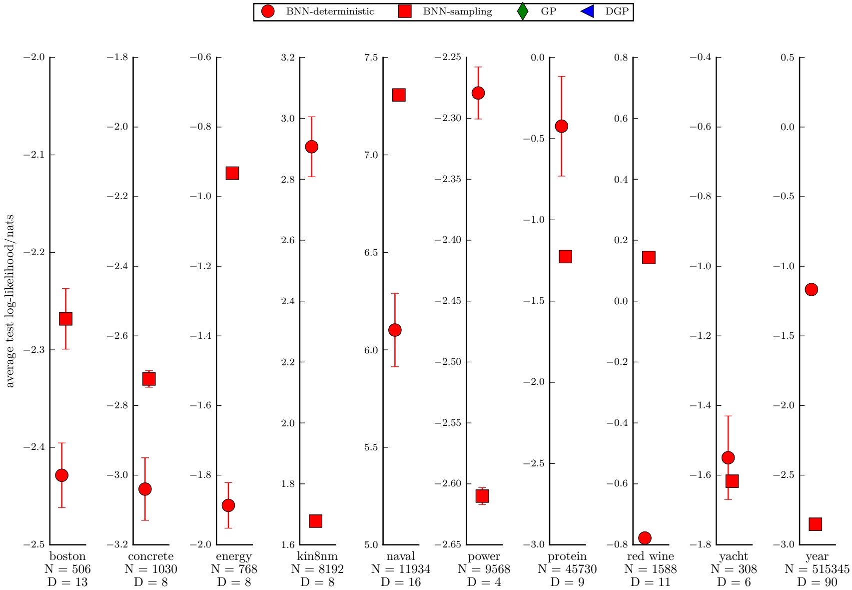
Experiment: Comparison to Bayesian neural networks [Best results]



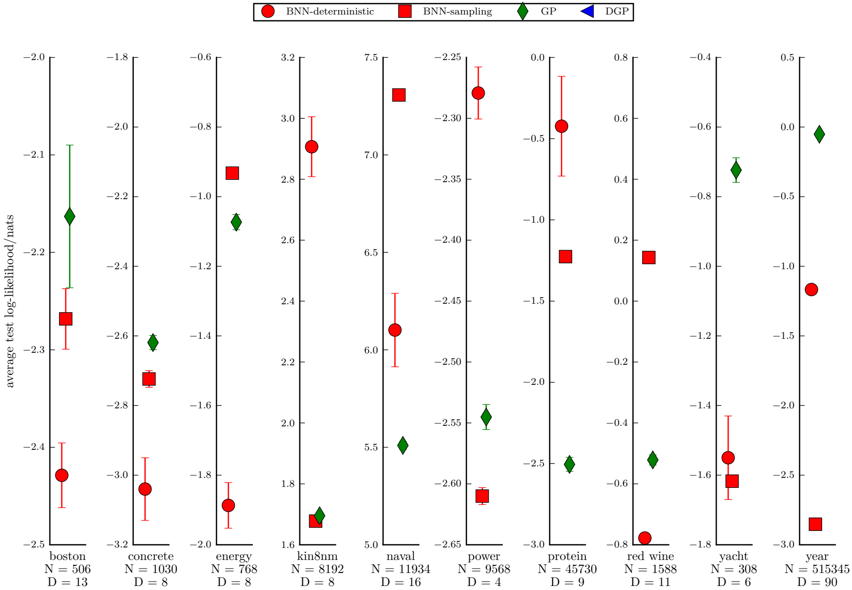
Experiment: Comparison to Bayesian neural networks [Best results]



