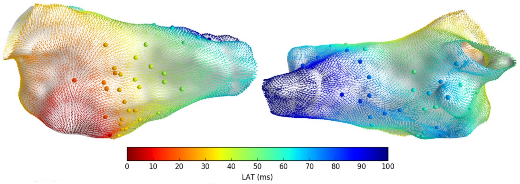
A second introduction to Gaussian Processes

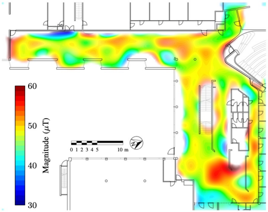
Richard Wilkinson

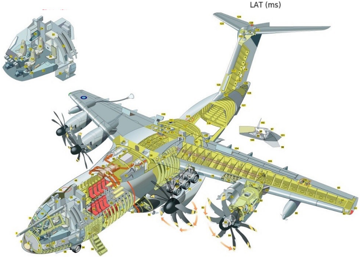
School of Mathematical Sciences
University of Nottingham

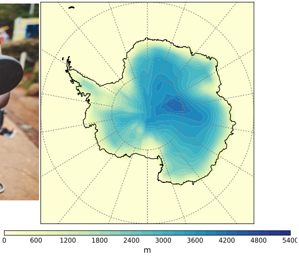
GP summer school
September 2024

Recent GP Applications







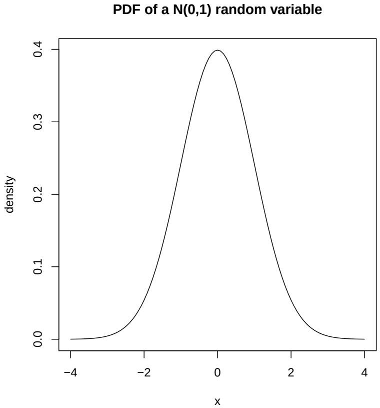


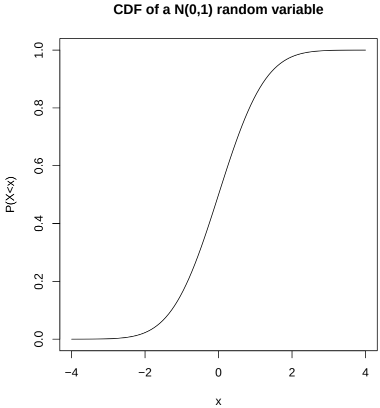
Introduction

- (Multivariate) Gaussian distributions
- Definition of Gaussian processes
- Motivations and derivations
- Difficulties

You can download a copy of these slides from `www.gpss.cc`

Univariate Gaussian distributions



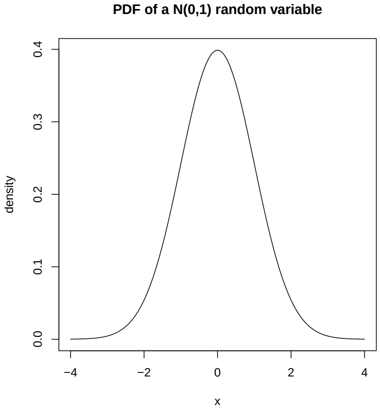


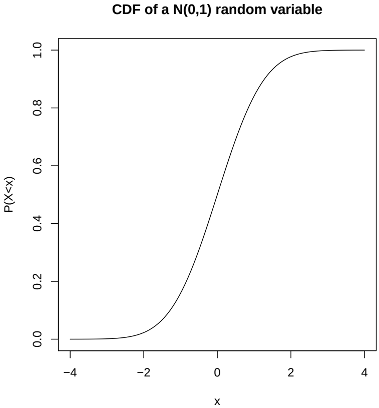
$$Y \sim N(\mu, \sigma^2)$$

PDF: $$f_Y(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right)$$

CDF: $F_Y(y) = \mathbb{P}(Y \leq y)$ not known in closed form

Univariate Gaussian distributions


PDF of a N(0,1) random variable


CDF of a N(0,1) random variable

$$Y \sim N(\mu, \sigma^2)$$

PDF: $$f_Y(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right)$$

CDF: $F_Y(y) = \mathbb{P}(Y \leq y)$ not known in closed form

If $Z \sim N(0, 1)$ then $Y = \mu + \sigma Z \sim N(\mu, \sigma^2)$

Univariate Gaussians

The normal/Gaussian distribution occurs naturally and is convenient mathematically

[1] max. ent. principle: the distribution with the largest entropy should be used as a least-informative default

Univariate Gaussians

The normal/Gaussian distribution occurs naturally and is convenient mathematically

- Family of normal distributions is closed under linear operations.

[1] max. ent. principle: the distribution with the largest entropy should be used as a least-informative default

Univariate Gaussians

The normal/Gaussian distribution occurs naturally and is convenient mathematically

- Family of normal distributions is closed under linear operations.
- Central limit theorem

[1] max. ent. principle: the distribution with the largest entropy should be used as a least-informative default

Univariate Gaussians

The normal/Gaussian distribution occurs naturally and is convenient mathematically

- Family of normal distributions is closed under linear operations.
- Central limit theorem
- Maximum entropy/surprisal: $N(\mu, \sigma^2)$ has maximum entropy[1] of any distribution with mean μ and variance σ^2

[1] max. ent. principle: the distribution with the largest entropy should be used as a least-informative default

Univariate Gaussians

The normal/Gaussian distribution occurs naturally and is convenient mathematically

- Family of normal distributions is closed under linear operations.
- Central limit theorem
- Maximum entropy/surprisal: $N(\mu, \sigma^2)$ has maximum entropy[1] of any distribution with mean μ and variance σ^2
- Infinite divisibility

[1] max. ent. principle: the distribution with the largest entropy should be used as a least-informative default

Univariate Gaussians

The normal/Gaussian distribution occurs naturally and is convenient mathematically

- Family of normal distributions is closed under linear operations.
- Central limit theorem
- Maximum entropy/surprisal: $N(\mu, \sigma^2)$ has maximum entropy[1] of any distribution with mean μ and variance σ^2
- Infinite divisibility
- If Y and Z are jointly normally distributed and are uncorrelated, then they are independent

[1] max. ent. principle: the distribution with the largest entropy should be used as a least-informative default

Univariate Gaussians

The normal/Gaussian distribution occurs naturally and is convenient mathematically

- Family of normal distributions is closed under linear operations.
- Central limit theorem
- Maximum entropy/surprisal: $N(\mu, \sigma^2)$ has maximum entropy[1] of any distribution with mean μ and variance σ^2
- Infinite divisibility
- If Y and Z are jointly normally distributed and are uncorrelated, then they are independent
- Square-loss functions lead to procedures that have a Gaussian probabilistic interpretation
  eg Fit model $f_\beta(x)$ to data y by mimizing $\sum(y_i - f_\beta(x_i))^2$ is equivalent to maximum likelihood estimation under the assumption that $y = f_\beta(x) + \epsilon$ where $\epsilon \sim N(0, \sigma^2)$.

[1] max. ent. principle: the distribution with the largest entropy should be used as a least-informative default

Multivariate Gaussian distributions

'Multivariate' = two or more random variables

Multivariate Gaussian distributions

'Multivariate' = two or more random variables

Suppose $Y \in \mathbb{R}^d$ has a multivariate Gaussian distribution with

- **mean vector** $\mu \in \mathbb{R}^d$
- **covariance matrix** $\Sigma \in \mathbb{R}^{d \times d}$.

Write

$$Y \sim N_d(\mu, \Sigma)$$

Multivariate Gaussian distributions

'Multivariate' = two or more random variables

Suppose $Y \in \mathbb{R}^d$ has a multivariate Gaussian distribution with

- **mean vector** $\mu \in \mathbb{R}^d$
- **covariance matrix** $\Sigma \in \mathbb{R}^{d \times d}$.

Write

$$Y \sim N_d(\mu, \Sigma)$$

Bivariate Gaussian: d=2

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} \qquad \mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 \\ \rho_{21}\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

Multivariate Gaussian distributions

'Multivariate' = two or more random variables

Suppose $Y \in \mathbb{R}^d$ has a multivariate Gaussian distribution with

- **mean vector** $\mu \in \mathbb{R}^d$
- **covariance matrix** $\Sigma \in \mathbb{R}^{d\times d}$.

Write

$$Y \sim N_d(\mu, \Sigma)$$

Bivariate Gaussian: d=2

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} \qquad \mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 \\ \rho_{21}\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

$$\mathbb{V}\text{ar}(Y_i) = \sigma_i^2 \quad \mathbb{C}\text{ov}(Y_1, Y_2) = \rho_{12}\sigma_1\sigma_2 \quad \text{Cor}(Y_1, Y_2) = \rho_{12}$$

Multivariate Gaussian distributions

'Multivariate' = two or more random variables

Suppose $Y \in \mathbb{R}^d$ has a multivariate Gaussian distribution with

- **mean vector** $\mu \in \mathbb{R}^d$
- **covariance matrix** $\Sigma \in \mathbb{R}^{d \times d}$.

Write

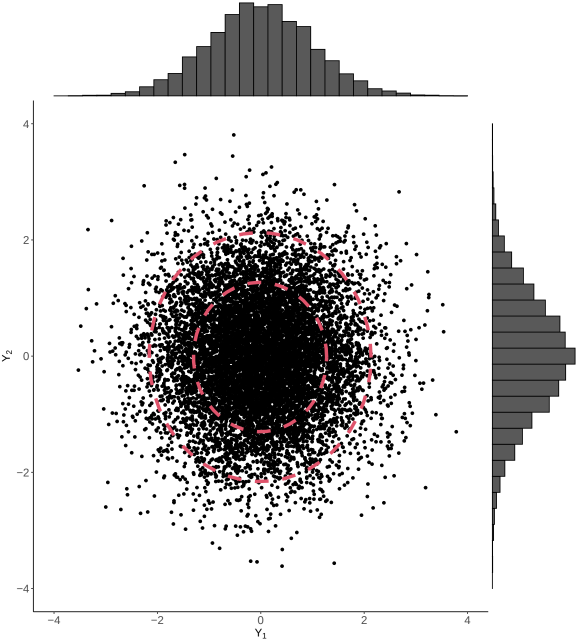
$$Y \sim N_d(\mu, \Sigma)$$

Bivariate Gaussian: d=2

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} \qquad \mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 \\ \rho_{21}\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

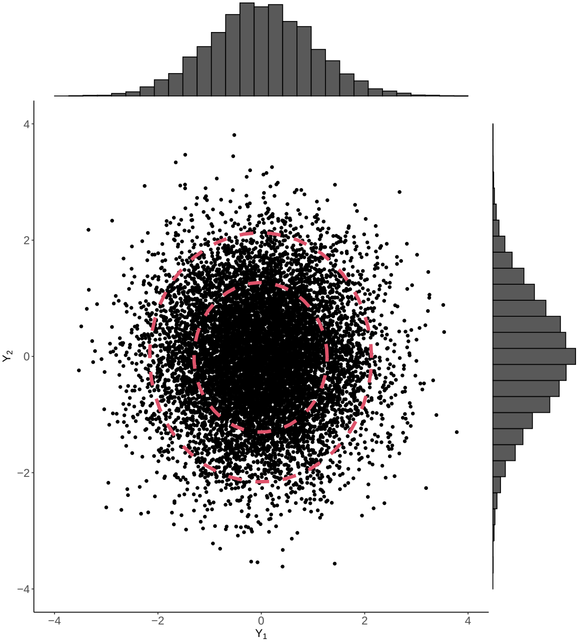
$$\mathbb{V}\text{ar}(Y_i) = \sigma_i^2 \quad \mathbb{C}\text{ov}(Y_1, Y_2) = \rho_{12}\sigma_1\sigma_2 \quad \text{Cor}(Y_1, Y_2) = \rho_{12}$$

pdf: $$f(y \mid \mu, \Sigma) = |\Sigma|^{-\frac{1}{2}}(2\pi)^{-\frac{d}{2}} \exp\left(-\frac{1}{2}(y-\mu)^\top \Sigma^{-1}(y-\mu)\right)$$



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

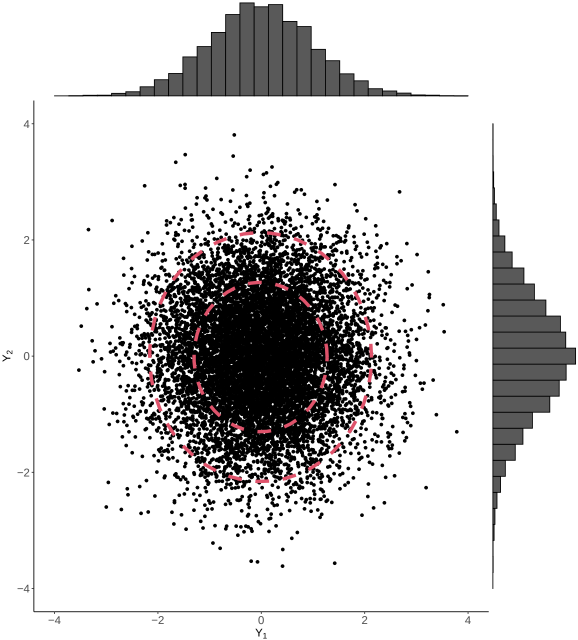
$$\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$Cor(Y_1, Y_2) = 0$
hence Y_1
independent of Y_2



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

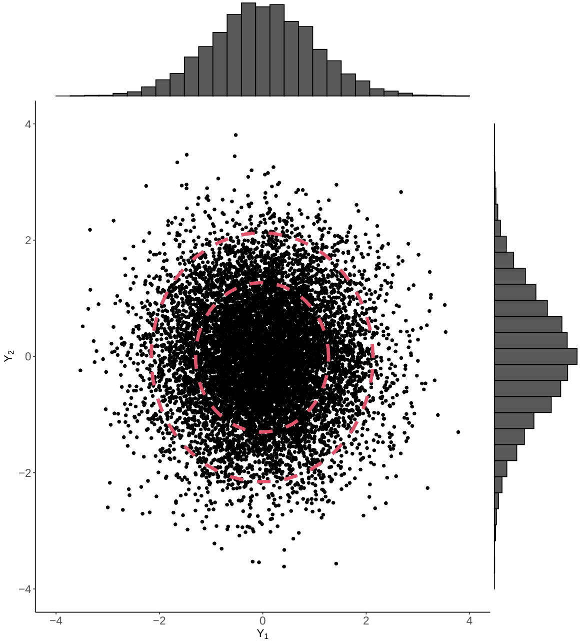
$$\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$Cor(Y_1, Y_2) = 0$
hence Y_1
independent of Y_2

