# C24 Advanced Probability Theory

Michael A. Osborne

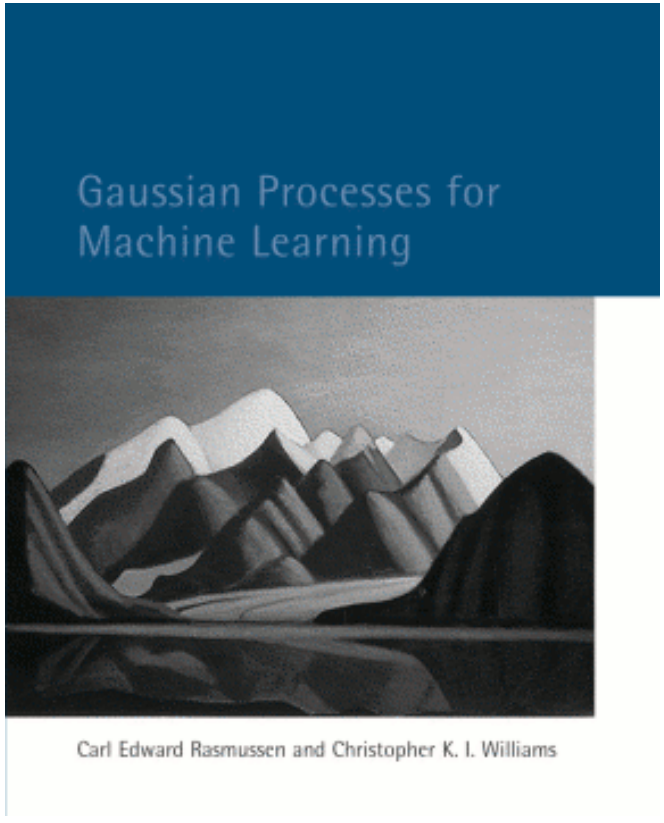
mosb@robots.ox.ac.uk
www.robots.ox.ac.uk/∼mosb/c24

Michaelmas 2014

# Topic 2: Gaussian processes

# The canonical text for Gaussian processes is Rasmussen and Williams (2006).

Available free online (legally)!



# Let's return to predicting in the presence of parameters $\theta$.

$$p(f_\star \mid \mathcal{D}) = \int p(f_\star \mid \mathcal{D}, \theta)\, p(\theta \mid \mathcal{D})\, \mathrm{d}\theta$$

1. $p(f_\star \mid \mathcal{D})$ is the posterior for $f_\star$; this is our goal.
2. $p(f_\star \mid \mathcal{D}, \theta)$ are the predictions given $\theta$.
3. $p(\theta \mid \mathcal{D})$ is the posterior for $\theta$. Using Bayes' rule,

$$p(\theta \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \theta)\, p(\theta)}{p(\mathcal{D})},$$

That is, the parameter posterior is proportional to its likelihood and prior.

The predictions are averaged over, weighted by the parameter posterior.

# Consider a linear model with a Gaussian prior.

That is, if we take a linear model for a function $f(x)$,

$$p(f_\star \mid \mathcal{D}, \theta) = p(f_\star \mid \theta) = \delta(f_\star - \theta\, x_\star),$$

where $f_\star = f(x_\star)$, and a Gaussian prior for the parameter,

$$p(\theta) = \mathcal{N}(\theta; \mu, \nu^2),$$

and if we assume our data $\mathcal{D}$ are $N$ iid noisy observations of the function, $\mathcal{D} = \left\{(x_i, z_i); i \in [1, \ldots, N]\right\}$, we have

$$p(\mathcal{D} \mid \theta) = \prod_{i=1}^{N} \mathcal{N}(z_i; \theta\, x_i, \sigma^2).$$

# We can actually compute the posterior in the linear-Gaussian case.

$$p(f_\star \mid \mathcal{D}) = \int p(f_\star \mid \mathcal{D}, \theta) \frac{p(\theta)\, p(\mathcal{D} \mid \theta)}{p(\mathcal{D})} \,\mathrm{d}\theta$$

$$= \int \delta(f_\star - \theta\, x_\star) \frac{\mathcal{N}(\theta; \mu, \nu^2) \prod_{i=1}^N \mathcal{N}(z_i; \theta\, x_i, \sigma^2)}{A} \,\mathrm{d}\theta$$

$$= \int \delta(f_\star - \theta\, x_\star) \frac{\mathcal{N}(\theta; \mu, \nu^2) \prod_{i=1}^N \mathcal{N}(z_i/x_i; \theta, \sigma^2/x_i^2)}{A'} \,\mathrm{d}\theta$$

Remember that $A$ and $A'$ are constants wrt $\theta$, and

$$\mathcal{N}(z; \alpha, \beta^2) = \frac{1}{\sqrt{2\pi\beta^2}} \exp -\frac{1}{2}\left(\frac{z-\alpha}{\beta}\right)^2$$

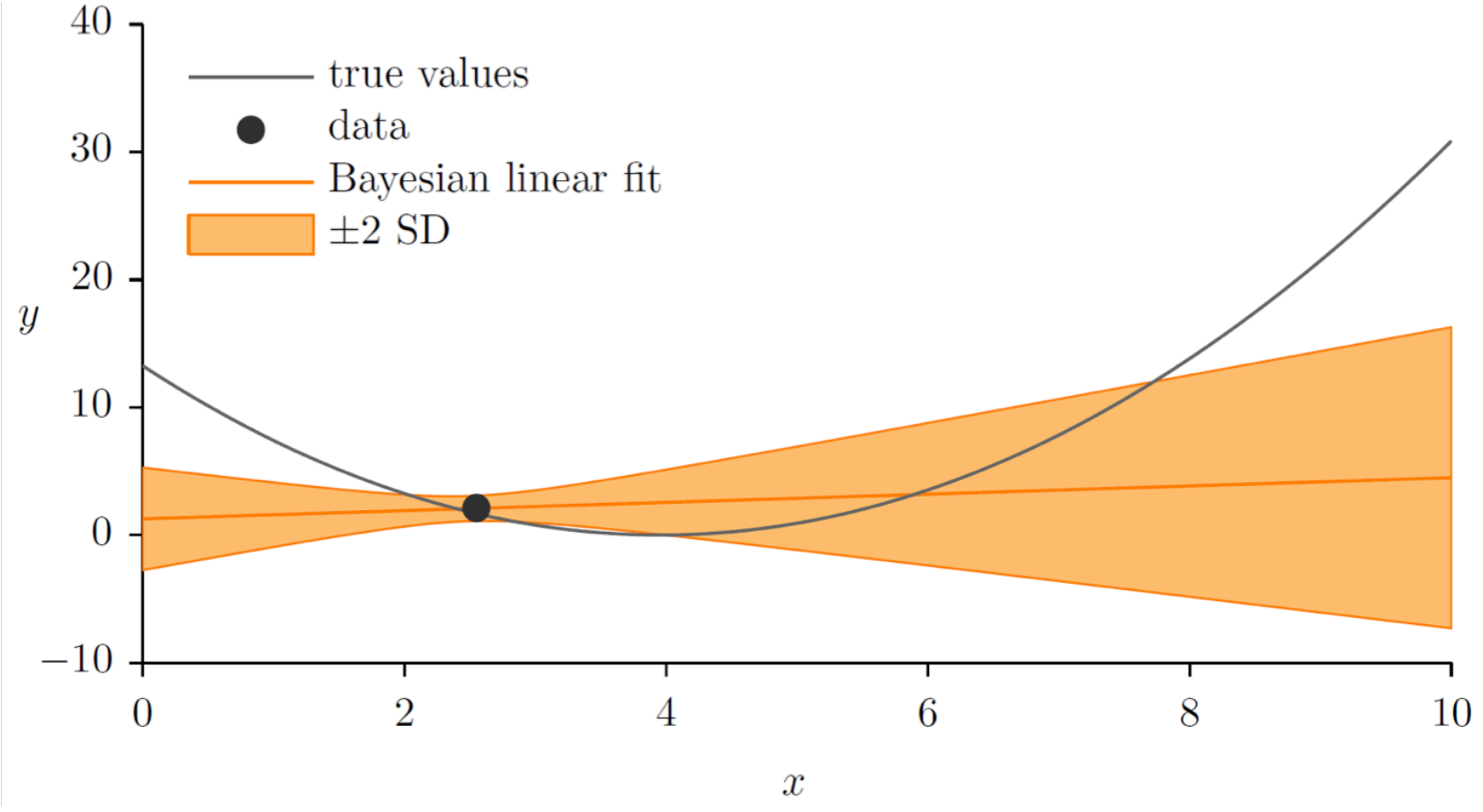
$$= \frac{1}{|x|} \frac{1}{\sqrt{2\pi(\beta/x)^2}} \exp -\frac{1}{2}\left(\frac{z/x - \alpha/x}{\beta/x}\right)^2.$$