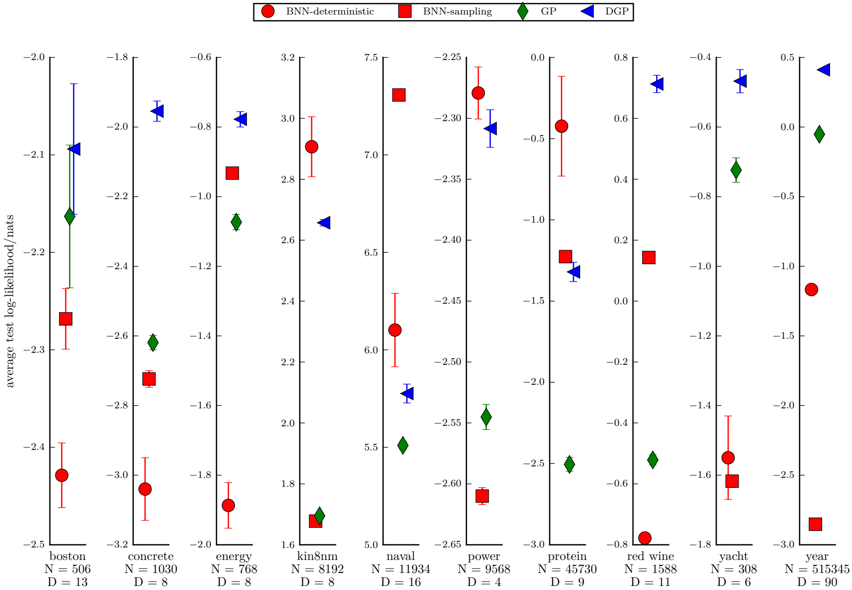
Experiment: Comparison to Bayesian neural networks [Best results]

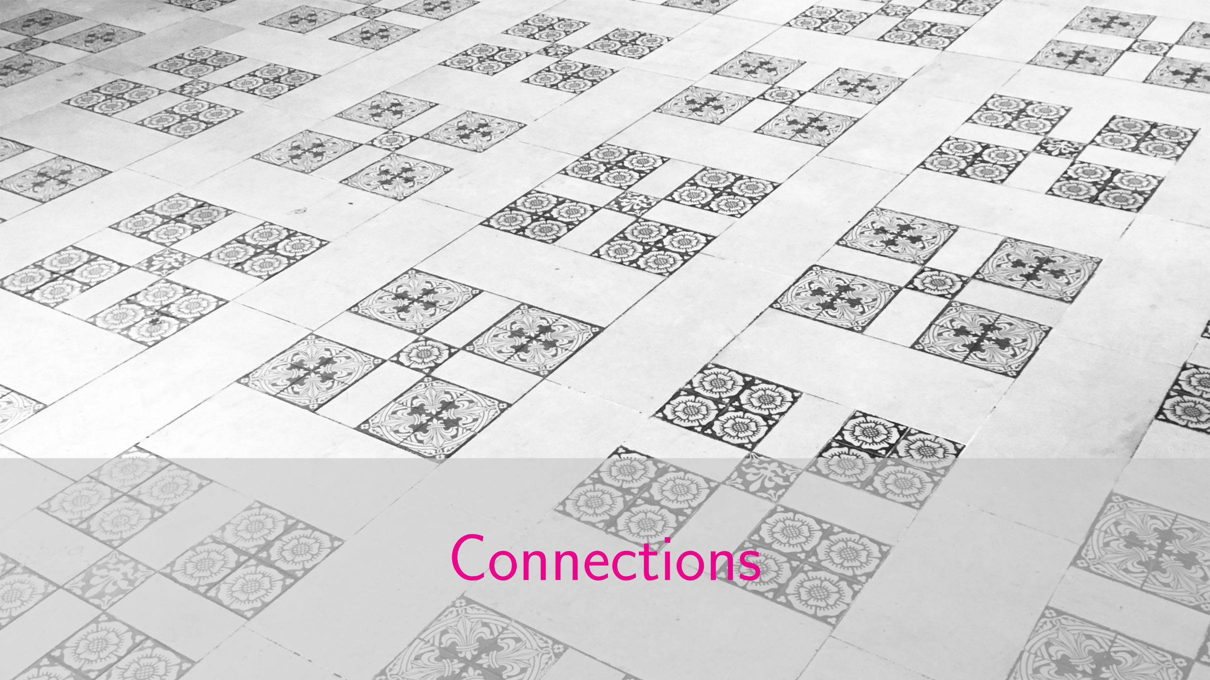


Experiment: Comparison to Bayesian neural networks [Best results]