Marginal
distributions:

$$Y_1 \sim N(0,1)$$

$$Y_2 \sim N(0,1)$$



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$Cor(Y_1, Y_2) = 0$
hence Y_1
independent of Y_2

Marginal distributions:

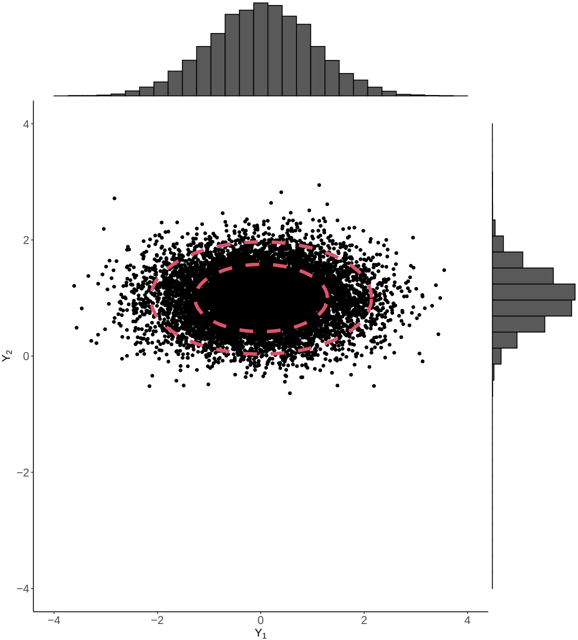
$$Y_1 \sim N(0, 1)$$

$$Y_2 \sim N(0, 1)$$

Conditional distributions:

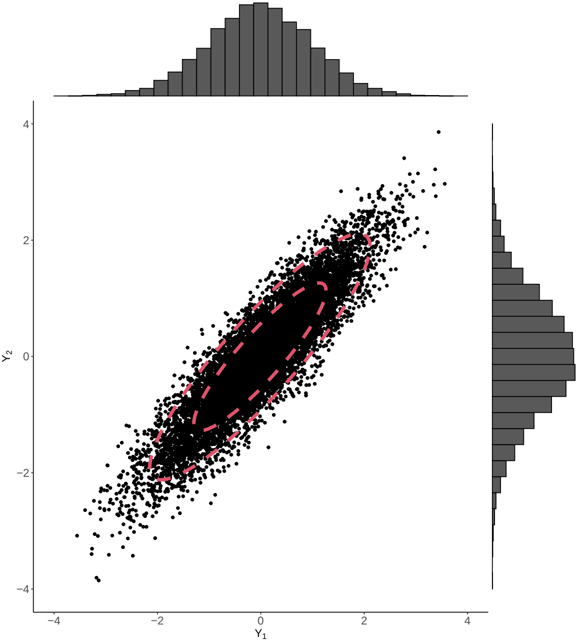
$$Y_1 | Y_2 \sim N(0, 1)$$

$$Y_2 | Y_1 \sim N(0, 1)$$



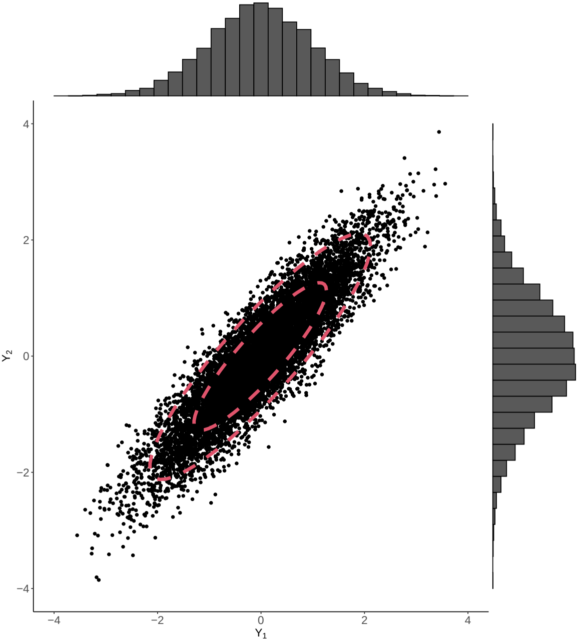
$$\mu = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 0.2 \end{pmatrix}$$



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}$$



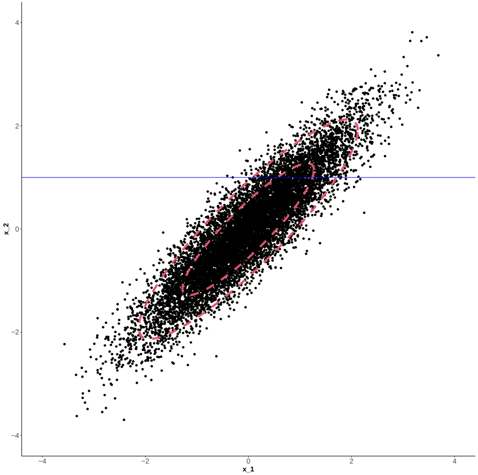
$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}$$

Marginal distributions:

$$Y_1 \sim N(0, 1)$$

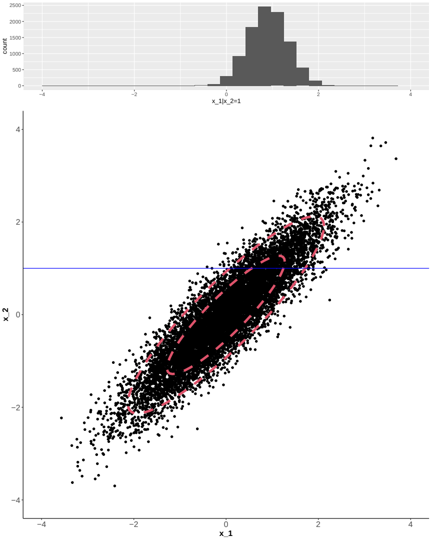
$$Y_2 \sim N(0, 1)$$



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}$$

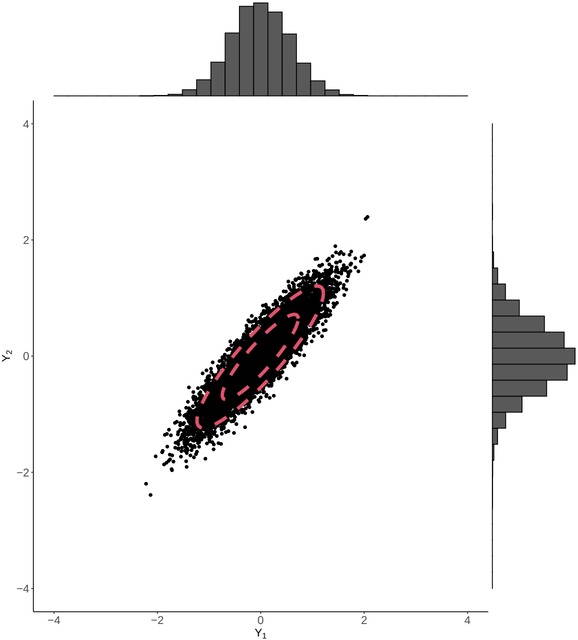
What is the conditional distribution of $Y_1 | Y_2 = 1$?



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

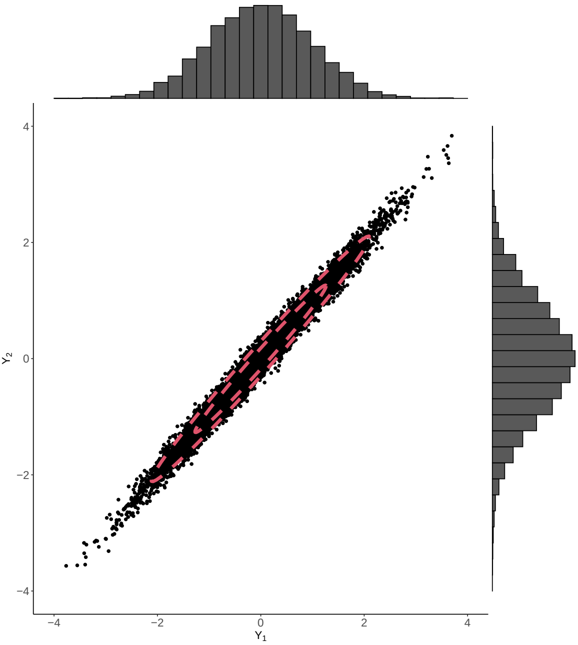
$$\Sigma = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}$$

$$Y_1 | Y_2 = 1 \sim N(0.9, 0.19)$$



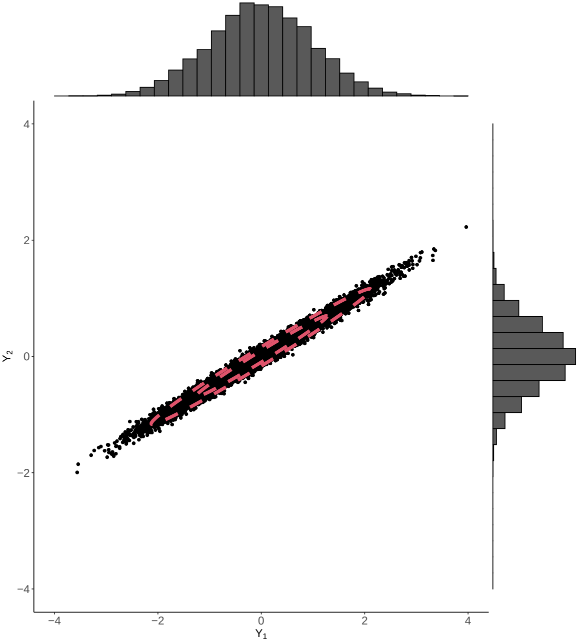
$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\Sigma = \frac{1}{3} \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}$$



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 1 & 0.99 \\ 0.99 & 1 \end{pmatrix}$$



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 1 & 0.54 \\ 0.54 & 0.3 \end{pmatrix}$$

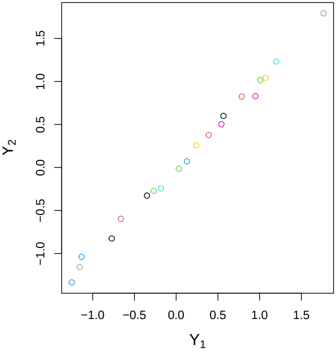
$Cor(Y_1, Y_2) =$
$0.54/\sqrt{}(0.3) =$
0.99

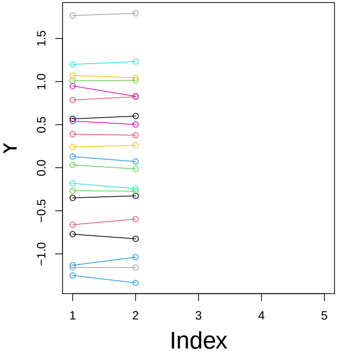
More pictures

Hard to visualise in dimensions > 2, so stack points next to each other.

More pictures

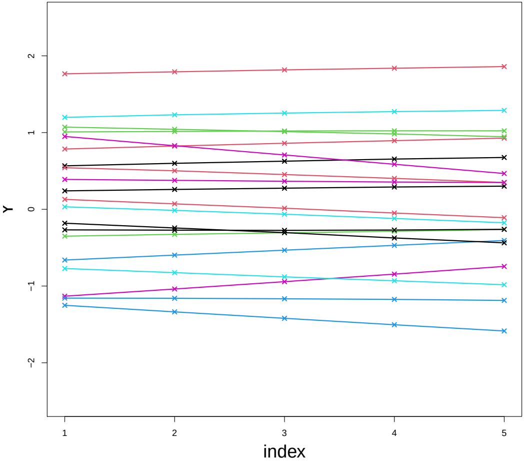
Hard to visualise in dimensions > 2, so stack points next to each other.
So for 2d instead of we have





Consider $d = 5$ with

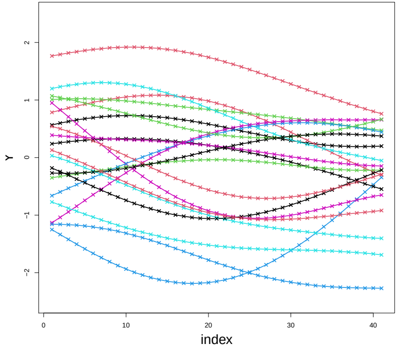
$$\mu = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} 1 & 0.99 & 0.98 & 0.97 & 0.96 \\ 0.99 & 1 & 0.99 & 0.98 & 0.97 \\ 0.98 & 0.99 & 1 & 0.99 & 0.98 \\ 0.97 & 0.98 & 0.99 & 1 & 0.99 \\ 0.96 & 0.97 & 0.98 & 0.99 & 1 \end{pmatrix}$$



Each line is one sample.

$d = 50$

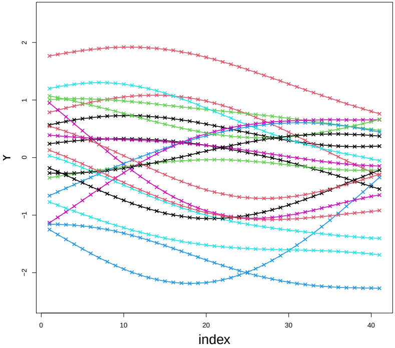
$$\mu = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} 1 & 0.99 & 0.98 & 0.97 & 0.96 & \cdots \\ 0.99 & 1 & 0.99 & 0.98 & 0.97 & \cdots \\ 0.98 & 0.99 & 1 & 0.99 & 0.98 & \cdots \\ 0.97 & 0.98 & 0.99 & 1 & 0.99 & \cdots \\ 0.96 & 0.97 & 0.98 & 0.99 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}$$



Each line is one sample.