# We can actually compute the posterior in the linear-Gaussian case.

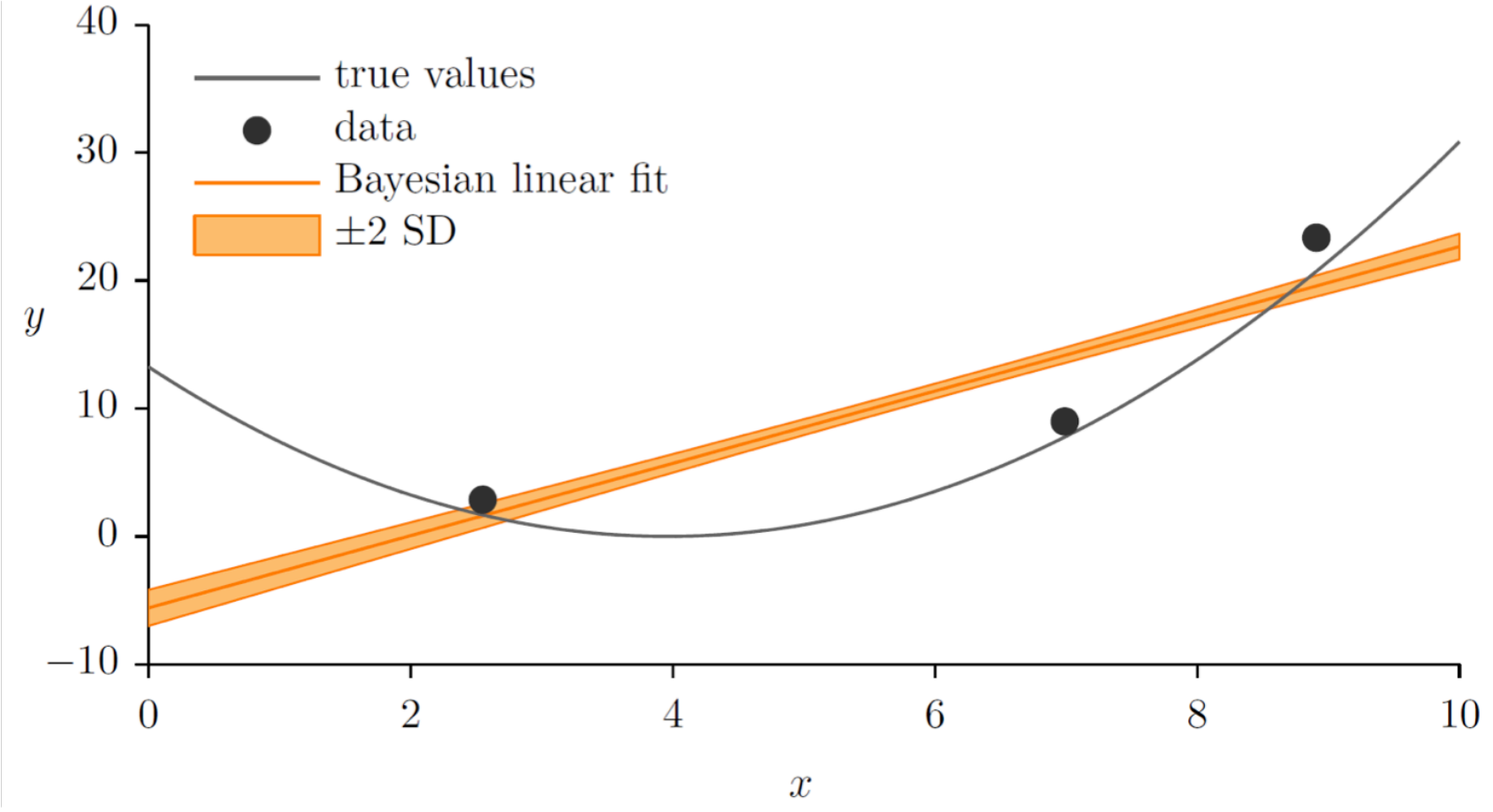
$$
\begin{aligned}
& p(f_\star \mid \mathcal{D}) \\
&= \int \delta(f_\star - \theta\, x_\star)\, \frac{\mathcal{N}(\theta; \mu, \nu^2) \prod_{i=1}^N \mathcal{N}\left(z_i/x_i; \theta, \sigma^2/x_i^2\right)}{A'}\, \mathrm{d}\theta \\
&= \int \delta(f_\star - \theta\, x_\star)\, \mathcal{N}\left(\theta; \frac{\nu^{-2}\mu + \sum_{i=1}^N \frac{x_i^2}{\sigma^2}\,\frac{z_i}{x_i}}{\nu^{-2} + \sum_{i=1}^N \frac{x_i^2}{\sigma^2}}, \frac{1}{\nu^{-2} + \sum_{i=1}^N \frac{x_i^2}{\sigma^2}}\right) \mathrm{d}\theta \\
&= \mathcal{N}\left(f_\star; \frac{\nu^{-2}\mu + \sum_{i=1}^N \frac{x_i z_i}{\sigma^2}}{\nu^{-2} + \sum_{i=1}^N \frac{x_i^2}{\sigma^2}}\, x_\star, \frac{x_\star^2}{\nu^{-2} + \sum_{i=1}^N \frac{x_i^2}{\sigma^2}}\right)
\end{aligned}
$$

Remember that for independent Gaussian measurements: precisions add; the mean is a precision-weighted sum; and a Gaussian prior is just like an additional measurement.

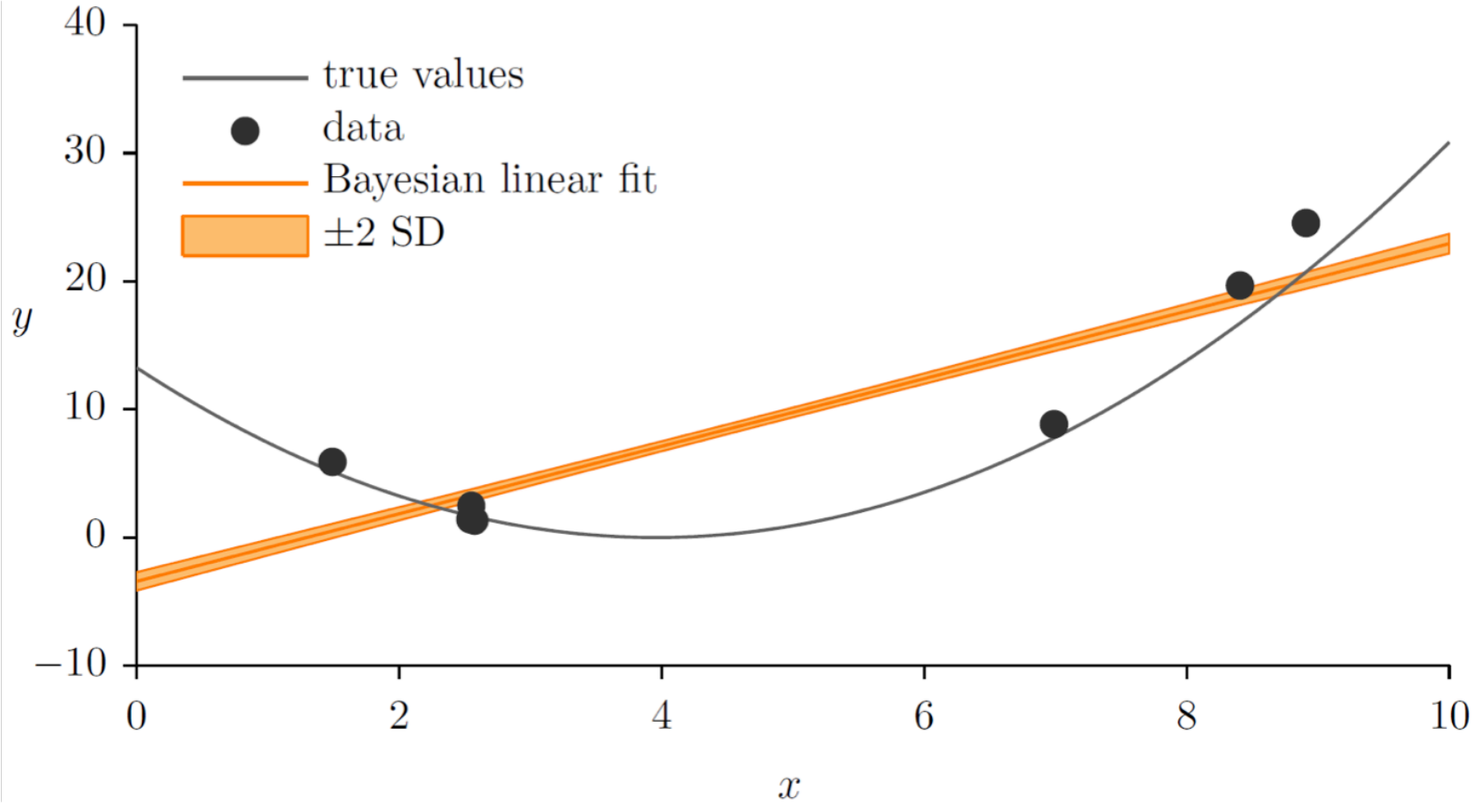
# Let's see Gaussian linear regression in action.