Experiment: Comparison to Bayesian neural networks [Best results]

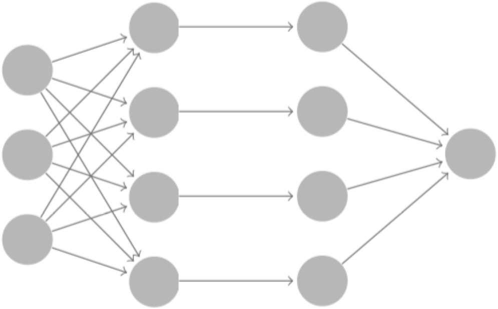




Connections

Infinitely wide neural nets as GPs

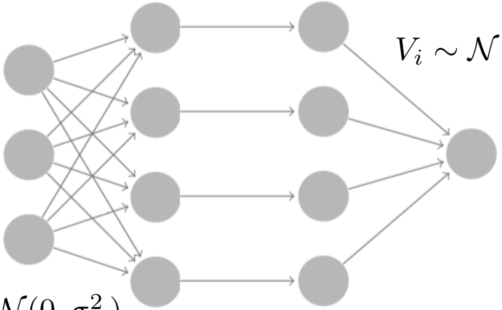
inputs	activations	activities	outputs
x	$a = Wx$	$h = \phi(a)$	$f = Vh$



$$f(x) = \mathrm{NN}(x; W, V)$$

Infinitely wide neural nets as GPs

inputs	activations	activities	outputs
x	$a = Wx$	$h = \phi(a)$	$f = Vh$



$$V_i \sim \mathcal{N}(0, \sigma_v^2 / D)$$

$$f(x) = \mathrm{NN}(x; W, V)$$

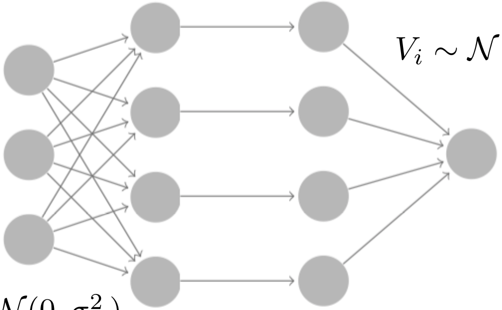
$$W_{i,j} \sim \mathcal{N}(0, \sigma_w^2)$$

Infinitely wide neural nets as GPs

inputs	activations	activities	outputs
x	$a = Wx$	$h = \phi(a)$	$f = Vh$

stops the variance of the output blowing up (cf. initialisation of neural nets)

$$V_i \sim \mathcal{N}(0, \sigma_v^2/D)$$

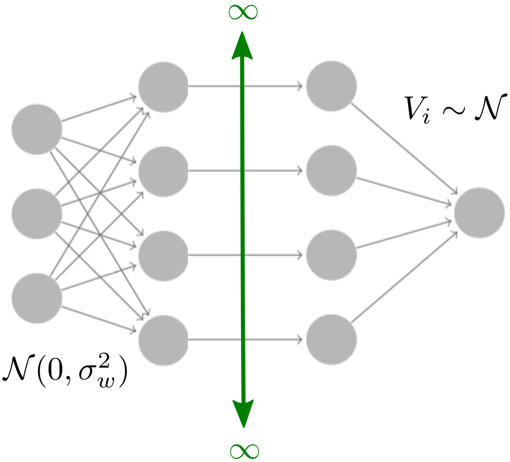


$$f(x) = \mathrm{NN}(x; W, V)$$

$$W_{i,j} \sim \mathcal{N}(0, \sigma_w^2)$$

Infinitely wide neural nets as GPs

inputs	activations	activities	outputs
x	$a = Wx$	$h = \phi(a)$	$f = Vh$



stops the variance of the output blowing up (cf. initialisation of neural nets)