$d = 50$

$$\mu = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} 1 & 0.99 & 0.98 & 0.97 & 0.96 & \cdots \\ 0.99 & 1 & 0.99 & 0.98 & 0.97 & \cdots \\ 0.98 & 0.99 & 1 & 0.99 & 0.98 & \cdots \\ 0.97 & 0.98 & 0.99 & 1 & 0.99 & \cdots \\ 0.96 & 0.97 & 0.98 & 0.99 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}$$



Each line is one sample.

We can think of Gaussian processes as an infinite dimensional distribution over functions - all we need to do is change the indexing

- We can let the index be x and let x take values in $\mathbb{R}$.

Gaussian processes

A stochastic process is a collection of random variables indexed by some variable $x \in \mathcal{X}$

$$y = \{y(x) : x \in \mathcal{X}\}$$

Gaussian processes

A stochastic process is a collection of random variables indexed by some variable $x \in \mathcal{X}$

$$y = \{y(x) : x \in \mathcal{X}\}$$

Usually $y(x) \in \mathbb{R}$ and $\mathcal{X} \subset \mathbb{R}^n$

Gaussian processes

A stochastic process is a collection of random variables indexed by some variable $x \in \mathcal{X}$

$$y = \{y(x) : x \in \mathcal{X}\}$$

Usually $y(x) \in \mathbb{R}$ and $\mathcal{X} \subset \mathbb{R}^n$ - think of y as a function of x.

Gaussian processes

A stochastic process is a collection of random variables indexed by some variable $x \in \mathcal{X}$

$$y = \{y(x) : x \in \mathcal{X}\}$$

Usually $y(x) \in \mathbb{R}$ and $\mathcal{X} \subset \mathbb{R}^n$ - think of y as a function of x.
If $|\mathcal{X}| = \infty$, y is an infinite dimensional process, e.g., $|\mathcal{X}| = [0, 1]$

Gaussian processes

A stochastic process is a collection of random variables indexed by some variable $x \in \mathcal{X}$

$$y = \{y(x) : x \in \mathcal{X}\}$$

Usually $y(x) \in \mathbb{R}$ and $\mathcal{X} \subset \mathbb{R}^n$ - think of y as a function of x.
If $|\mathcal{X}| = \infty$, y is an infinite dimensional process, e.g., $|\mathcal{X}| = [0, 1]$

Thankfully, the law of y is uniquely determined by the finite dimensional distributions (FDDs), i.e., for all $x_1, \ldots, x_n$ and for all $n \in \mathbb{N}$

$$\mathbb{P}(y(x_1) \leq c_1, \ldots, y(x_n) \leq c_n)$$

Gaussian processes

A stochastic process is a collection of random variables indexed by some variable $x \in \mathcal{X}$

$$y = \{y(x) : x \in \mathcal{X}\}$$

Usually $y(x) \in \mathbb{R}$ and $\mathcal{X} \subset \mathbb{R}^n$ - think of y as a function of x.
If $|\mathcal{X}| = \infty$, y is an infinite dimensional process, e.g., $|\mathcal{X}| = [0, 1]$

Thankfully, the law of y is uniquely determined by the finite dimensional distributions (FDDs), i.e., for all $x_1, \ldots, x_n$ and for all $n \in \mathbb{N}$

$$\mathbb{P}(y(x_1) \leq c_1, \ldots, y(x_n) \leq c_n)$$

A **Gaussian process** is a stochastic process with Gaussian FDDs, i.e.,

$$(y(x_1), \ldots, y(x_n)) \sim N_n(\mu, \Sigma)$$

Gaussian processes

A stochastic process is a collection of random variables indexed by some variable $x \in \mathcal{X}$

$$y = \{y(x) : x \in \mathcal{X}\}$$

Usually $y(x) \in \mathbb{R}$ and $\mathcal{X} \subset \mathbb{R}^n$ - think of y as a function of x.
If $|\mathcal{X}| = \infty$, y is an infinite dimensional process, e.g., $|\mathcal{X}| = [0, 1]$

Thankfully, the law of y is uniquely determined by the finite dimensional distributions (FDDs), i.e., for all $x_1, \ldots, x_n$ and for all $n \in \mathbb{N}$

$$\mathbb{P}(y(x_1) \leq c_1, \ldots, y(x_n) \leq c_n)$$

A **Gaussian process** is a stochastic process with Gaussian FDDs, i.e.,

$$(y(x_1), \ldots, y(x_n)) \sim N_n(\mu, \Sigma)$$

Write $y(\cdot) \sim GP$ to denote that we model the *function* y as a GP.

Mean and covariance function

To fully specify the law of a Gaussian *distribution* we only need the mean and variance.

$$X \sim N(\mu, \Sigma)$$

Mean and covariance function

To fully specify the law of a Gaussian *distribution* we only need the mean and variance.

$$X \sim N(\mu, \Sigma)$$

To fully specify the law of a Gaussian *process*, we need to specify mean and covariance functions.

Mean and covariance function

To fully specify the law of a Gaussian *distribution* we only need the mean and variance.

$$X \sim N(\mu, \Sigma)$$

To fully specify the law of a Gaussian *process*, we need to specify mean and covariance functions.

$$y(\cdot) \sim GP(m(\cdot), k(\cdot, \cdot))$$

Mean and covariance function

To fully specify the law of a Gaussian *distribution* we only need the mean and variance.

$$X \sim N(\mu, \Sigma)$$

To fully specify the law of a Gaussian *process*, we need to specify mean and covariance functions.

$$y(\cdot) \sim GP(m(\cdot), k(\cdot, \cdot))$$

where

$$\begin{aligned} \mathbb{E}(y(x)) &= m(x) \\ \mathbb{C}\text{ov}(y(x), y(x')) &= k(x, x') \end{aligned}$$

Specifying the mean function

We are free to choose the mean $m(x) = \mathbb{E}(y(x))$ and covariance $k(x, x') = \mathbb{C}\mathrm{ov}(y(x), y(x'))$ functions however we like (e.g. trial and error), subject to some ‘rules’:

Specifying the mean function

We are free to choose the mean $m(x) = \mathbb{E}(y(x))$ and covariance $k(x, x') = \mathbb{C}\text{ov}(y(x), y(x'))$ functions however we like (e.g. trial and error), subject to some 'rules':

- We can use any mean function we want:

  $$m(x) = \mathbb{E}(y(x))$$

  Most popular choices are $m(x) = 0$ or $m(x) = \textit{const}$ for all x, or $m(x) = \beta^\top x$

Covariance functions

We usually use a covariance function that is a function of the indexes/locations

$$k(x, x') = \mathbb{C}\text{ov}(y(x), y(x')),$$

Covariance functions

We usually use a covariance function that is a function of the indexes/locations

$$k(x, x') = \mathbb{C}\text{ov}(y(x), y(x')),$$

k must be a positive semi-definite function, i.e., lead to valid covariance matrices:

- Given locations $x_1, \dots, x_n$, the $n \times n$ Gram matrix K with $K_{ij} = k(x_i, x_j)$

$$\begin{pmatrix} k(x_1, x_1) & k(x_1, x_2) & \dots & k(x_1, x_n) \\ \vdots & & & \vdots \\ k(x_n, x_1) & k(x_n, x_2) & \dots & k(x_n, x_n) \end{pmatrix}$$

  must be a positive semi-definite matrix (ie a covariance matrix).

Covariance functions

We usually use a covariance function that is a function of the indexes/locations

$$k(x, x') = \mathbb{C}\text{ov}(y(x), y(x')),$$

k must be a positive semi-definite function, i.e., lead to valid covariance matrices:

- Given locations $x_1, \dots, x_n$, the $n \times n$ Gram matrix K with $K_{ij} = k(x_i, x_j)$

$$\begin{pmatrix} k(x_1, x_1) & k(x_1, x_2) & \dots & k(x_1, x_n) \\ \vdots & & & \vdots \\ k(x_n, x_1) & k(x_n, x_2) & \dots & k(x_n, x_n) \end{pmatrix}$$

  must be a positive semi-definite matrix (ie a covariance matrix).
- This can be problematic...
  - Difficult to create semi-definite functions $k(x, x')$

We often assume k is a function of only the distance between locations

$$\mathbb{C}\mathrm{ov}(y(x), y(x')) = k(x - x')$$

which results in a stationary process.

We often assume k is a function of only the distance between locations

$$\mathbb{C}\text{ov}(y(x), y(x')) = k(x - x')$$

which results in a stationary process.

If $\mathbb{C}\text{ov}(y(x), y(x')) = k(||x - x'||)$ the covariance function is said to be isotropic.

We often assume k is a function of only the distance between locations

$$\mathbb{C}\text{ov}(y(x), y(x')) = k(x - x')$$

which results in a stationary process.

If $\mathbb{C}\text{ov}(y(x), y(x')) = k(||x - x'||)$ the covariance function is said to be isotropic.

The covariance function determines the *nature* of the GP.

- k determines the hypothesis space/space of functions

We often assume k is a function of only the distance between locations

$$\mathbb{C}\text{ov}(y(x), y(x')) = k(x - x')$$

which results in a stationary process.

If $\mathbb{C}\text{ov}(y(x), y(x')) = k(||x - x'||)$ the covariance function is said to be isotropic.

The covariance function determines the *nature* of the GP.

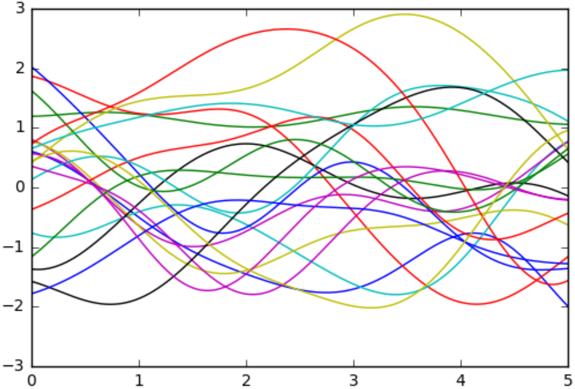
- k determines the hypothesis space/space of functions

We usually build k by selecting from a small candidate set of functions (that contain free parameters), and modifying them according to some rules.

Examples

RBF/Squared-exponential/exponentiated quadratic

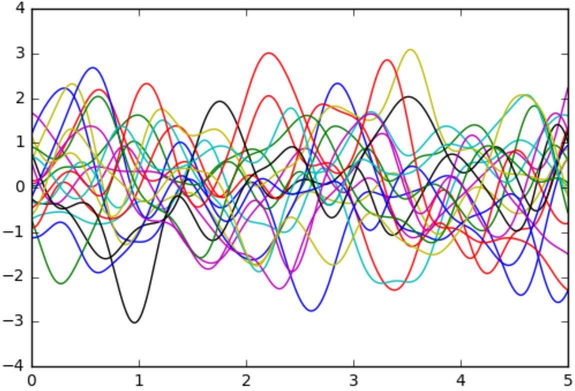
$$k(x, x') = \exp\left(-\frac{1}{2}(x - x')^2\right)$$



Examples

RBF/Squared-exponential/exponentiated quadratic

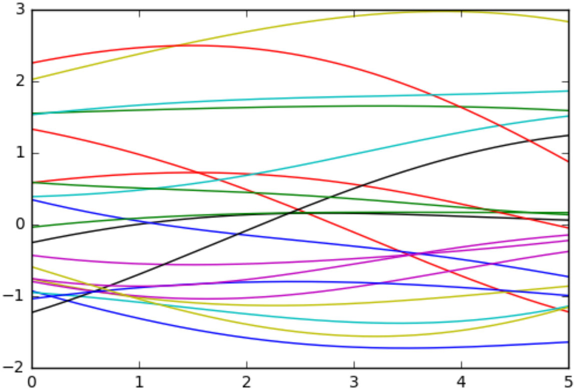
$$k(x, x') = \exp\left(-\frac{1}{2}\frac{(x - x')^2}{\color{red}{0.25}^2}\right)$$



Examples

RBF/Squared-exponential/exponentiated quadratic

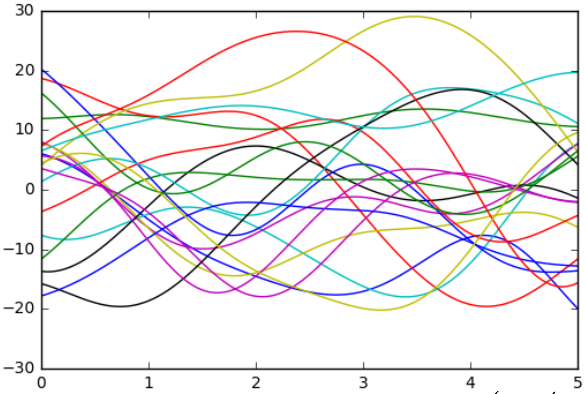
$$k(x, x') = \exp\left(-\frac{1}{2}\frac{(x - x')^2}{4^2}\right)$$



Examples

RBF/Squared-exponential/exponentiated quadratic

$$k(x, x') = 100 \exp\left(-\frac{1}{2}(x - x')^2\right)$$

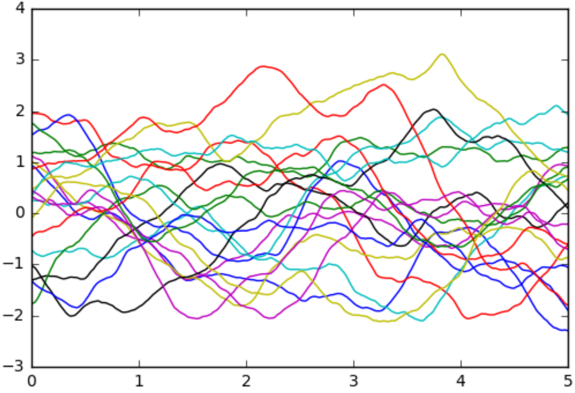


In general $\quad k(x, x') = \tau^2 \exp\left(-\frac{1}{2}\frac{(x - x')^2}{\lambda^2}\right)$