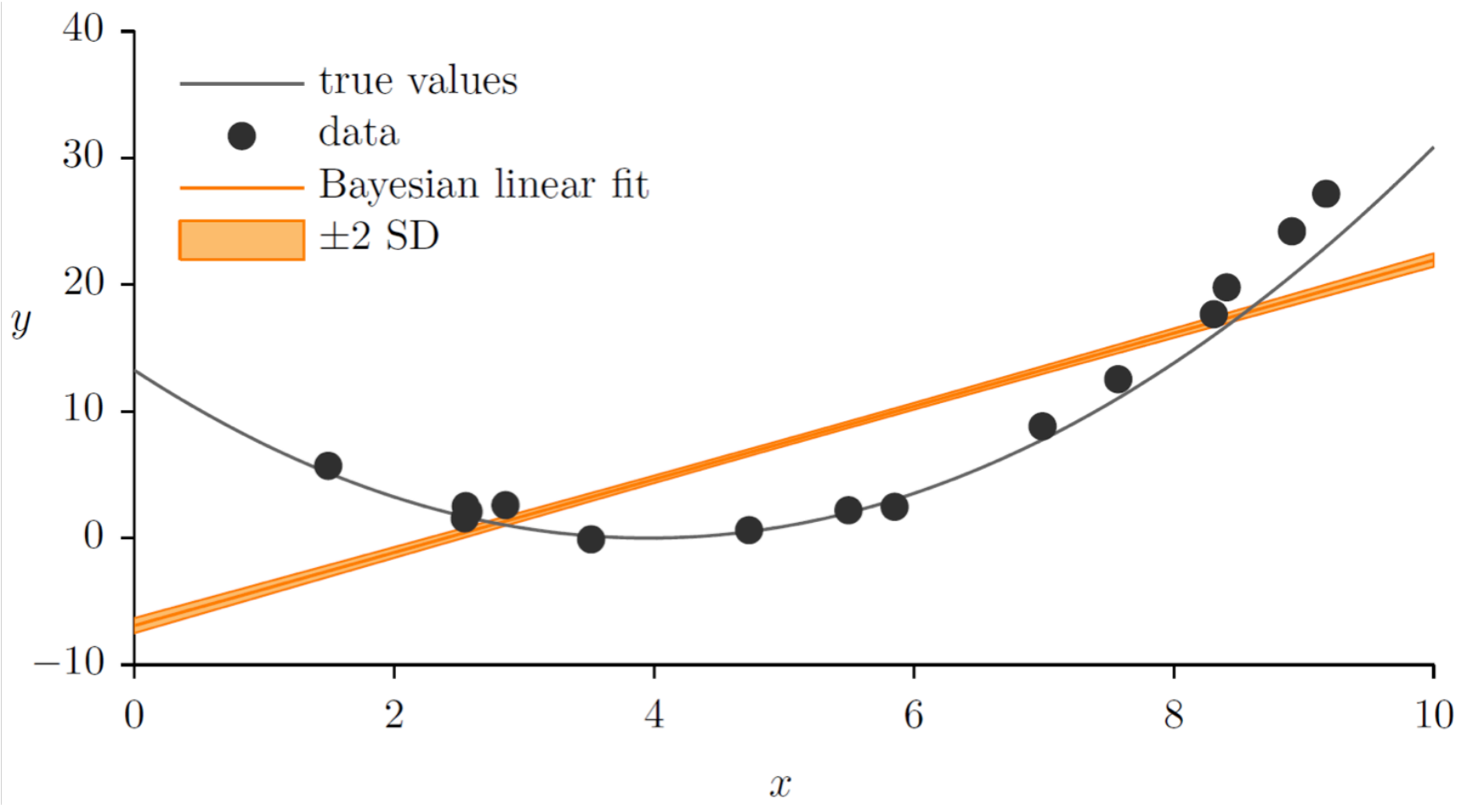
# Let's see Gaussian linear regression in action.



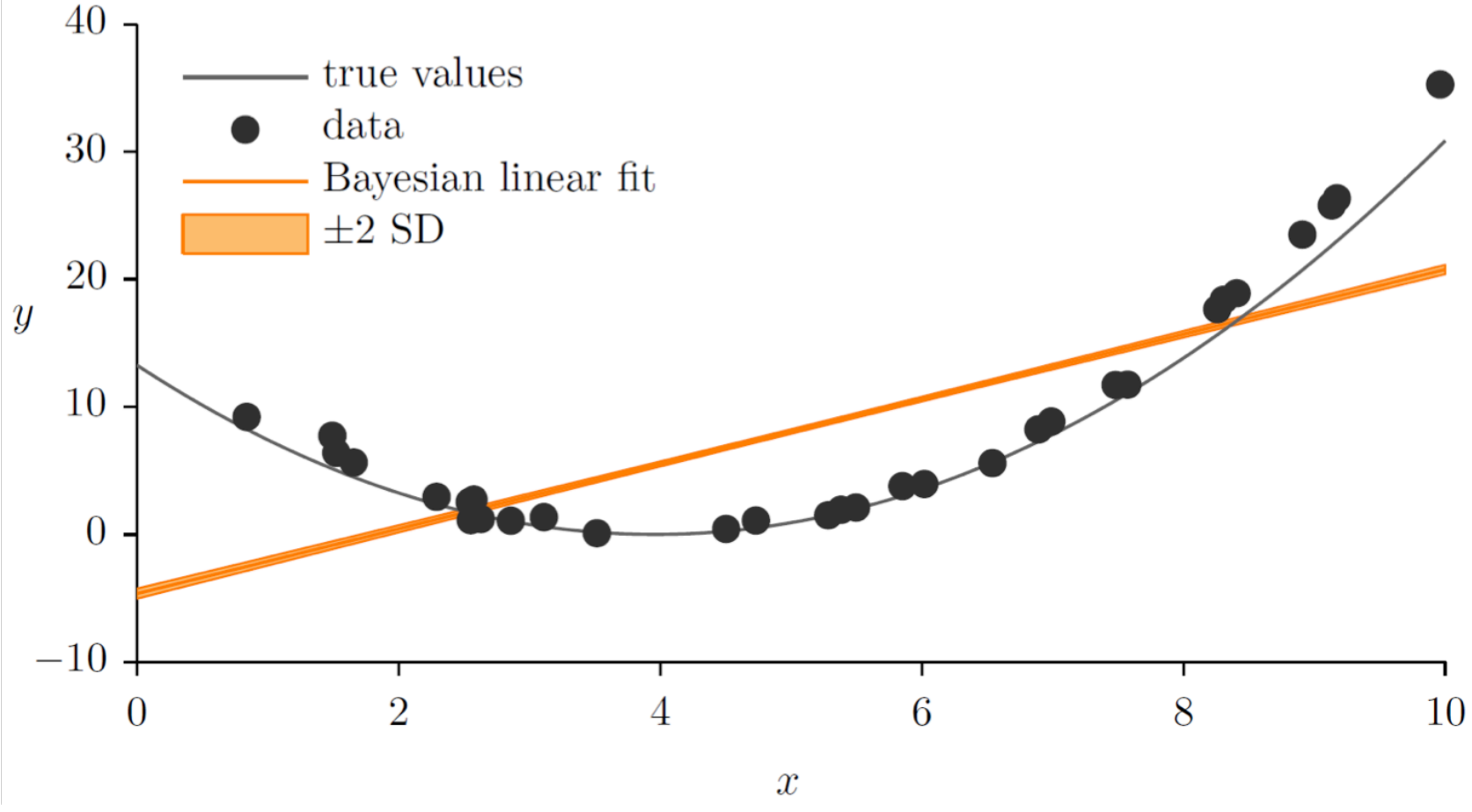
# Let's see Gaussian linear regression in action.