$$V_i \sim \mathcal{N}(0, \sigma_v^2/D)$$

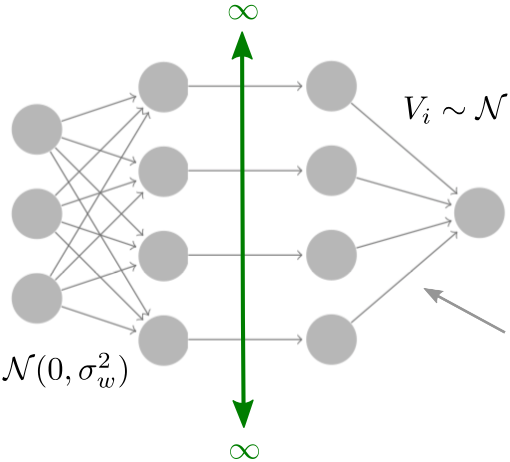
$$f(x) = \mathrm{NN}(x; W, V) \rightarrow \; ?$$

as $D \rightarrow \infty$

$$W_{i,j} \sim \mathcal{N}(0, \sigma_w^2)$$

Infinitely wide neural nets as GPs

inputs	activations	activities	outputs
x	$a = Wx$	$h = \phi(a)$	$f = Vh$



stops the variance of the output blowing up (cf. initialisation of neural nets)

$$V_i \sim \mathcal{N}(0, \sigma_v^2/D)$$

$$f(x) = \mathrm{NN}(x; W, V) \to \textbf{?}$$

as $D \to \infty$

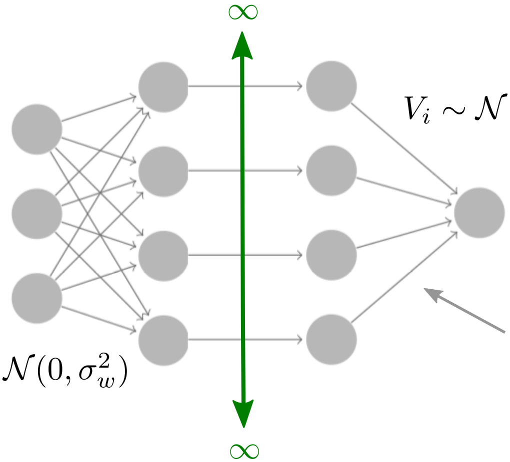
central limit kicks in: sum of independent random variables

$$W_{i,j} \sim \mathcal{N}(0, \sigma_w^2)$$

Neal, 1996

Infinitely wide neural nets as GPs

inputs	activations	activities	outputs
x	$a = Wx$	$h = \phi(a)$	$f = Vh$



stops the variance of the output blowing up (cf. initialisation of neural nets)

$$V_i \sim \mathcal{N}(0, \sigma_v^2 / D)$$

$$f(x) = \mathrm{NN}(x; W, V) \rightarrow f(x) \sim \mathcal{GP}$$

as $D \rightarrow \infty$

central limit kicks in: sum of independent random variables

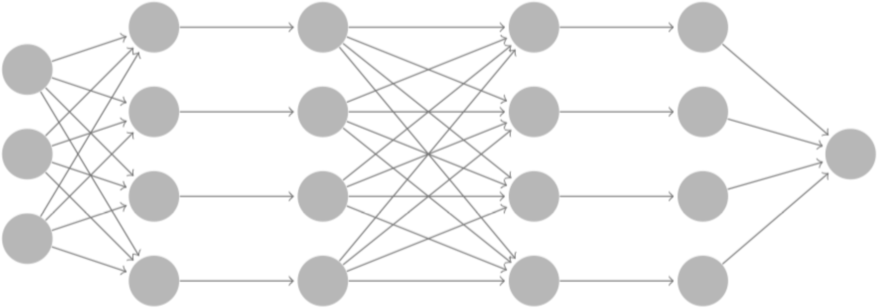
$$W_{i,j} \sim \mathcal{N}(0, \sigma_w^2)$$