Examples

Matern 3/2

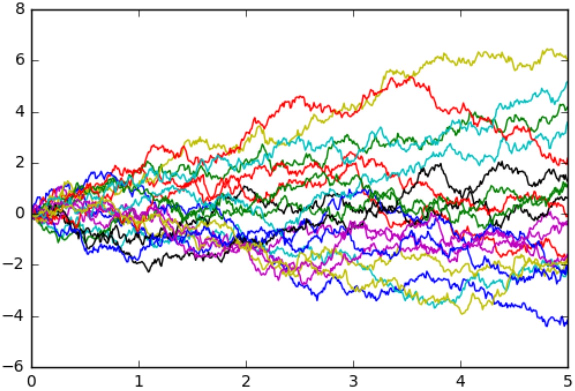
$$k(x, x') \sim (1 + |x - x'|) \exp\left(-|x - x'|\right)$$



Examples

Brownian motion

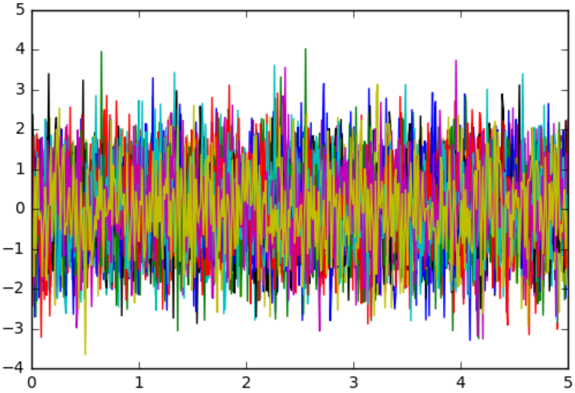
$$k(x, x') = \min(x, x')$$



Examples

White noise

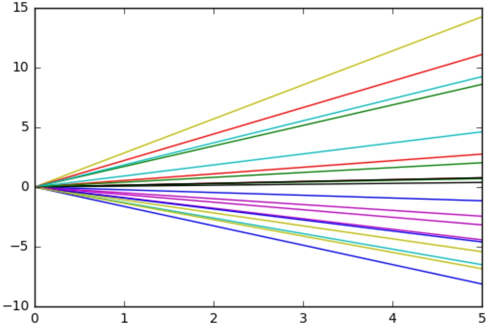
$$k(x, x') = \begin{cases} 1 \text{ if } x = x' \\ 0 \text{ otherwise} \end{cases}$$



Examples

A final example:

$$k(x, x') = xx'$$

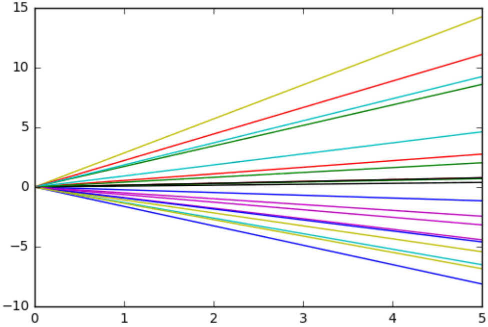


What is happening?

Examples

A final example:

$$k(x, x') = xx'$$



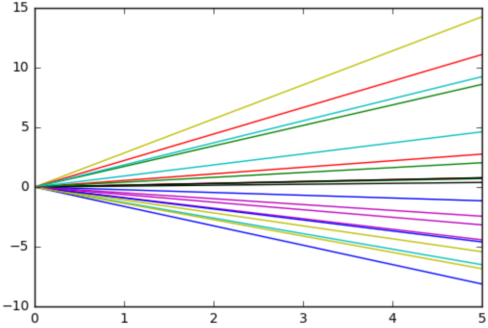
What is happening?

Suppose $y(x) = cx$ where $c \sim N(0, 1)$.

Examples

A final example:

$$k(x, x') = xx'$$



What is happening?

Suppose $y(x) = cx$ where $c \sim N(0, 1)$.

Then

$$\begin{aligned}\mathbb{C}\text{ov}(y(x), y(x')) &= \mathbb{C}\text{ov}(cx, cx') = x\mathbb{C}\text{ov}(c, c)x' \\ &= xx'\end{aligned}$$

So $y(\cdot) \sim GP(0, k(x, x'))$ with $k(x, x') = xx'$

Rules for combining covariance functions

Let k_1 and k_2 be valid covariance functions. Then

- $k_1 + k_2$ is a valid covariance function (even if k_1 and k_2 take different inputs).
- $k_1 k_2$ is a valid covariance function (also true if different inputs).
- For any function g, $k_1(g(x), g(x'))$ is a valid covariance function as is $g(x)k_1(x, x')g(x')$

Using these rules we can combine our standard set of positive definite covariance functions to create a richer family of covariance functions.

Choosing kernels and hyperparameters

GP properties are inherited primarily from the covariance function k.

- Continuity
- Differentiability
- Variance and length-scale

[2] f is mean square cts at x if for all sequences $x_k \to x$ we have $\mathbb{E}(f(x_k) - f(x))^2 \to 0$

Choosing kernels and hyperparameters

GP properties are inherited primarily from the covariance function k.

- Continuity
  - $f(x) \sim GP$ is (mean square[2]) continuous at x^* ifF $k(x, x')$ and $m(x)$ are continuous at $x = x' = x^*$
  - For stationary kernels, only require continuity at $k(0)$
- Differentiability
  - $f(x) \sim GP$ is (mean square) differentiable if $k'(x, x') = \frac{\partial^2}{\partial x \partial x'} k(x, x')$ exists.
- Variance and length-scale

[2] f is mean square cts at x if for all sequences $x_k \to x$ we have $\mathbb{E}(f(x_k) - f(x))^2 \to 0$

Choosing kernels and hyperparameters

GP properties are inherited primarily from the covariance function k.

- Continuity
  - $f(x) \sim GP$ is (mean square[2]) continuous at x^* ifF $k(x,x')$ and $m(x)$ are continuous at $x = x' = x^*$
  - For stationary kernels, only require continuity at $k(0)$
- Differentiability
  - $f(x) \sim GP$ is (mean square) differentiable if $k'(x,x') = \frac{\partial^2}{\partial x \partial x'} k(x,x')$ exists.
- Variance and length-scale controlled by hyper-parameters $k = k_\psi$:
  - how much f varies between samples
  - how fast $f(x)$ changes with x within a sample.

[2] f is mean square cts at x if for all sequences $x_k \to x$ we have $\mathbb{E}(f(x_k) - f(x))^2 \to 0$

Choosing kernels and hyperparameters

GP properties are inherited primarily from the covariance function k.

- Continuity
  - $f(x) \sim GP$ is (mean square[2]) continuous at x^* ifF $k(x, x')$ and $m(x)$ are continuous at $x = x' = x^*$
  - For stationary kernels, only require continuity at $k(0)$
- Differentiability
  - $f(x) \sim GP$ is (mean square) differentiable if $k'(x, x') = \frac{\partial^2}{\partial x \partial x'} k(x, x')$ exists.
- Variance and length-scale controlled by hyper-parameters $k = k_\psi$:
  - how much f varies between samples
  - how fast $f(x)$ changes with x within a sample.

Typically choose the family of kernels by

- measures of fit (marginal likelihood, Bayes factors, ...)
- predictive skill (held-out data, cross-validation, ...)

Choose hyperparameters by maximum likelihood, Bayes, etc.

[2] f is mean square cts at x if for all sequences $x_k \to x$ we have $\mathbb{E}(f(x_k) - f(x))^2 \to 0$

Why use Gaussian processes?

Why would we want to use this very restricted class of model?

Gaussian **distributions** have several properties that make them easy to work with:

Why use Gaussian processes?

Why would we want to use this very restricted class of model?

Gaussian **distributions** have several properties that make them easy to work with:

Proposition:

$$Y \sim N_d(\mu, \Sigma) \text{ if and only if } AY \sim N_p(A\mu, A\Sigma A^\top)$$

for all $A \in \mathbb{R}^{p \times d}$.

Why use Gaussian processes?

Why would we want to use this very restricted class of model?

Gaussian **distributions** have several properties that make them easy to work with:

Proposition:

$$Y \sim N_d(\mu, \Sigma) \text{ if and only if } AY \sim N_p(A\mu, A\Sigma A^\top)$$

for all $A \in \mathbb{R}^{p \times d}$.
So sums of Gaussians are Gaussian, and marginal distributions of multivariate Gaussians are still Gaussian.

Property 2: Conditional distributions are still Gaussian

Suppose

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} \sim N_2(\mu, \Sigma)$$

where

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$$

Property 2: Conditional distributions are still Gaussian

Suppose

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} \sim N_2(\mu, \Sigma)$$

where

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$$

Then

$$Y_2 \mid Y_1 = y_1 \sim N\left(\mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(y_1 - \mu_1), \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\right)$$

Proof:

$$\pi(y_2|y_1) = \frac{\pi(y_1, y_2)}{\pi(y_1)} \propto \pi(y_1, y_2)$$

Proof:

$$\begin{aligned}\pi(y_2|y_1) &= \frac{\pi(y_1, y_2)}{\pi(y_1)} \propto \pi(y_1, y_2)\\ &\propto \exp\left[-\frac{1}{2}(y-\mu)^\top \Sigma^{-1}(y-\mu)\right]\\ &= \exp\left[-\frac{1}{2}\left(\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} - \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}\right)^\top \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} \cdots\right]\end{aligned}$$

where

$$\Sigma^{-1} := Q := \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix}$$

Proof:

$$
\begin{aligned}
\pi(y_2|y_1) = \frac{\pi(y_1, y_2)}{\pi(y_1)} &\propto \pi(y_1, y_2) \\
&\propto \exp\left[-\frac{1}{2}(y-\mu)^\top \Sigma^{-1}(y-\mu)\right] \\
&= \exp\left[-\frac{1}{2}\left(\left(\begin{array}{c} y_1 \\ y_2 \end{array}\right) - \left(\begin{array}{c} \mu_1 \\ \mu_2 \end{array}\right)\right)^\top \left(\begin{array}{cc} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{array}\right) \cdots\right] \\
&\propto \exp\left[-\frac{1}{2}\left((y_2-\mu_2)^\top Q_{22}(y_2-\mu_2) + 2(y_2-\mu_2)^\top Q_{21}(y_1-\mu_1)\right)\right]
\end{aligned}
$$

where

$$
\Sigma^{-1} := Q := \left(\begin{array}{cc} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{array}\right)
$$

Proof:

$$
\begin{aligned}
\pi(y_2|y_1) = \frac{\pi(y_1, y_2)}{\pi(y_1)} &\propto \pi(y_1, y_2) \\
&\propto \exp\left[-\frac{1}{2}(y-\mu)^\top \Sigma^{-1}(y-\mu)\right] \\
&= \exp\left[-\frac{1}{2}\left(\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} - \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}\right)^\top \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} \cdots \right] \\
&\propto \exp\left[-\frac{1}{2}\Big((y_2-\mu_2)^\top Q_{22}(y_2-\mu_2) + 2(y_2-\mu_2)^\top Q_{21}(y_1-\mu_1)\Big)\right]
\end{aligned}
$$

where

$$
\Sigma^{-1} := Q := \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix}
$$

So $Y_2|Y_1 = y_1$ is Gaussian.

$$\pi(y_2|y_1) \propto \exp\left(-\frac{1}{2}\left[(y_2-\mu_2)^\top Q_{22}(y_2-\mu_2) + 2(y_2-\mu_2)^\top Q_{21}(y_1-\mu_1)\right]\right)$$

$$\begin{aligned}\pi(y_2|y_1) &\propto \exp\left(-\frac{1}{2}\left[(y_2-\mu_2)^\top Q_{22}(y_2-\mu_2) + 2(y_2-\mu_2)^\top Q_{21}(y_1-\mu_1)\right]\right)\\ &\propto \exp\left(-\frac{1}{2}\left[y_2^\top Q_{22}y_2 - 2y_2^\top\left(Q_{22}\mu_2 + Q_{21}(y_1-\mu_1)\right)\right]\right)\end{aligned}$$

$$
\begin{aligned}
\pi(y_2|y_1) &\propto \exp\left(-\frac{1}{2}\left[(y_2-\mu_2)^\top Q_{22}(y_2-\mu_2) + 2(y_2-\mu_2)^\top Q_{21}(y_1-\mu_1)\right]\right) \\
&\propto \exp\left(-\frac{1}{2}\left[y_2^\top Q_{22} y_2 - 2y_2^\top (Q_{22}\mu_2 + Q_{21}(y_1-\mu_1))\right]\right) \\
&\propto \exp\left(-\frac{1}{2}\left(y_2 - Q_{22}^{-1}(Q_{22}\mu_2 + Q_{21}(y_1-\mu_1))\right)^\top Q_{22}\left(y_2 - \ldots\right)\right)
\end{aligned}
$$

$$\begin{aligned}
\pi(y_2|y_1) &\propto \exp\left(-\frac{1}{2}\left[(y_2-\mu_2)^\top Q_{22}(y_2-\mu_2) + 2(y_2-\mu_2)^\top Q_{21}(y_1-\mu_1)\right]\right) \\
&\propto \exp\left(-\frac{1}{2}\left[y_2^\top Q_{22} y_2 - 2y_2^\top (Q_{22}\mu_2 + Q_{21}(y_1-\mu_1))\right]\right) \\
&\propto \exp\left(-\frac{1}{2}\left(y_2 - Q_{22}^{-1}(Q_{22}\mu_2 + Q_{21}(y_1-\mu_1))\right)^\top Q_{22}\left(y_2 - \ldots\right)\right)
\end{aligned}$$