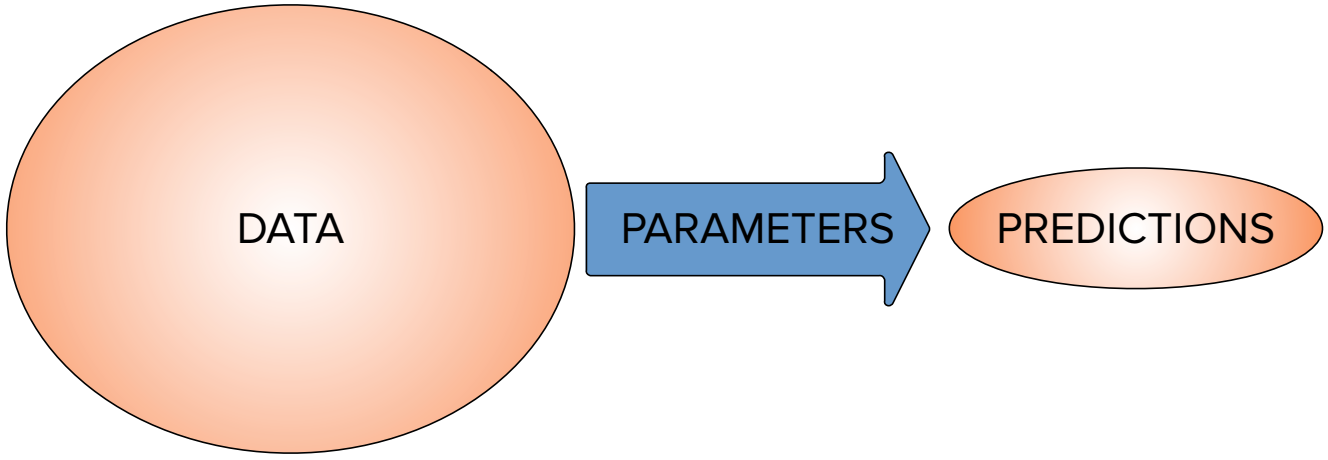
# Notice anything?



The algorithm becomes increasingly certain in its predictions, despite the linear model being completely inappropriate for the data.

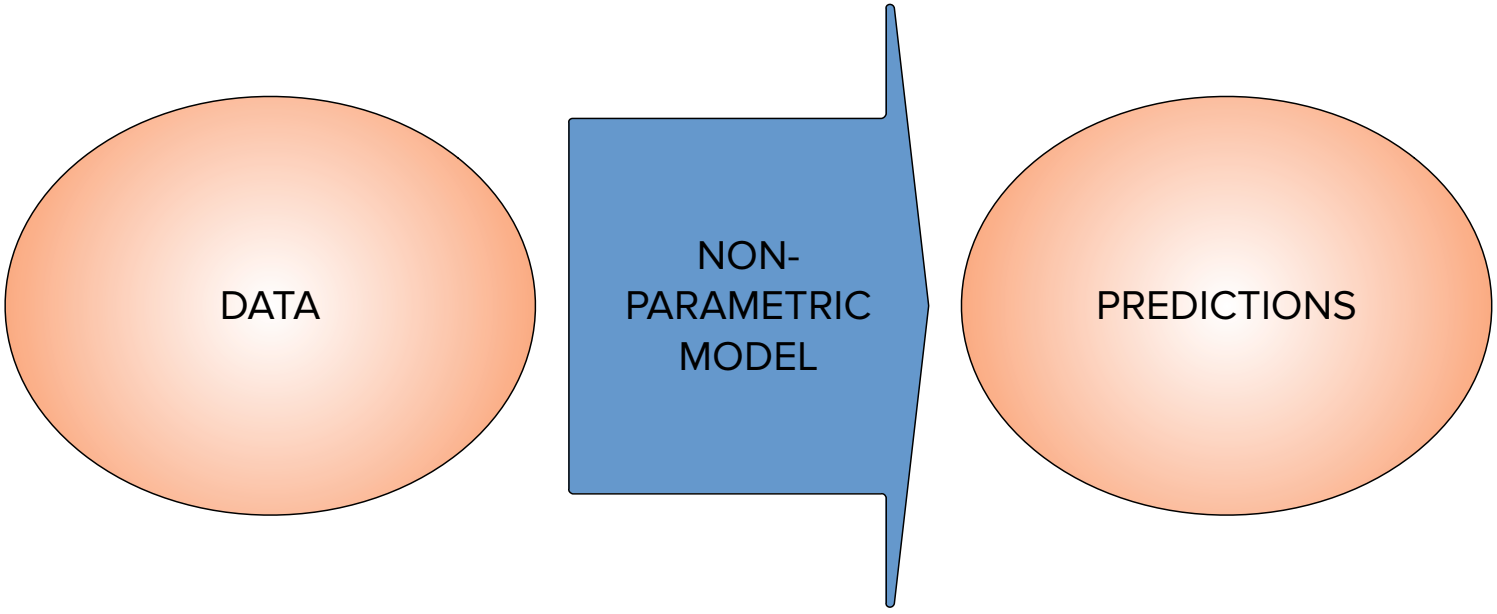


# The algorithm becomes increasingly certain in its predictions, despite the linear model being completely inappropriate for the data.



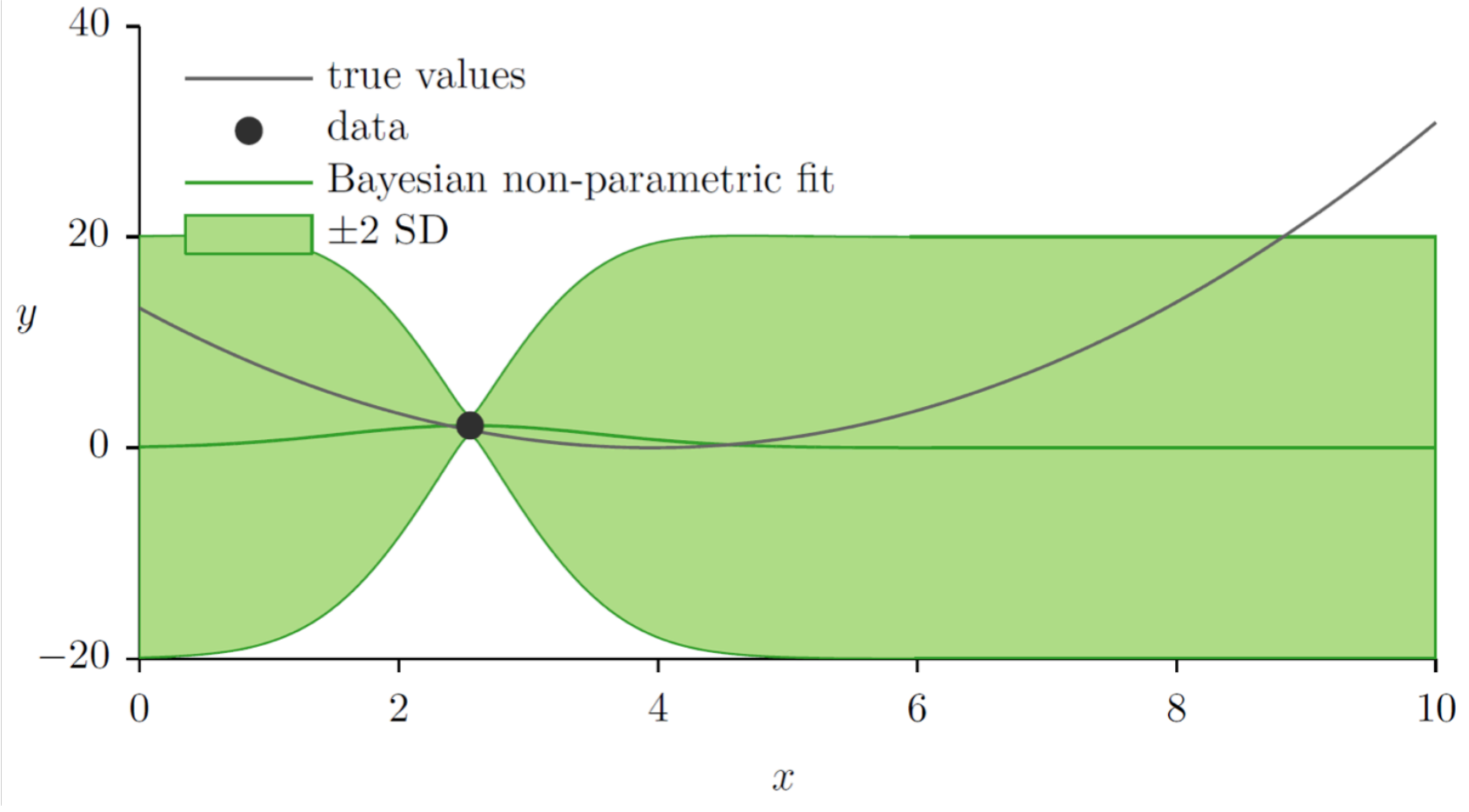

The problem is that the expressivity of the single-parameter linear model, $f = \theta x$, is saturated by the data. Increasing data means that the predictive variance, $\frac{x_\star^2}{\nu^{-2} + \sum_{i=1}^{N} \frac{x_i^2}{\sigma^2}}$, decreases, as we get more certain about $\theta$.