Neal, 1996

Infinitely wide neural nets as GPs

inputs activations activities activations activities outputs

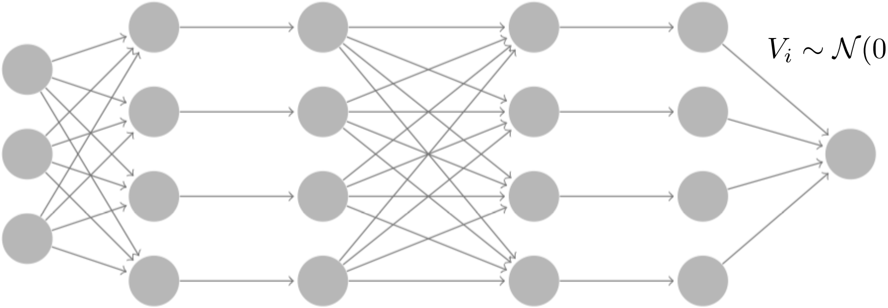
x



Infinitely wide neural nets as GPs

inputs activations activities activations activities outputs

x

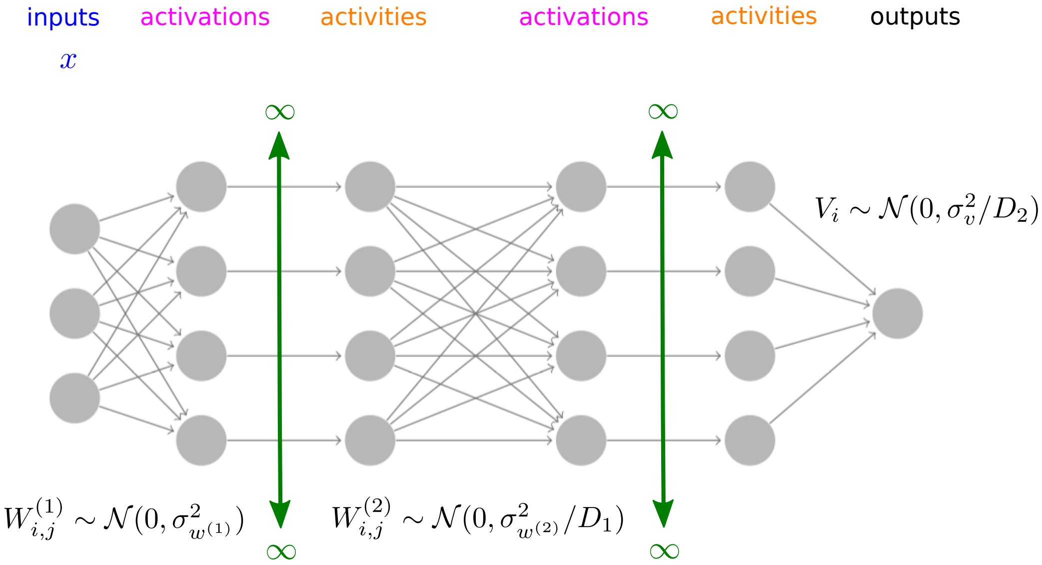


$V_i \sim \mathcal{N}(0, \sigma_v^2 / D_2)$

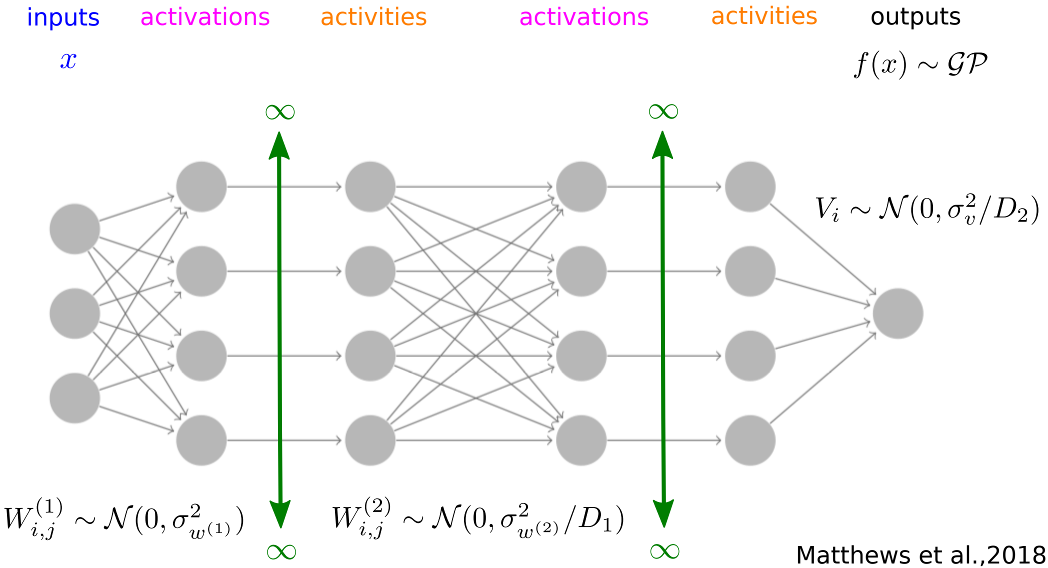
$W_{i,j}^{(1)} \sim \mathcal{N}(0, \sigma_{w^{(1)}}^2)$