So

$$Y_2|Y_1 = y_1 \sim N(\mu_2 + Q_{22}^{-1}Q_{21}(y_1-\mu_1), Q_{22}^{-1})$$

$$\begin{aligned}\pi(y_2|y_1) &\propto \exp\left(-\frac{1}{2}\left[(y_2-\mu_2)^\top Q_{22}(y_2-\mu_2) + 2(y_2-\mu_2)^\top Q_{21}(y_1-\mu_1)\right]\right)\\ &\propto \exp\left(-\frac{1}{2}\left[y_2^\top Q_{22} y_2 - 2y_2^\top(Q_{22}\mu_2 + Q_{21}(y_1-\mu_1))\right]\right)\\ &\propto \exp\left(-\frac{1}{2}\left(y_2 - Q_{22}^{-1}(Q_{22}\mu_2 + Q_{21}(y_1-\mu_1))\right)^\top Q_{22}\left(y_2 - \ldots\right)\right)\end{aligned}$$

So

$$Y_2|Y_1 = y_1 \sim N(\mu_2 + Q_{22}^{-1}Q_{21}(y_1-\mu_1), Q_{22}^{-1})$$

The Woodbury matrix identity then gives

$$\begin{aligned} Q_{22}^{-1} &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\\ \text{and} Q_{22}^{-1}Q_{21} &= \Sigma_{21}\Sigma_{11}^{-1}\end{aligned}$$

$$
\begin{aligned}
\pi(y_2|y_1) &\propto \exp\left(-\frac{1}{2}\left[(y_2-\mu_2)^\top Q_{22}(y_2-\mu_2) + 2(y_2-\mu_2)^\top Q_{21}(y_1-\mu_1)\right]\right)\\
&\propto \exp\left(-\frac{1}{2}\left[y_2^\top Q_{22}y_2 - 2y_2^\top(Q_{22}\mu_2 + Q_{21}(y_1-\mu_1))\right]\right)\\
&\propto \exp\left(-\frac{1}{2}\left(y_2 - Q_{22}^{-1}(Q_{22}\mu_2 + Q_{21}(y_1-\mu_1))\right)^\top Q_{22}\left(y_2 - \ldots\right)\right)
\end{aligned}
$$

So

$$Y_2|Y_1 = y_1 \sim N(\mu_2 + Q_{22}^{-1}Q_{21}(y_1-\mu_1), Q_{22}^{-1})$$

The Woodbury matrix identity then gives

$$
\begin{aligned}
Q_{22}^{-1} &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\\
\text{and}\, Q_{22}^{-1}Q_{21} &= \Sigma_{21}\Sigma_{11}^{-1}
\end{aligned}
$$

giving

$$Y_2|Y_1 = y_1 \sim N\left(\mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(y_1-\mu_1), \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\right)$$

Conditional updates of Gaussian processes

If f is a Gaussian process $f(\cdot) \sim GP(m(\cdot), k(\cdot, \cdot))$, then

$$f(x_1), \ldots, f(x_n), f(x) \sim N_{n+1}(\mu, \Sigma)$$

where

Conditional updates of Gaussian processes

If f is a Gaussian process $f(\cdot) \sim GP(m(\cdot), k(\cdot,\cdot))$, then

$$f(x_1), \dots, f(x_n), f(x) \sim N_{n+1}(\mu, \Sigma)$$

where

$$\mu = \left(\begin{array}{c} m(x_1) \\ \vdots \\ m(x_n) \\ \hline m(x) \end{array}\right) = \left(\begin{array}{c} | \\ m_X \\ | \\ \hline m(x) \end{array}\right)$$

and

$$\Sigma = \left(\begin{array}{ccc|c} k(x_1,x_1) & \dots & k(x_1,x_n) & k(x_1,x) \\ \vdots & & \vdots & \vdots \\ k(x_n,x_1) & \dots & k(x_n,x_n) & k(x_n,x) \\ \hline k(x,x_1) & \dots & k(x,x_n) & k(x,x) \end{array}\right) = \left(\begin{array}{c|c} K_{XX} & k_X(x) \\ \hline k_X(x)^\top & k(x,x) \end{array}\right)$$

where $X = \{x_1, \dots, x_n\}$, $[K_{XX}]_{ij} = k(x_i, x_j)$ is the Gram/kernel matrix,

Conditional updates of Gaussian processes

Then

$$f(x)|f(x_1), \ldots, f(x_n) \sim N(\bar{m}(x), \bar{k}(x))$$

Conditional updates of Gaussian processes

Then

$$f(x)|f(x_1), \ldots, f(x_n) \sim N(\bar{m}(x), \bar{k}(x))$$

where

$$\bar{m}(x) = m(x) + k_X(x)^\top {K_{XX}}^{-1}(\mathbf{f} - m_X)$$
$$\bar{k}(x) = k(x, x) - k_X(x)^\top {K_{XX}}^{-1} k_X(x)$$

where

$$\mathbf{f} = (f(x_1), \ldots, f(x_n))^\top$$

Conditional updates of Gaussian processes

Then

$$f(x) | f(x_1), \dots, f(x_n) \sim N(\bar{m}(x), \bar{k}(x))$$

where

$$\bar{m}(x) = m(x) + {k_X(x)}^\top {K_{XX}}^{-1} (\mathbf{f} - m_X)$$
$$\bar{k}(x) = k(x, x) - {k_X(x)}^\top {K_{XX}}^{-1} k_X(x)$$

where

$$\mathbf{f} = (f(x_1), \dots, f(x_n))^\top$$

Cf

$$Y_2 | Y_1 = y_1 \sim N\left(\mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(y_1 - \mu_1), \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\right)$$

Conditional updates of Gaussian processes

Then

$$f(x)|f(x_1),\dots,f(x_n) \sim N(\bar{m}(x), \bar{k}(x))$$

where

$$\begin{aligned}\bar{m}(x) &= m(x) + k_X(x)^\top K_{XX}{}^{-1}(\mathbf{f} - m_X)\\ \bar{k}(x) &= k(x,x) - k_X(x)^\top K_{XX}{}^{-1} k_X(x)\end{aligned}$$

where

$$\mathbf{f} = (f(x_1),\dots,f(x_n))^\top$$

Cf

$$Y_2|Y_1 = y_1 \sim N\left(\mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(y_1 - \mu_1), \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\right)$$

The prior mean and covariance functions m and k, are replaced by the posterior mean and covariance functions $\bar{m}$ and $\bar{k}$,

Summary: Conditional updates of Gaussian processes

If f is a Gaussian process,

$$f(\cdot) \sim GP(m(\cdot), k(\cdot, \cdot))$$

then if we observe f at $x_1, \ldots, x_n$, then

$$f(\cdot)|f(x_1), \ldots, f(x_n) \sim GP(\bar{m}(\cdot), \bar{k}(\cdot, \cdot))$$

where $\bar{m}$ and $\bar{k}$ are the *posterior* mean and covariance functions on the previous slide.

Summary: Conditional updates of Gaussian processes

If f is a Gaussian process,

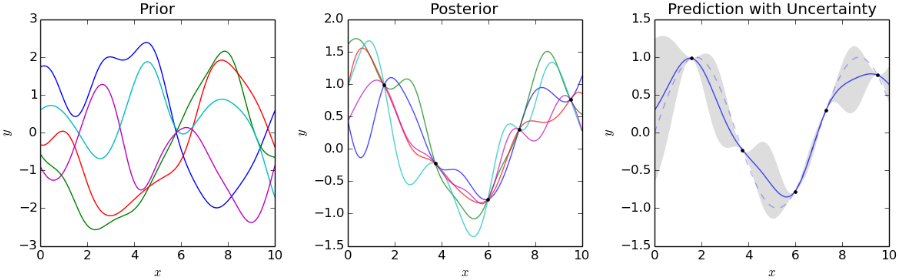
$$f(\cdot) \sim GP(m(\cdot), k(\cdot,\cdot))$$

then if we observe f at $x_1, \ldots, x_n$, then

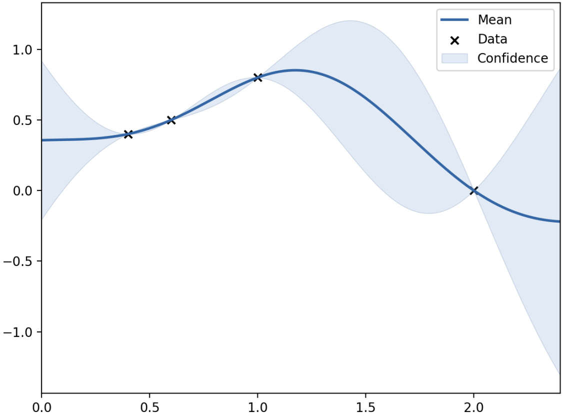
$$f(\cdot)|f(x_1), \ldots, f(x_n) \sim GP(\bar{m}(\cdot), \bar{k}(\cdot,\cdot))$$

where $\bar{m}$ and $\bar{k}$ are the *posterior* mean and covariance functions on the previous slide.

- f is still a GP even though we've observed its value at a number of locations.



No noise/*nugget* - Interpolation



Solid line $\bar{m}(x) = m(x) + k_X(x)^\top K_{XX}^{-1}(\mathbf{f} - m_X)$

Shaded region $\bar{m}(x) \pm 1.96\sqrt{\bar{k}(x)}$

$$\bar{k}(x) = k(x,x) - k_X(x)^\top K_{XX}^{-1} k_X(x)$$

Noisy observations/with nugget - Regression

In practice, we don't usually observe $f(x)$ directly. If we observe

$$y_i = f(x_i) + N(0, \sigma^2)$$

Noisy observations/with nugget - Regression

In practice, we don't usually observe $f(x)$ directly. If we observe

$$
\begin{aligned}
& y_i = f(x_i) + N(0, \sigma^2) \\
\text{then} \quad & y_1, \ldots, y_n, f(x) \sim N_{n+1}(0, \Sigma) \\
\text{where} \quad & \Sigma = \left(\begin{array}{cccc|c}
 & & & & k(x_1, x) \\
 & \color{red}{K_{XX} + \sigma^2 I} & & & k(x_2, x) \\
 & & & & \vdots \\
 & & & & k(x_n, x) \\
\hline
k(x, x_1) & k(x, x_2) & \ldots & k(x, x_n) & k(x, x)
\end{array}\right)
\end{aligned}
$$

Noisy observations/with nugget - Regression

In practice, we don't usually observe $f(x)$ directly. If we observe

$$y_i = f(x_i) + N(0, \sigma^2)$$

then $$y_1, \ldots, y_n, f(x) \sim N_{n+1}(0, \Sigma)$$

where $$\Sigma = \left(\begin{array}{cccc|c} & & & & k(x_1, x) \\ & K_{XX} + \sigma^2 I & & & k(x_2, x) \\ & & & & \vdots \\ & & & & k(x_n, x) \\ \hline k(x, x_1) & k(x, x_2) & \ldots & k(x, x_n) & k(x, x) \end{array}\right)$$

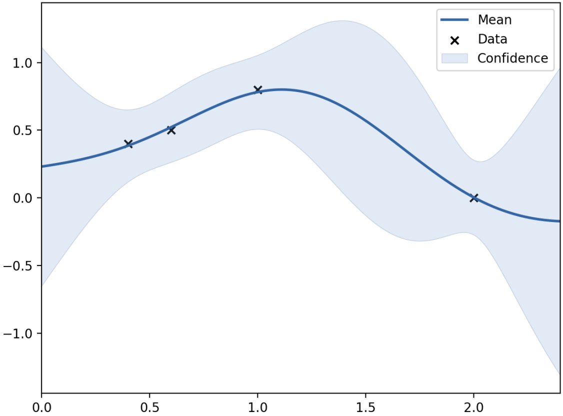
Then

$$f(x) \mid y_1, \ldots, y_n \sim N(\bar{m}(x), \bar{k}(x))$$

where

$$\bar{m}(x) = m(x) + k_X(x)^\top (K_{XX} + \sigma^2 I)^{-1} (\mathbf{y} - m_X)$$
$$\bar{k}(x) = k(x, x) - k_X(x)^\top (K_{XX} + \sigma^2 I)^{-1} k_X(x)$$

Nugget standard deviation $\sigma = 0.1$

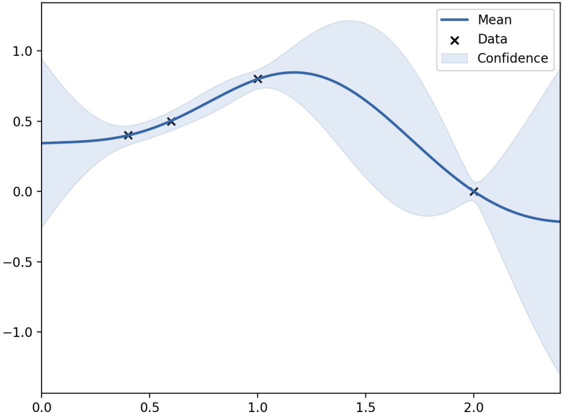


Solid line $\bar{m}(x) = m(x) + k_X(x)^\top K_{XX}^{-1}(\mathbf{y} - m_X)$

Shaded region $\bar{m}(x) \pm 1.96\sqrt{\bar{k}(x)}$

$$\bar{k}(x) = k(x,x) - k_X(x)^\top (K_{XX}^{-1} + \sigma^2 I) k_X(x)$$

Nugget standard deviation $\sigma = 0.025$



Solid line $\bar{m}(x) = m(x) + k_X(x)^\top K_{XX}^{-1}(\mathbf{y} - m_X)$

Shaded region $\bar{m}(x) \pm 1.96\sqrt{\bar{k}(x)}$

$$\bar{k}(x) = k(x,x) - k_X(x)^\top (K_{XX}^{-1} + \sigma^2 I) k_X(x)$$

- If mean is a linear combination of known regressor functions,

  $$m(x) = \beta^\top h(x) \text{ for known } h(x)$$

  and β is given a normal prior distribution (including $\pi(\beta) \propto 1$), then $y(\cdot) \mid D, \beta \sim GP$ and

  $$y(\cdot) \mid D \sim GP$$

  with slightly modified mean and variance formulas.