Non-parametric models have a number of parameters (and hence an expressivity) that grows with data.

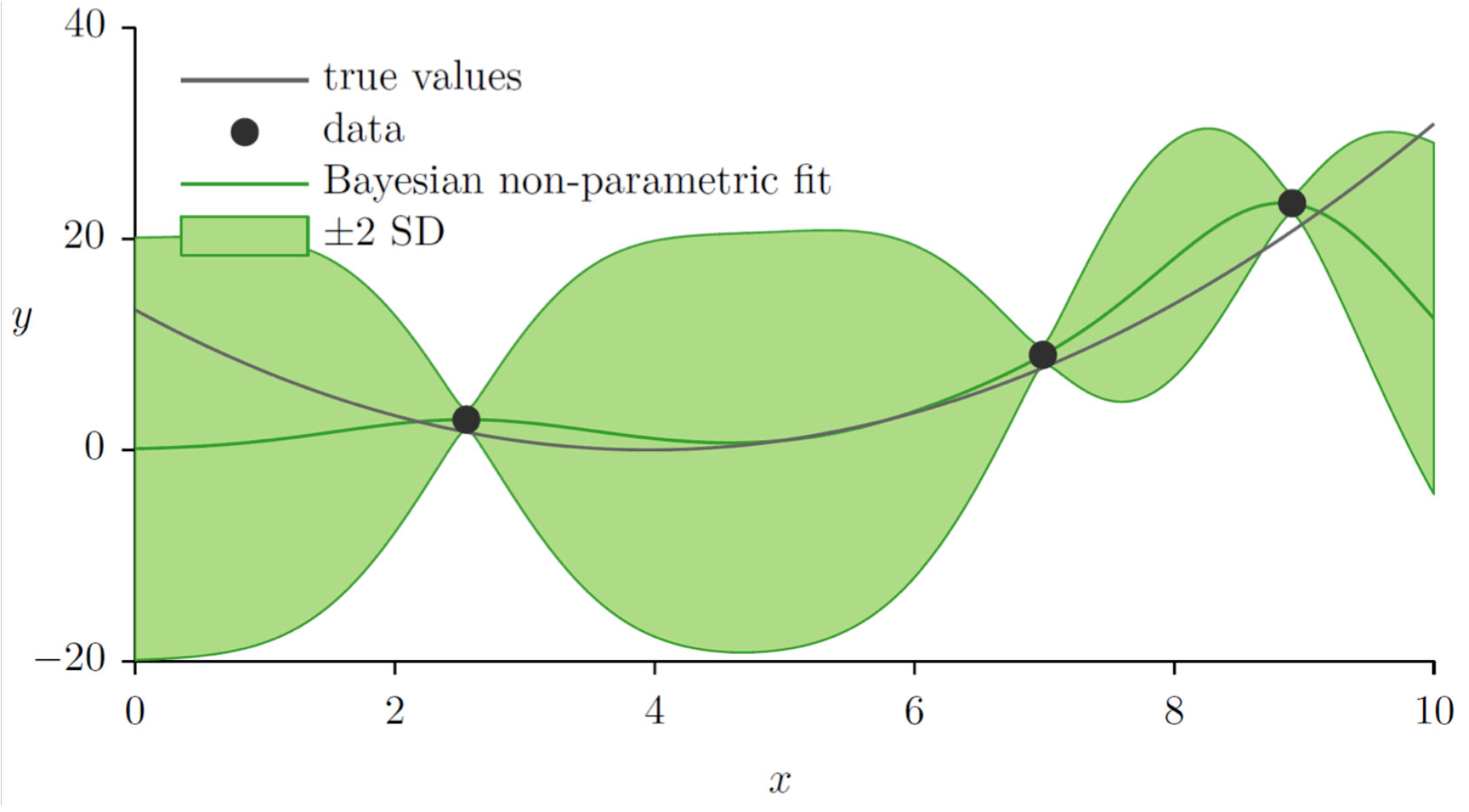



One example of such a model is a Gaussian process.

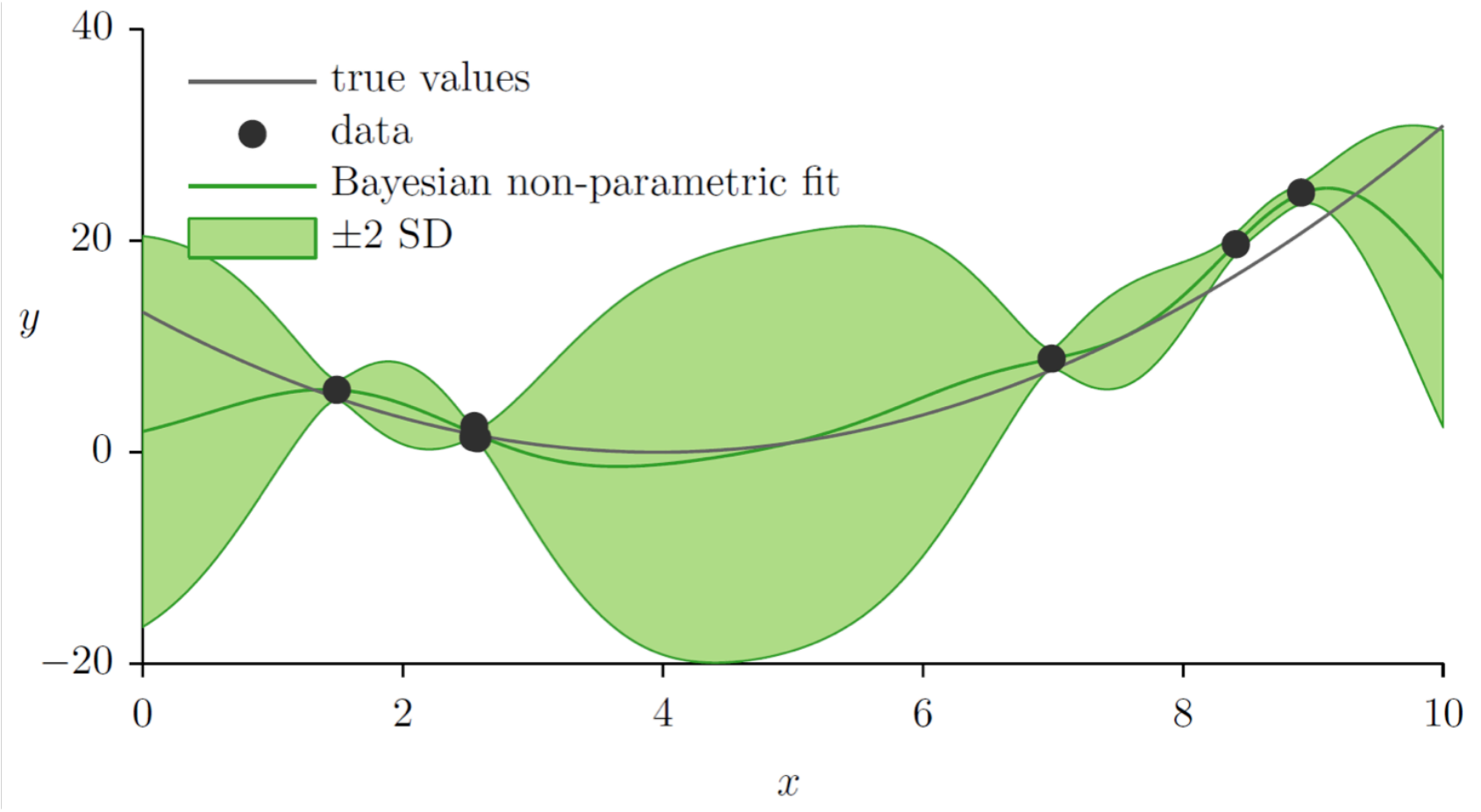
# Let's see Gaussian process regression in action.