$W_{i,j}^{(2)} \sim \mathcal{N}(0, \sigma_{w^{(2)}}^2 / D_1)$

Infinitely wide neural nets as GPs



Infinitely wide neural nets as GPs



Matthews et al.,2018

Gaussian Processes in Disguise

- **Wide neural networks** (perhaps don't need to be so wide) and **CNNs with many features**

Gaussian Processes in Disguise

- **Wide neural networks** (perhaps don't need to be so wide) and **CNNs with many features**
- **linear-in-the-parameters models with Gaussian priors**: linear regression, RBF networks, ...

Gaussian Processes in Disguise

- **Wide neural networks** (perhaps don't need to be so wide) and **CNNs with many features**
- **linear-in-the-parameters models with Gaussian priors**: linear regression, RBF networks, ...
- **Linear Gaussian State Space Model** (Kalman Filter): $x_t = Ax_{t-1} + Q^{1/2}\epsilon_t$ and $y_t = Cx_t + R^{1/2}\eta_t$

Gaussian Processes in Disguise

- **Wide neural networks** (perhaps don't need to be so wide) and **CNNs with many features**
- **linear-in-the-parameters models with Gaussian priors**: linear regression, RBF networks, ...
- **Linear Gaussian State Space Model** (Kalman Filter): $x_t = Ax_{t-1} + Q^{1/2}\epsilon_t$ and $y_t = Cx_t + R^{1/2}\eta_t$
- **Linear stochastic differential equations**: $dX_t = a(t)X_t dt + b(t)dW_t$ where $W_t =$ Wiener Process aka Brownian motion

Gaussian Processes in Disguise

- **Wide neural networks** (perhaps don't need to be so wide) and **CNNs with many features**
- **linear-in-the-parameters models with Gaussian priors**: linear regression, RBF networks, ...
- **Linear Gaussian State Space Model** (Kalman Filter): $x_t = Ax_{t-1} + Q^{1/2}\epsilon_t$ and $y_t = Cx_t + R^{1/2}\eta_t$
- **Linear stochastic differential equations**: $dX_t = a(t)X_t dt + b(t) dW_t$ where $W_t =$ Wiener Process aka Brownian motion
- Krigging (geostatistics), splines (curve fitting), moving average processes, time-frequency analysis, ...

References (hyperlinked)

Gaussian Process Models

- Gaussian Process Latent Variable Models for Visualisation of High Dimensional Data
  Lawrence, NIPS, 2005
- Local Distance Preservation in the GP-LVM through Back Constraints, Lawrence and Quinonero-Candela, ICML 2006

The Automatic Statistician

- The Automatic Statistician, Ghahramani et al., (website link)

GPs for Reinforcement Learning and Control

- PILCO: A Model-Based and Data-Efficient Approach to Policy Search, Deisenroth and Rasmussen, ICML 2011

GPs and Neural Networks

- Bayesian Learning for Neural Networks Neal, 1996
- Gaussian Process Behaviour in Wide Deep Neural Networks, Matthews et al., arXiv, 2018