- If mean is a linear combination of known regressor functions,

  $$m(x) = \beta^\top h(x) \text{ for known } h(x)$$

  and β is given a normal prior distribution (including $\pi(\beta) \propto 1$), then $y(\cdot) \mid D, \beta \sim GP$ and

  $$y(\cdot) \mid D \sim GP$$

  with slightly modified mean and variance formulas.
- If

  $$k(x, x') = \sigma^2 c(x, x')$$

  and we give σ^2 an inverse gamma prior (including $\pi(\sigma^2) \propto 1/\sigma^2$) then $y|D, \sigma^2 \sim GP$ and

  $$y|D \sim \text{t-process}$$

  with $n - p$ degrees of freedom.
  In practice, for reasonable n, this is indistinguishable from a GP.

Why use GPs? Answer 1

The GP class of models is closed under various operations.

Why use GPs? Answer 1

The GP class of models is closed under various operations.

- Closed under Bayesian conditioning, i.e., if we observe

  $$\mathbf{D} = (f(x_1), \ldots, f(x_n))$$

  then

  $$f|D \sim GP$$

  but with updated mean and covariance functions.

Why use GPs? Answer 1

The GP class of models is closed under various operations.

- Closed under Bayesian conditioning, i.e., if we observe

  $$\mathbf{D} = (f(x_1), \ldots, f(x_n))$$

  then

  $$f|D \sim GP$$

  but with updated mean and covariance functions.
- Closed under addition

  $$f_1(\cdot), f_2(\cdot) \sim GP \quad \text{then} \quad (f_1 + f_2)(\cdot) \sim GP$$

Why use GPs? Answer 1

The GP class of models is closed under various operations.

- Closed under Bayesian conditioning, i.e., if we observe

  $$\mathbf{D} = (f(x_1), \dots, f(x_n))$$

  then

  $$f|D \sim GP$$

  but with updated mean and covariance functions.

- Closed under addition

  $$f_1(\cdot), f_2(\cdot) \sim GP \quad \text{then} \quad (f_1 + f_2)(\cdot) \sim GP$$

- Closed under any linear operator. If $f \sim GP(m(\cdot), k(\cdot, \cdot))$, then if $\mathcal{L}$ is a linear operator

  $$\mathcal{L} \circ f \sim GP(\mathcal{L} \circ m, \mathcal{L}^2 \circ k)$$

  e.g. $\frac{df}{dx}$, $\int f(x)dx$, Af are all GPs

Why use GPs? Answer 2: non-parametric/kernel regression

We can also view GPs as a non-parametric extension to linear regression.

- k determines the space of functions that sample paths live in.

Why use GPs? Answer 2: non-parametric/kernel regression

We can also view GPs as a non-parametric extension to linear regression.

- k determines the space of functions that sample paths live in.

Suppose we're given data $\{(x_i, y_i)_{i=1}^n\}$ with $x_i \in \mathbb{R}^p, y_i \in \mathbb{R}$

Why use GPs? Answer 2: non-parametric/kernel regression

We can also view GPs as a non-parametric extension to linear regression.

- k determines the space of functions that sample paths live in.

Suppose we're given data $\{(x_i, y_i)_{i=1}^n\}$ with $x_i \in \mathbb{R}^p, y_i \in \mathbb{R}$

$$\hat{\beta} = \arg\min_{\beta} ||y - X\beta||_2^2 + \sigma^2 ||\beta||_2^2 \quad \text{regularised least squares}$$

$$\text{where} \quad X = \begin{pmatrix} x_1^\top \\ x_2^\top \\ \vdots \\ x_n^\top \end{pmatrix}$$

Why use GPs? Answer 2: non-parametric/kernel regression

We can also view GPs as a non-parametric extension to linear regression.

- k determines the space of functions that sample paths live in.

Suppose we're given data $\{(x_i, y_i)_{i=1}^n\}$ with $x_i \in \mathbb{R}^p, y_i \in \mathbb{R}$

$$\begin{aligned}\hat{\beta} &= \arg\min_\beta ||y - X\beta||_2^2 + \sigma^2||\beta||_2^2 \quad \text{regularised least squares} \\ &= (X^\top X + \sigma^2 I)^{-1} X^\top y \quad \text{usual ridge regression estimator}\end{aligned}$$

$$\text{where} \quad X = \begin{pmatrix} x_1^\top \\ x_2^\top \\ \vdots \\ x_n^\top \end{pmatrix}$$

Why use GPs? Answer 2: non-parametric/kernel regression

We can also view GPs as a non-parametric extension to linear regression.

- k determines the space of functions that sample paths live in.

Suppose we're given data $\{(x_i, y_i)_{i=1}^n\}$ with $x_i \in \mathbb{R}^p, y_i \in \mathbb{R}$

$$\begin{aligned}
\hat{\beta} &= \arg\min_{\beta} ||y - X\beta||_2^2 + \sigma^2 ||\beta||_2^2 \quad \text{regularised least squares} \\
&= (X^\top X + \sigma^2 I)^{-1} X^\top y \quad \text{usual ridge regression estimator} \\
&= X^\top (X X^\top + \sigma^2 I)^{-1} y \quad \text{the dual form}
\end{aligned}$$

$$\text{where} \quad X = \begin{pmatrix} x_1^\top \\ x_2^\top \\ \vdots \\ x_n^\top \end{pmatrix}$$

Why use GPs? Answer 2: non-parametric/kernel regression

We can also view GPs as a non-parametric extension to linear regression.

- k determines the space of functions that sample paths live in.

Suppose we're given data $\{(x_i, y_i)_{i=1}^n\}$ with $x_i \in \mathbb{R}^p, y_i \in \mathbb{R}$

$$\begin{aligned}
\hat{\beta} &= \arg\min_{\beta} ||y - X\beta||_2^2 + \sigma^2||\beta||_2^2 \quad \text{regularised least squares} \\
&= (X^\top X + \sigma^2 I)^{-1} X^\top y \quad \text{usual ridge regression estimator} \\
&= X^\top (XX^\top + \sigma^2 I)^{-1} y \quad \text{the dual form}
\end{aligned}$$

$$\text{as} \qquad (X^\top X + \sigma^2 I) X^\top = X^\top (XX^\top + \sigma^2 I)$$

$$\text{so} \qquad X^\top (XX^\top + \sigma^2 I)^{-1} = (X^\top X + \sigma^2 I)^{-1} X^\top$$

$$\text{where} \qquad X = \begin{pmatrix} x_1^\top \\ x_2^\top \\ \vdots \\ x_n^\top \end{pmatrix}$$

At first the dual form

$$\hat{\beta} = X^\top (XX^\top + \sigma^2 I)^{-1} y$$

looks harder to compute than the usual

$$\hat{\beta} = (X^\top X + \sigma^2 I)^{-1} X^\top y$$

- $X^\top X$ is $p \times p$ $\qquad$ $p =$ number of features/parameters
- $XX^\top$ is $n \times n$ $\qquad$ n is the number of data points

At first the dual form

$$\hat{\beta} = X^\top (XX^\top + \sigma^2 I)^{-1} y$$

looks harder to compute than the usual

$$\hat{\beta} = (X^\top X + \sigma^2 I)^{-1} X^\top y$$

- $X^\top X$ is $p \times p$ $p =$ number of features/parameters
- $XX^\top$ is $n \times n$ n is the number of data points

But the dual form only uses inner products between vectors in $\mathbb{R}^n$

$$XX^\top = \begin{pmatrix} x_1^\top \\ \vdots \\ x_n^\top \end{pmatrix} (x_1 \dots x_n) = \begin{pmatrix} x_1^\top x_1 & \dots & x_1^\top x_n \\ \vdots & & \\ x_n^\top x_1 & \dots & x_n^\top x_n \end{pmatrix}$$

$$= K_{XX} \text{ if } k(x, x') = x^\top x'$$

— This is useful!

Prediction

The best prediction of y at a new location x' is

$$\begin{aligned}\hat{y}' &= x'^\top \hat{\beta} \\ &= x'^\top X^\top (XX^\top + \sigma^2 I)^{-1} y \\ &= k_X(x')^\top (K_{XX} + \sigma^2 I)^{-1} y\end{aligned}$$

where $k_X(x')^\top := (x'^\top x_1, \ldots, x'^\top x_n)$ and $[K_{XX}]_{ij} := x_i^\top x_j$

Prediction

The best prediction of y at a new location x' is

$$\begin{aligned}\hat{y}' &= x'^\top \hat{\beta} \\ &= x'^\top X^\top (XX^\top + \sigma^2 I)^{-1} y \\ &= k_X(x')^\top (K_{XX} + \sigma^2 I)^{-1} y\end{aligned}$$

where $k_X(x')^\top := (x'^\top x_1, \ldots, x'^\top x_n)$ and $[K_{XX}]_{ij} := x_i^\top x_j$
K_{XX} and $k_X(x)$ are kernel matrices:

- every element is an inner product between 2 points: $k(x, x') = x^\top x'$

Prediction

The best prediction of y at a new location x' is

$$\begin{aligned}\hat{y}' &= x'^\top \hat{\beta} \\ &= x'^\top X^\top (XX^\top + \sigma^2 I)^{-1} y \\ &= k_X(x')^\top (K_{XX} + \sigma^2 I)^{-1} y\end{aligned}$$

where $k_X(x')^\top := (x'^\top x_1, \ldots, x'^\top x_n)$ and $[K_{XX}]_{ij} := x_i^\top x_j$
K_{XX} and $k_X(x)$ are kernel matrices:

- every element is an inner product between 2 points: $k(x, x') = x^\top x'$

Note this is the GP conditional mean we derived before with $m(x) = 0$.

$$m(x) = k_X(x)^\top (K_{XX} + \sigma^2 I)^{-1} y$$

- linear regression and GP regression are equivalent when $k(x, x') = x^\top x'$.

Including features I

We can replace x by a feature vector in linear regression, e.g., $\phi(x) = (1 \; x \; x^2)$

It doesn't change the expressions other than the inner product if

$$k(x', x) = x'^\top x$$

is replaced by

$$k(x', x) = \phi(x')^\top \phi(x)$$

Including features I

We can replace x by a feature vector in linear regression, e.g., $\phi(x) = (1\ x\ x^2)$

It doesn't change the expressions other than the inner product if

$$k(x', x) = x'^{\top} x$$

is replaced by

$$k(x', x) = \phi(x')^{\top} \phi(x)$$

Note $k(x', x) = \phi(x')^{\top} \phi(x)$ is a semi-definite function for any choice of $\phi(x)$.

Including features II

For some sets of features, $\phi(x)$, computation of the inner product doesn't require us to evaluate the individual features.

Including features II

For some sets of features, $\phi(x)$, computation of the inner product doesn't require us to evaluate the individual features.

E.g., Consider $\mathcal{X} = \mathbb{R}^2$ and let

$$\phi : \mathbf{x} = (x_1, x_2) \mapsto (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, \sqrt{2}x_1x_2, x_2^2)^\top$$

i.e., linear regression using all the linear and quadratic terms, and first order interactions.

Including features II

For some sets of features, $\phi(x)$, computation of the inner product doesn't require us to evaluate the individual features.

E.g., Consider $\mathcal{X} = \mathbb{R}^2$ and let

$$\phi : \mathbf{x} = (x_1, x_2) \mapsto (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, \sqrt{2}x_1x_2, x_2^2)^\top$$

i.e., linear regression using all the linear and quadratic terms, and first order interactions.
Then

$$\begin{aligned}
k(\mathbf{x}, \mathbf{z}) &= \phi(x)^\top \phi(z) \\
&= (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, \sqrt{2}x_1x_2, x_2^2)(1, \sqrt{2}z_1, \sqrt{2}z_2, z_1^2, \sqrt{2}z_1z_2, z_2^2)^\top \\
&= (1 + (x_1, x_2)(z_1, z_2)^\top)^2 \\
&= (1 + \mathbf{x}^\top \mathbf{z})^2
\end{aligned}$$

Including features II

For some sets of features, $\phi(x)$, computation of the inner product doesn't require us to evaluate the individual features.

E.g., Consider $\mathcal{X} = \mathbb{R}^2$ and let

$$\phi : \mathbf{x} = (x_1, x_2) \mapsto (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, \sqrt{2}x_1x_2, x_2^2)^\top$$

i.e., linear regression using all the linear and quadratic terms, and first order interactions.
Then

$$\begin{aligned} k(\mathbf{x}, \mathbf{z}) &= \phi(x)^\top \phi(z) \\ &= (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, \sqrt{2}x_1x_2, x_2^2)(1, \sqrt{2}z_1, \sqrt{2}z_2, z_1^2, \sqrt{2}z_1z_2, z_2^2)^\top \\ &= (1 + (x_1, x_2)(z_1, z_2)^\top)^2 \\ &= (1 + \mathbf{x}^\top \mathbf{z})^2 \end{aligned}$$

To evaluate $k(\mathbf{x}, \mathbf{z})$ we didn't need to explicitly compute the feature vector $\phi(\mathbf{x})$

Including features III

To evaluate $k(\mathbf{x}, \mathbf{z})$ we didn't need to explicitly compute the feature vectors $\phi(\mathbf{x}), \phi(\mathbf{z}) \in \mathbb{R}^6$

The same idea works with much larger feature vectors, sometimes even when $\phi(\mathbf{x}) \in \mathbb{R}^\infty$

Including features III

To evaluate $k(\mathbf{x}, \mathbf{z})$ we didn't need to explicitly compute the feature vectors $\phi(\mathbf{x}), \phi(\mathbf{z}) \in \mathbb{R}^6$

The same idea works with much larger feature vectors, sometimes even when $\phi(\mathbf{x}) \in \mathbb{R}^\infty$

Theorem: A function

$$k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$$

is positive semi-definite (and thus a valid covariance function) if and only if we can write

$$k(x, x') = \phi(x)^\top \phi(x')$$

for some (possibly infinite dimensional) feature vector $\phi(x)$.

Including features III

To evaluate $k(\mathbf{x}, \mathbf{z})$ we didn't need to explicitly compute the feature vectors $\phi(\mathbf{x}), \phi(\mathbf{z}) \in \mathbb{R}^6$

The same idea works with much larger feature vectors, sometimes even when $\phi(\mathbf{x}) \in \mathbb{R}^\infty$

Theorem: A function

$$k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$$

is positive semi-definite (and thus a valid covariance function) if and only if we can write

$$k(x, x') = \phi(x)^\top \phi(x')$$

for some (possibly infinite dimensional) feature vector $\phi(x)$.