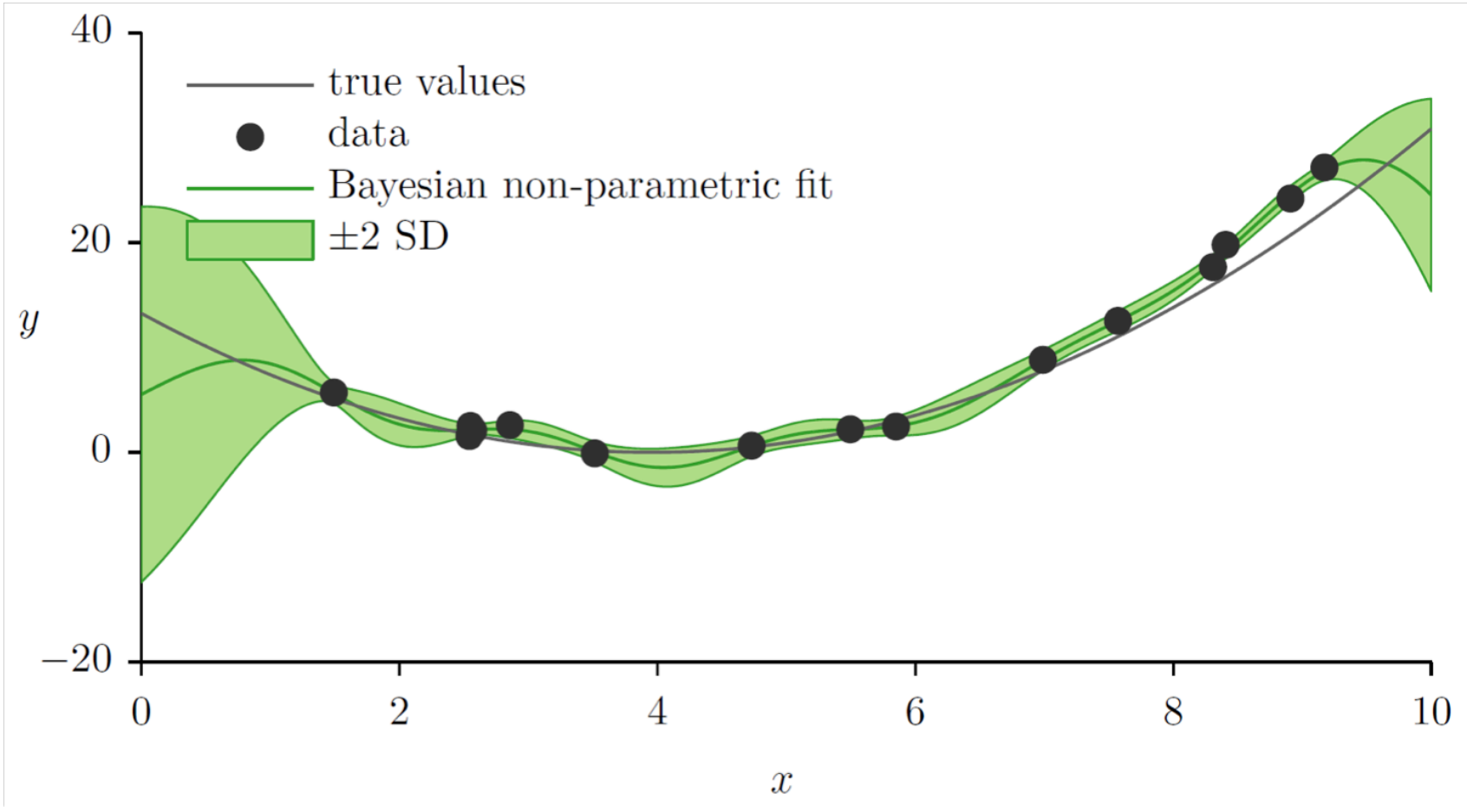
# Let's see Gaussian process regression in action.



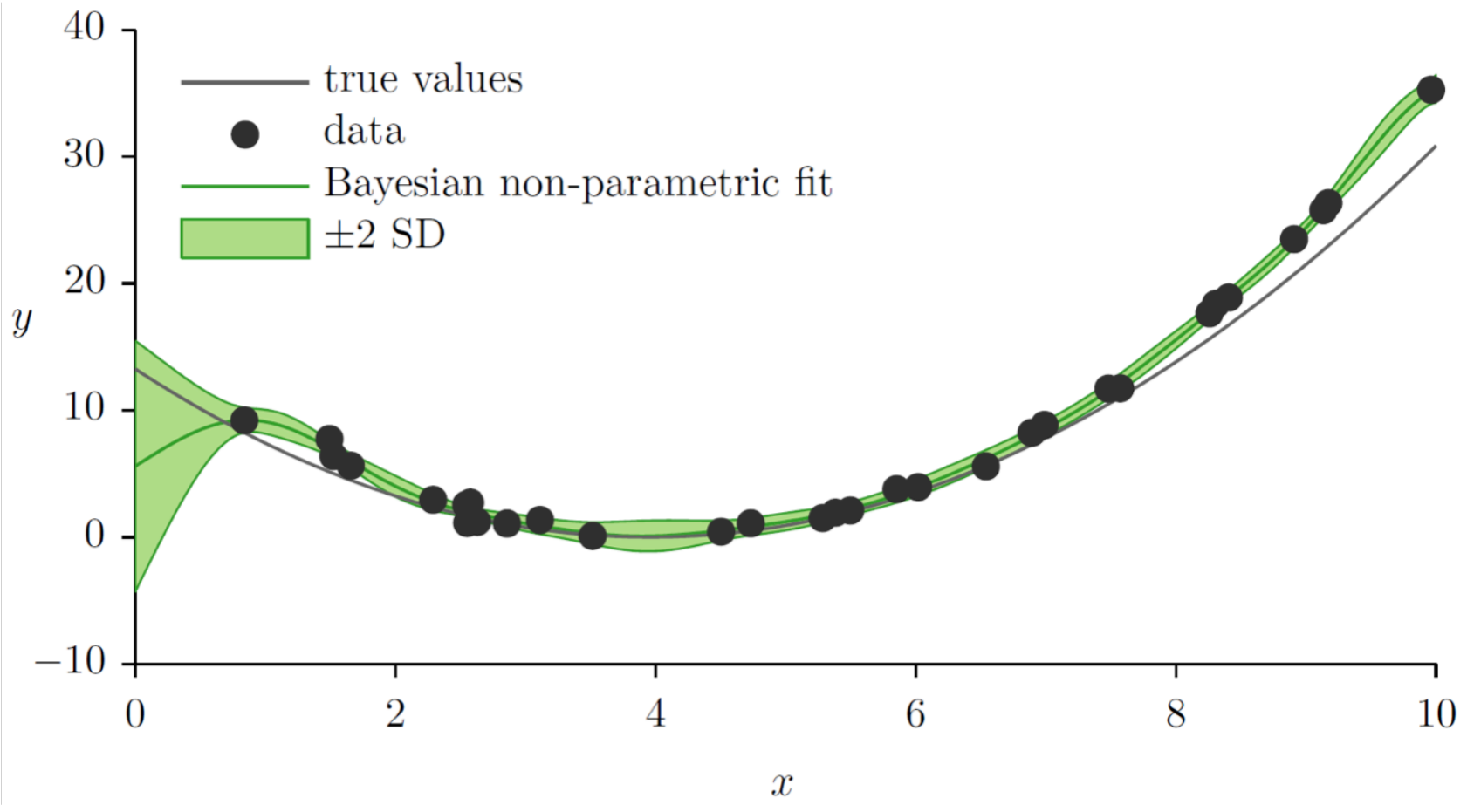
# Let's see Gaussian process regression in action.



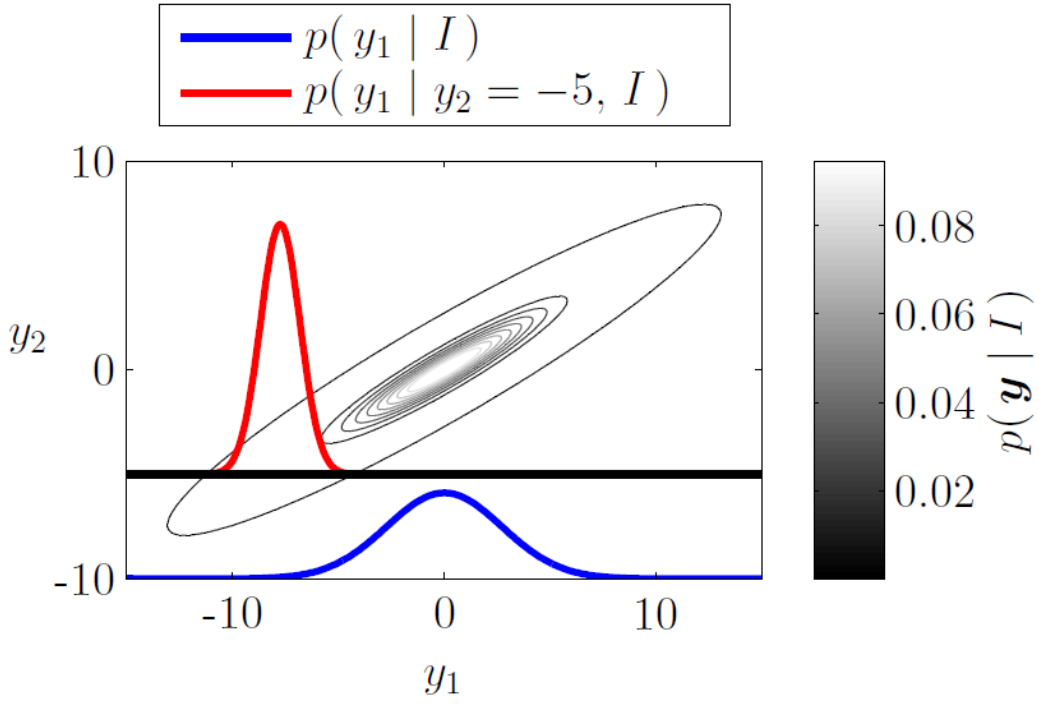
# Let's see Gaussian process regression in action.

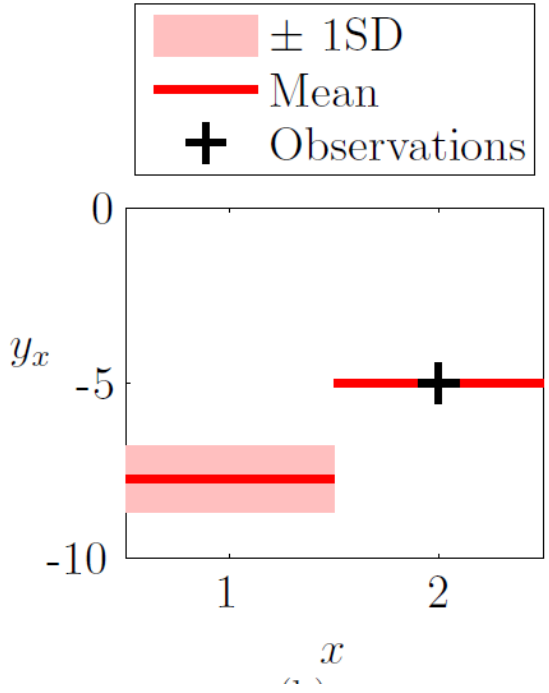


# The complexity adapts to the data, and the predictive uncertainties are more honest.

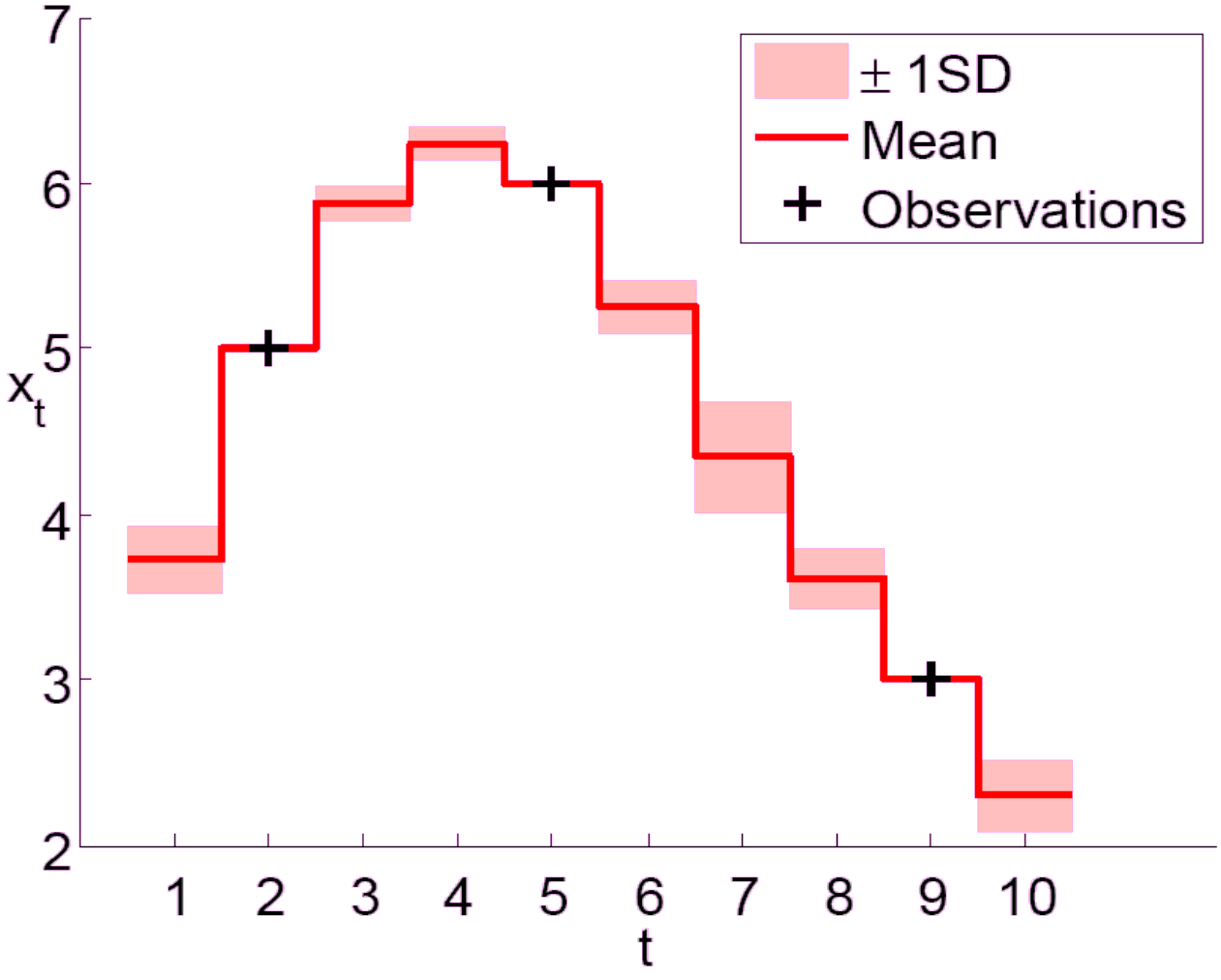


Recall that the Gaussian distribution allows us to produce distributions for variables conditioned on any other observed variables.

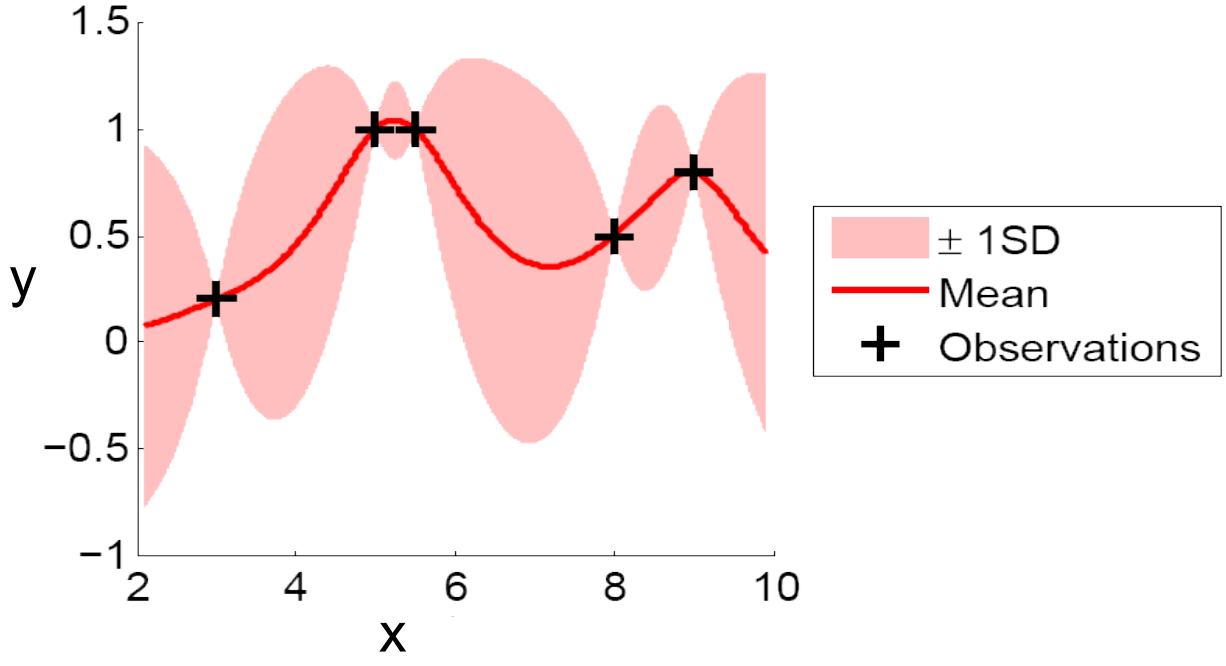


# These properties hold as we generalise to ten variables.



# A Gaussian process is the generalisation to a potentially infinite number of variables: a function!




A multivariate Gaussian is specified by a mean vector and a covariance matrix. A Gaussian process is specified by a mean function $\mu$ and a covariance function $K$.