So GP regression with k can be thought of as linear regression with $\phi(x)$.

Example: If $\mathcal{X} = \mathbb{R}$, set $\phi_c(x) = e^{-\frac{(x-c_0)^2}{2\lambda^2}}$ and

$$\phi^N(x) = \frac{1}{\sqrt{N}}(\phi_{c_0}(x), \ldots, \phi_{c_N}(x))$$

with $c_0 = -\log N, c_N = \log N, c_{i+1} - c_i = 2\frac{\log N}{N}$.

Example: If $\mathcal{X} = \mathbb{R}$, set $\phi_c(x) = e^{-\frac{(x-c_0)^2}{2\lambda^2}}$ and

$$\phi^N(x) = \frac{1}{\sqrt{N}}(\phi_{c_0}(x), \ldots, \phi_{c_N}(x))$$

with $c_0 = -\log N, c_N = \log N, c_{i+1} - c_i = 2\frac{\log N}{N}$. Then

$$\begin{aligned} \phi^N(x)^\top \phi^N(x') &= \frac{1}{N}\sum_{i=0}^{N} \phi_{c_i}(x)\phi_{c_i}(x') \\ &= \int_{-\log N}^{\log N} \phi_c(x)\phi(x')\mathrm{d}c \to \exp\left(-\frac{(x-x')^2}{2\lambda^2}\right) \text{ as } N \to \infty \end{aligned}$$

So the RBF kernel arises when we do linear regression with an infinite feature vector containing Gaussian bumps

Example: If $\mathcal{X} = \mathbb{R}$, set $\phi_c(x) = e^{-\frac{(x-c_0)^2}{2\lambda^2}}$ and

$$\phi^N(x) = \frac{1}{\sqrt{N}}(\phi_{c_0}(x), \ldots, \phi_{c_N}(x))$$

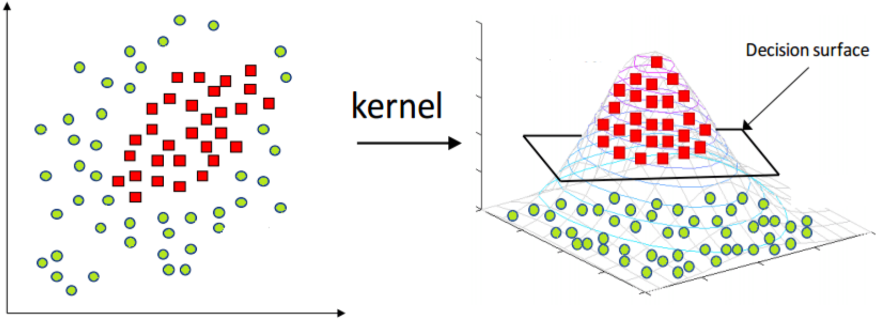
with $c_0 = -\log N, c_N = \log N, c_{i+1} - c_i = 2\frac{\log N}{N}$. Then

$$\begin{aligned}\phi^N(x)^\top \phi^N(x') &= \frac{1}{N}\sum_{i=0}^{N} \phi_{c_i}(x)\phi_{c_i}(x') \\ &= \int_{-\log N}^{\log N} \phi_c(x)\phi(x')\mathrm{d}c \to \exp\left(-\frac{(x-x')^2}{2\lambda^2}\right) \text{ as } N \to \infty\end{aligned}$$

So the RBF kernel arises when we do linear regression with an infinite feature vector containing Gaussian bumps
We can use an infinite dimensional feature vector $\phi(x)$, and because linear regression can be done solely in terms of inner-products (inverting a $n \times n$ matrix in the dual form) we never need evaluate the feature vector, only the kernel.

Kernel trick:

lift x into feature space by replacing inner products $x^\top x'$ by $k(x, x')$



Kernel regression (see Kanagawa et al. 2019)

Kernel regression and GP regression are closely related.

Kernel regression (see Kanagawa et al. 2019)

Kernel regression and GP regression are closely related.
Consider the space of functions

$$\mathcal{H}_k = \overline{\mathsf{span}}\{k(\cdot, x) : x \in \mathcal{X}\}$$

ie functions of the form $\sum_{i=1}^n \alpha_i k(x, x_i)$ with inner product

$$\langle \sum a_i k(\cdot, x_i), \sum b_i k(\cdot, y_i) \rangle = \sum_{ij} a_i b_j k(x_i, y_j)$$

Kernel regression (see Kanagawa et al. 2019)

Kernel regression and GP regression are closely related.

Consider the space of functions

$$\mathcal{H}_k = \overline{\text{span}}\{k(\cdot, x) : x \in \mathcal{X}\}$$

ie functions of the form $\sum_{i=1}^{n} \alpha_i k(x, x_i)$ with inner product

$$\langle \sum a_i k(\cdot, x_i), \sum b_i k(\cdot, y_i) \rangle = \sum_{ij} a_i b_j k(x_i, y_j)$$

This is the reproducing kernel Hilbert space (RKHS) associated with k.

Kernel regression (see Kanagawa et al. 2019)

Kernel regression and GP regression are closely related.
Consider the space of functions

$$\mathcal{H}_k = \overline{\text{span}}\{k(\cdot, x) : x \in \mathcal{X}\}$$

ie functions of the form $\sum_{i=1}^{n} \alpha_i k(x, x_i)$ with inner product

$$\langle \sum a_i k(\cdot, x_i), \sum b_i k(\cdot, y_i) \rangle = \sum_{ij} a_i b_j k(x_i, y_j)$$

This is the reproducing kernel Hilbert space (RKHS) associated with k.
Kernel ridge regression chooses $f \in \mathcal{H}_k$ to minimise

$$L(f) = \sum_i (f(x_i) - y_i)^2 + \sigma^2 ||f||^2_{\mathcal{H}_k}$$

Kernel regression (see Kanagawa et al. 2019)

Kernel regression and GP regression are closely related.
Consider the space of functions

$$\mathcal{H}_k = \overline{\text{span}}\{k(\cdot, x) : x \in \mathcal{X}\}$$

ie functions of the form $\sum_{i=1}^n \alpha_i k(x, x_i)$ with inner product

$$\langle \sum a_i k(\cdot, x_i), \sum b_i k(\cdot, y_i) \rangle = \sum_{ij} a_i b_j k(x_i, y_j)$$

This is the reproducing kernel Hilbert space (RKHS) associated with k.
Kernel ridge regression chooses $f \in \mathcal{H}_k$ to minimise

$$L(f) = \sum_i (f(x_i) - y_i)^2 + \sigma^2 ||f||^2_{\mathcal{H}_k}$$

We can show that

$$\bar{m}(x) = \arg\min_{f \in \mathcal{H}_k} L(f)$$

where $\bar{m}(x)$ is the same as the posterior mean when we assume $y_i = f(x_i) + N(0, \sigma^2)$ and $f(\cdot) \sim GP(0, k(\cdot, \cdot))$

Kernel regression (see Kanagawa et al. 2019)

Kernel regression and GP regression are closely related.
Consider the space of functions

$$\mathcal{H}_k = \overline{\text{span}}\{k(\cdot, x) : x \in \mathcal{X}\}$$

ie functions of the form $\sum_{i=1}^n \alpha_i k(x, x_i)$ with inner product

$$\langle \sum a_i k(\cdot, x_i), \sum b_i k(\cdot, y_i) \rangle = \sum_{ij} a_i b_j k(x_i, y_j)$$

This is the reproducing kernel Hilbert space (RKHS) associated with k.
Kernel ridge regression chooses $f \in \mathcal{H}_k$ to minimise

$$L(f) = \sum_i (f(x_i) - y_i)^2 + \sigma^2 ||f||^2_{\mathcal{H}_k}$$

We can show that

$$\bar{m}(x) = \arg \min_{f \in \mathcal{H}_k} L(f)$$

where $\bar{m}(x)$ is the same as the posterior mean when we assume
$y_i = f(x_i) + N(0, \sigma^2)$ and $f(\cdot) \sim GP(0, k(\cdot, \cdot))$
Note: $\bar{m}(\cdot) \in \mathcal{H}_k$ but samples from a GP live in a slightly larger RKHS.

TL;DR

Functions live in function spaces (vector spaces with inner products). There are lots of different function spaces: the GP kernel implicitly determines which particular (RKHS) space we work with - our hypothesis space.

- Generally, we don't think too hard about this space, we just choose a kernel and attempt to validate it empirically.

[3] and can be dense in some sets of continuous bounded functions

TL;DR

Functions live in function spaces (vector spaces with inner products). There are lots of different function spaces: the GP kernel implicitly determines which particular (RKHS) space we work with - our hypothesis space.

- Generally, we don't think too hard about this space, we just choose a kernel and attempt to validate it empirically.

Although reality may not lie in the RKHS defined by k, this space is much richer than any parametric regression model[3],

- thus is more likely to contain an element close to the true functional form than any class of models that contains only a finite number of features.

This is the motivation for non-parametric methods.

[3] and can be dense in some sets of continuous bounded functions

Why use GPs? Answer 3: Naturalness of GP framework

Why use **Gaussian** processes as non-parametric models?

Why use GPs? Answer 3: Naturalness of GP framework

Why use **Gaussian** processes as non-parametric models?

If we only knew the expectation and variance of some random variables, X and Y, then how should we best do statistics?

Why use GPs? Answer 3: Naturalness of GP framework

Why use **Gaussian** processes as non-parametric models?

If we only knew the expectation and variance of some random variables, X and Y, then how should we best do statistics?

It has been shown, using coherency arguments, or geometric arguments, or..., that the best second-order inference we can do is to update our beliefs about X given Y using

$$\mathbb{E}(X|Y) = \mathbb{E}(X) + \mathbb{C}\text{ov}(X, Y)\mathbb{V}\text{ar}(Y)^{-1}(Y - \mathbb{E}(Y))$$

i.e., exactly the Gaussian process update for the posterior mean.

So GPs are in some sense second-order optimal.

Kriging

Kriging

Suppose $Y(x)$ is a (second order stationary) stochastic process with

$$\mathbb{E}Y(x) = \mu \;\forall\; x$$
$$\mathbb{C}\mathrm{ov}(Y(x), Y(x')) = k(x - x') \;\forall\; x, x'$$

NB we're not assuming Y has a Gaussian distribution.

Kriging

Suppose $Y(x)$ is a (second order stationary) stochastic process with

$$\mathbb{E}Y(x) = \mu \;\forall\; x$$
$$\mathbb{C}\text{ov}(Y(x), Y(x')) = k(x - x') \;\forall\; x, x'$$

NB we're not assuming Y has a Gaussian distribution.

If someone tells you $\mathbf{y} = (Y(x_1), \ldots, Y(x_n))^\top$, how would you predict $Y(x)$?

Kriging

Suppose $Y(x)$ is a (second order stationary) stochastic process with

$$\mathbb{E}Y(x) = \mu \; \forall \; x$$
$$\mathbb{C}\text{ov}(Y(x), Y(x')) = k(x - x') \; \forall \; x, x'$$

NB we're not assuming Y has a Gaussian distribution.

If someone tells you $\mathbf{y} = (Y(x_1), \ldots, Y(x_n))^\top$, how would you predict $Y(x)$?
One option is to find the best linear unbiased predictor (BLUP) of $Y(x)$.

Best Linear Unbiased Predictors (BLUP)

Consider the linear estimator

$$\hat{Y}(x) = c + \sum w_i Y(x_i) = c + \mathbf{w}^\top \mathbf{y}$$

Best Linear Unbiased Predictors (BLUP)

Consider the linear estimator

$$\hat{Y}(x) = c + \sum w_i Y(x_i) = c + \mathbf{w}^\top \mathbf{y}$$

If we require $\hat{Y}(x)$ to be unbiased,

$$\begin{aligned} \mu &= \mathbb{E}\hat{Y}(x) \\ &= \mathbb{E}(c + \mathbf{w}^\top \mathbf{y}) \\ &= c + \mathbf{w}^\top \boldsymbol{\mu} \end{aligned}$$

where $\boldsymbol{\mu} = (\mu, \ldots, \mu)^\top$.

Best Linear Unbiased Predictors (BLUP)

Consider the linear estimator

$$\hat{Y}(x) = c + \sum w_i Y(x_i) = c + \mathbf{w}^\top \mathbf{y}$$

If we require $\hat{Y}(x)$ to be unbiased,

$$\begin{aligned} \mu &= \mathbb{E}\hat{Y}(x) \\ &= \mathbb{E}(c + \mathbf{w}^\top \mathbf{y}) \\ &= c + \mathbf{w}^\top \boldsymbol{\mu} \end{aligned}$$

where $\boldsymbol{\mu} = (\mu, \ldots, \mu)^\top$.

Thus $c = \mu - \mathbf{w}^\top \boldsymbol{\mu}$ and we must have

$$\hat{Y}(x) = \mu + \mathbf{w}^\top (\mathbf{y} - \boldsymbol{\mu})$$

Best Linear Unbiased Predictors (BLUP) - II

The best linear unbiased predictor minimises the mean square error

$$\begin{aligned} MSE(\hat{Y}(x)) &= \mathbb{E}((\hat{Y}(x) - Y(x))^2) \\ &= \mathbb{E}\left((\mathbf{w}^\top(\mathbf{y} - \boldsymbol{\mu}) + (\mu - Y(x))^2\right) \\ &= \mathbf{w}^\top \mathbb{V}\mathrm{ar}(\mathbf{y})\mathbf{w} + \mathbb{V}\mathrm{ar}(Y(x)) - 2\mathbf{w}^\top \mathbb{C}\mathrm{ov}(\mathbf{y}, Y(x)) \\ &= \mathbf{w}^\top K_{XX}\mathbf{w} + k(0) - 2\mathbf{w}^\top \mathbf{k}_X(x) \end{aligned}$$

Best Linear Unbiased Predictors (BLUP) - II

The best linear unbiased predictor minimises the mean square error

$$
\begin{aligned}
MSE(\hat{Y}(x)) &= \mathbb{E}((\hat{Y}(x) - Y(x))^2) \\
&= \mathbb{E}\left((\mathbf{w}^\top(\mathbf{y} - \boldsymbol{\mu}) + (\mu - Y(x))^2\right) \\
&= \mathbf{w}^\top \mathbb{V}\text{ar}(\mathbf{y})\mathbf{w} + \mathbb{V}\text{ar}(Y(x)) - 2\mathbf{w}^\top \mathbb{C}\text{ov}(\mathbf{y}, Y(x)) \\
&= \mathbf{w}^\top K_{XX}\mathbf{w} + k(0) - 2\mathbf{w}^\top \mathbf{k}_X(x)
\end{aligned}
$$

If we differentiate wrt w and set the gradient equal to zero, we find

$$0 = 2K_{XX}\mathbf{w} - 2\mathbf{k}_X(x)$$

Best Linear Unbiased Predictors (BLUP) - II

The best linear unbiased predictor minimises the mean square error

$$\begin{aligned} MSE(\hat{Y}(x)) &= \mathbb{E}((\hat{Y}(x) - Y(x))^2) \\ &= \mathbb{E}\left((\mathbf{w}^\top(\mathbf{y} - \boldsymbol{\mu}) + (\mu - Y(x))^2\right) \\ &= \mathbf{w}^\top \mathbb{V}\mathrm{ar}(\mathbf{y})\mathbf{w} + \mathbb{V}\mathrm{ar}(Y(x)) - 2\mathbf{w}^\top \mathbb{C}\mathrm{ov}(\mathbf{y}, Y(x)) \\ &= \mathbf{w}^\top K_{XX}\mathbf{w} + k(0) - 2\mathbf{w}^\top \mathbf{k}_X(x) \end{aligned}$$

If we differentiate wrt w and set the gradient equal to zero, we find

$$0 = 2K_{XX}\mathbf{w} - 2\mathbf{k}_X(x)$$

and thus

$$\hat{Y}(x) = \mu + \mathbf{k}_X(x)^\top K_{XX}^{-1}(\mathbf{y} - \mu)$$

as before.

So the Gaussian process posterior mean is optimal (i.e. is the BLUP) even if we don't assume Gaussianity.

Why use GPs? Answer 4: Uncertainty estimates

GP predictions consist of two parts:

- point estimate $m(x) = \mathbb{E}f(x)$
- uncertainty about the estimate $k(x, x) = \mathbb{V}\mathrm{ar}f(x)$

Quantification of prediction uncertainty (cf NNs) is one of the main advantages of GPs.

Why use GPs? Answer 4: Uncertainty estimates

GP predictions consist of two parts:

- point estimate $m(x) = \mathbb{E}f(x)$
- uncertainty about the estimate $k(x, x) = \mathbb{V}\mathrm{ar} f(x)$

Quantification of prediction uncertainty (cf NNs) is one of the main advantages of GPs.

It is important to check both aspects.

Why use GPs? Answer 4: Uncertainty estimates

GP predictions consist of two parts:

- point estimate $m(x) = \mathbb{E}f(x)$
- uncertainty about the estimate $k(x, x) = \mathbb{V}\mathrm{ar} f(x)$

Quantification of prediction uncertainty (cf NNs) is one of the main advantages of GPs.

It is important to check both aspects.

Warning: the uncertainty estimates from a GP can be flawed. Note that given data $D = \{X, y\}$

$$\mathbb{V}\mathrm{ar}(f(x)|X, y) = k(x, x) - k_X(x) K_{XX}^{-1} k_X(x)$$

The posterior variance of $f(x)$ does not directly depend upon y!

Variance estimates are particularly sensitive to the hyper-parameter estimates.

Difficulties of using GPs

If we know the covariance function, GPs work great!

Difficulties of using GPs

If we know the covariance function, GPs work great!
Unfortunately, we don't usually know this.

- We pick a covariance function from a small set, based usually on differentiability considerations.

Difficulties of using GPs

If we know the covariance function, GPs work great!
Unfortunately, we don't usually know this.

- We pick a covariance function from a small set, based usually on differentiability considerations.
- Possibly try a few (plus combinations of a few), and use empirical evaluation to make a good choice.

Difficulties of using GPs

If we know the covariance function, GPs work great!
Unfortunately, we don't usually know this.

- We pick a covariance function from a small set, based usually on differentiability considerations.
- Possibly try a few (plus combinations of a few), and use empirical evaluation to make a good choice.
- Covariance functions often contain hyper-parameters. E.g
    - RBF kernel

$$k(x, x') = \sigma^2 \exp\left(-\frac{1}{2}\frac{(x-x')^2}{\lambda^2}\right)$$

Estimate these using your favourite statistical procedure

Difficulties of using GPs

Gelman *et al.* 2017, Bachoc 2020

Assuming a GP model for your data imposes a complex structure on the data.

Difficulties of using GPs

Gelman *et al.* 2017, Bachoc 2020

Assuming a GP model for your data imposes a complex structure on the data.

The number of parameters in a GP is essentially infinite, and so they are not always identified even asymptotically.

Difficulties of using GPs

Gelman *et al.* 2017, Bachoc 2020

Assuming a GP model for your data imposes a complex structure on the data.

The number of parameters in a GP is essentially infinite, and so they are not always identified even asymptotically.

The posterior can often concentrate on a sub-manifold of parameter space (rather than a single point), and the projection of the prior on this space continues to impact the posterior even as more and more data are collected.

Difficulties of using GPs

Gelman *et al.* 2017, Bachoc 2020

Assuming a GP model for your data imposes a complex structure on the data.

The number of parameters in a GP is essentially infinite, and so they are not always identified even asymptotically.

The posterior can often concentrate on a sub-manifold of parameter space (rather than a single point), and the projection of the prior on this space continues to impact the posterior even as more and more data are collected.

E.g. consider a zero mean GP on $[0, 1]$ with covariance function

$$k(x, x') = \sigma^2 \exp(-\kappa^2|x - x|)$$

We can consistently estimate $\sigma^2\kappa$, but not σ^2 or κ, even as $n \to \infty$.

Problems with hyper-parameter optimization

The likelihood surface that is maximized in hyper-parameter estimation is often flat and multi-modal,

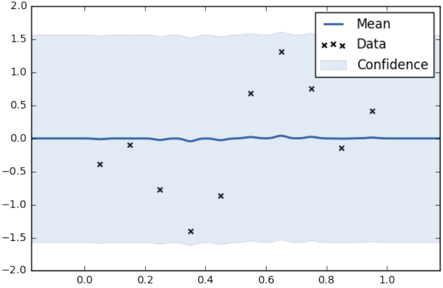
- optimizers can sometimes fail to converge, or get stuck in local-maxima.

Problems with hyper-parameter optimization

The likelihood surface that is maximized in hyper-parameter estimation is often flat and multi-modal,

- optimizers can sometimes fail to converge, or get stuck in local-maxima.

In practice, it is not uncommon to optimize hyper parameters and find solutions such as

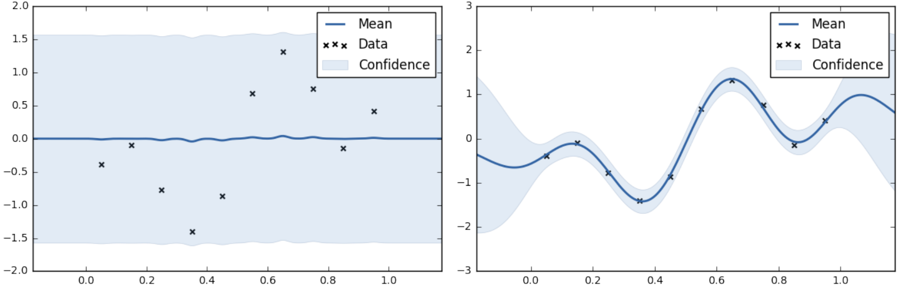


Problems with hyper-parameter optimization

The likelihood surface that is maximized in hyper-parameter estimation is often flat and multi-modal,

- optimizers can sometimes fail to converge, or get stuck in local-maxima.

In practice, it is not uncommon to optimize hyper parameters and find solutions such as



Work around these problems by running the optimizer multiple times from random start points, using prior distributions, constraining or fixing hyper-parameters, or adding white noise.

Computational cost

The computational cost of GP training is $O(n^3)$ with $O(n^2)$ memory requirements

Computational cost

The computational cost of GP training is $O(n^3)$ with $O(n^2)$ memory requirements
There are many ways to reduce these costs. E.g. consider basis expansions and using the primal form for linear regression.

Computational cost

The computational cost of GP training is $O(n^3)$ with $O(n^2)$ memory requirements

There are many ways to reduce these costs. E.g. consider basis expansions and using the primal form for linear regression.

Suppose

$$k(x, x') = \sum_{i=1}^{m} \phi_i(x)\phi_i(x') = \boldsymbol{\phi}(x)^\top \boldsymbol{\phi}(x')$$

Computational cost

The computational cost of GP training is $O(n^3)$ with $O(n^2)$ memory requirements

There are many ways to reduce these costs. E.g. consider basis expansions and using the primal form for linear regression.

Suppose

$$k(x, x') = \sum_{i=1}^{m} \phi_i(x)\phi_i(x') = \boldsymbol{\phi}(x)^\top \boldsymbol{\phi}(x')$$

Then GP regression is equivalent to linear regression with covariates $\phi(x)$

- Dual form for regression coefficients costs $O(n^3)$, but primal solution only costs $O(m^3)$

In practice we may use a basis expansion with $m << n$ such that

$$k(x, x') \approx \sum_{i=1}^{m} \phi_i(x)\phi_i(x')$$

Choice of basis

There are many choices of basis. Two examples:

- **Mercer basis**: Consider the map

$$T_k(f)(\cdot) = \int_{\mathcal{X}} k(x, \cdot) f(x) \mathrm{d}x$$

  Consider the eigenfunctions of this map, i.e., $\phi : \mathcal{X} \mapsto \mathbb{R}$ s.t. $T_k(\phi)(\cdot) = \lambda\phi(\cdot)$. Mercer's theorem says that

$$k(x, x') = \sum_{i=1}^{\infty} \lambda_i \phi_i(x) \phi_i(x')$$

Choice of basis

There are many choices of basis. Two examples:

- **Mercer basis**: Consider the map

$$T_k(f)(\cdot) = \int_{\mathcal{X}} k(x, \cdot) f(x) \mathrm{d}x$$

Consider the eigenfunctions of this map, i.e., $\phi : \mathcal{X} \mapsto \mathbb{R}$ s.t. $T_k(\phi)(\cdot) = \lambda \phi(\cdot)$. Mercer's theorem says that

$$k(x, x') = \sum_{i=1}^{\infty} \lambda_i \phi_i(x) \phi_i(x')$$

The Karhunen-Loeve thm says we can write $f(\cdot) \sim GP(0, k(\cdot, \cdot))$ as

$$f(x) = \sum_{i=1}^{\infty} Z_i \sqrt{\lambda_i} \phi_i(x) \quad \text{where } Z_i \overset{iid}{\sim} N(0, 1)$$

Choice of basis

There are many choices of basis. Two examples:

- **Mercer basis**: Consider the map

$$T_k(f)(\cdot) = \int_{\mathcal{X}} k(x,\cdot) f(x) \mathrm{d}x$$

Consider the eigenfunctions of this map, i.e., $\phi : \mathcal{X} \mapsto \mathbb{R}$ s.t. $T_k(\phi)(\cdot) = \lambda\phi(\cdot)$. Mercer's theorem says that

$$k(x,x') = \sum_{i=1}^{\infty} \lambda_i \phi_i(x) \phi_i(x')$$

The Karhunen-Loeve thm says we can write $f(\cdot) \sim GP(0, k(\cdot,\cdot))$ as

$$f(x) = \sum_{i=1}^{\infty} Z_i \sqrt{\lambda_i} \phi_i(x) \quad \text{where } Z_i \overset{iid}{\sim} N(0,1)$$

We can approximate the process (& reduce cost to $O(m^3)$) by truncating the sum

$$f(x) = \sum_{i=1}^{m} Z_i \sqrt{\lambda_i} \phi_i(x)$$

The Mercer/KL basis minimizes the mean square truncation error.

Choice of basis

There are many choices of basis. Two examples:

- **Random Fourier features (Rahimi and Recht 2007)** :
  Bochner's theorem says that a stationary kernel can be represented as a Fourier transform of some distribution p

$$k(x - x') = \int \exp(iw^\top(x - x'))p(w)dw = \mathbb{E}_{w \sim p} \exp(iw^\top(x - x'))$$

Choice of basis

There are many choices of basis. Two examples:

- **Random Fourier features (Rahimi and Recht 2007)** :
  Bochner's theorem says that a stationary kernel can be represented as a Fourier transform of some distribution p

$$
\begin{aligned}
k(x-x') &= \int \exp(iw^\top(x-x'))p(w)dw = \mathbb{E}_{w\sim p}\exp(iw^\top(x-x')) \\
&\approx \frac{1}{m}\sum_{i=1}^{m}(\cos(w_i^\top x),\ \sin(w_i^\top x))\begin{pmatrix}\cos(w_i^\top x') \\ \sin(w_i^\top x')\end{pmatrix} \text{ if } w_i \sim p(\cdot)
\end{aligned}
$$

  by using Euler's identity and discarding the imaginary part.
  This again reduces the complexity to $O(m^3)$.

Conclusions

- GPs are now ubiquitous in statistics/ML.
- Popularity stems from
  - Naturalness of the framework
  - Mathematical tractability
  - Empirical success

Conclusions

- GPs are now ubiquitous in statistics/ML.
- Popularity stems from
  - Naturalness of the framework
  - Mathematical tractability
  - Empirical success

Thank you for listening!

References

- Rasmussen and Williams. *Gaussian processes for machine learning*. MIT press, 2006.
- Stein. *Interpolation of Spatial Data: Some Theory for Kriging*. Springer, 1999
- Kanagawa, Hennig, Sejdinovic, and Sriperumbudur. *Gaussian processes and kernel methods: A review on connections and equivalences.*. arXiv:1807.02582 2018.