# Recall the multivariate Gaussian pdf is

$$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{|2\pi\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

where $|2\pi\boldsymbol{\Sigma}|$ is the determinant of the matrix that is $\boldsymbol{\Sigma}$ with all its entries multiplied by $2\pi$.

# The covariance matrix must be positive semi-definite.

A matrix $\Sigma$ is positive semi-definite if, for non-zero vector $v$,

$$v^\top \Sigma v \geq 0.$$

Note that this means that $\Sigma$ is symmetric and invertible, and $\Sigma^{-1}$ is positive semi-definite:

$$0 \leq v^\top \Sigma v = (v^\top \Sigma)\Sigma^{-1}(\Sigma v) = (\Sigma v)^\top \Sigma^{-1}\Sigma v = u^\top \Sigma^{-1} u.$$

The Gaussian pdf is hence

$$\frac{1}{(2\pi)^{\frac{n}{2}}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right) \leq \frac{1}{(2\pi)^{\frac{n}{2}}|\boldsymbol{\Sigma}|^{1/2}}.$$

# Formally, for some function $f(x)$ and points $x_1, x_2, x_3, \ldots$, we write $p(f) = \mathcal{GP}(f; \mu, K)$ if

$$p\left(\begin{bmatrix} f(x_1) \\ f(x_2) \\ f(x_3) \\ \vdots \end{bmatrix}\right) =$$

$$\mathcal{N}\left(\begin{bmatrix} f(x_1) \\ f(x_2) \\ f(x_3) \\ \vdots \end{bmatrix}; \begin{bmatrix} \mu(x_1) \\ \mu(x_2) \\ \mu(x_3) \\ \vdots \end{bmatrix}, \begin{bmatrix} K(x_1,x_1) & K(x_1,x_2) & K(x_1,x_3) & \cdots \\ K(x_2,x_1) & K(x_2,x_2) & K(x_2,x_3) & \cdots \\ K(x_3,x_1) & K(x_3,x_2) & K(x_3,x_3) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}\right).$$

That is, a Gaussian process gives a good old multivariate Gaussian over any finite collection of function values.

# Recalling what you know about covariance matrices, which of the following could be a covariance function?

Assuming $x, x' \in [0, 1) \subset \mathbb{R}$

1. $K(x, x') = 1 - |x - x'|$.
2. $K(x, x') = \mathcal{N}(x; \mu, x')$.
3. $K(x, x') = \mathcal{P}o(x; x')$.
4. $K(x, x') = 1 - 2x\,x'$.

# Which could be a covariance function?

1. $K(x, x') = 1 - |x - x'|$.
   *This is an example of a spline covariance, but that's not so important right now.*
2. $K(x, x') = \mathcal{N}(x; \mu, x')$.
   *This function is not symmetric in $x$ and $x'$, meaning that the covariance matrix will not be symmetric.*
3. $K(x, x') = \mathcal{P}o(x; x')$.
   $\mathcal{P}o(x; x') = \frac{\mu^x}{x!} e^{-\mu}$ *is not symmetric in $x$ and $x'$, meaning that the covariance matrix will not be symmetric.*
4. $K(x, x') = 1 - 2x\,x'$.
   *For sufficiently large $x$, $K(x, x)$, which must be a variance, will be negative. Hence the covariance matrix will not be positive semi-definite.*

# The posterior mean and covariance equations follows simply from Gaussian identities.

Consider predicting for predictants $f_* = f(x_*)$ given data $f_d = f(x_d)$.

$$p\left(\begin{bmatrix} f_* \\ f_d \end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix} f_* \\ f_d \end{bmatrix}; \begin{bmatrix} \mu(x_*) \\ \mu(x_d) \end{bmatrix}, \begin{bmatrix} K(x_*, x_*) & K(x_*, x_d) \\ K(x_d, x_*) & K(x_d, x_d) \end{bmatrix}\right)$$

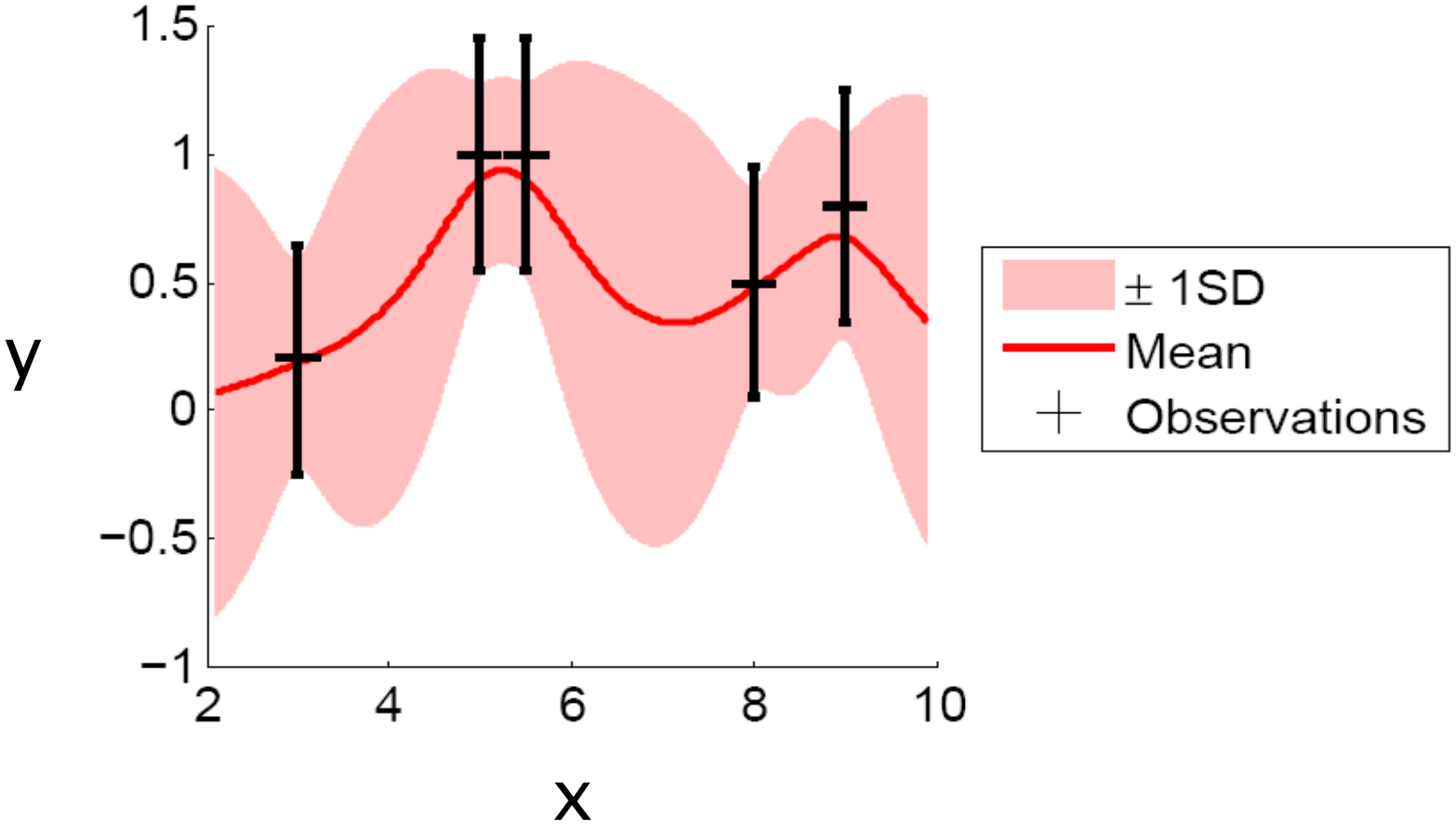
$$\Rightarrow p(f_* \mid f_d) = \mathcal{N}\left(f_*; m_{*|d}, C_{*|d}\right),$$

where

$$m_{*|d} = \mu(x_*) + K(x_*, x_d)K(x_d, x_d)^{-1}\left(f_d - \mu(x_d)\right)$$
$$C_{*|d} = K(x_*, x_*) - K(x_*, x_d)K(x_d, x_d)^{-1}K(x_d, x_*).$$

Note that this all holds even if $x_*$ and $x_d$ are matrices, with rows corresponding to instances, and columns to features.

# A Gaussian process can accommodate noise.



We usually consider making independent and identically distributed (IID) Gaussian noisy measurements $y_d$, of $f_d$; giving

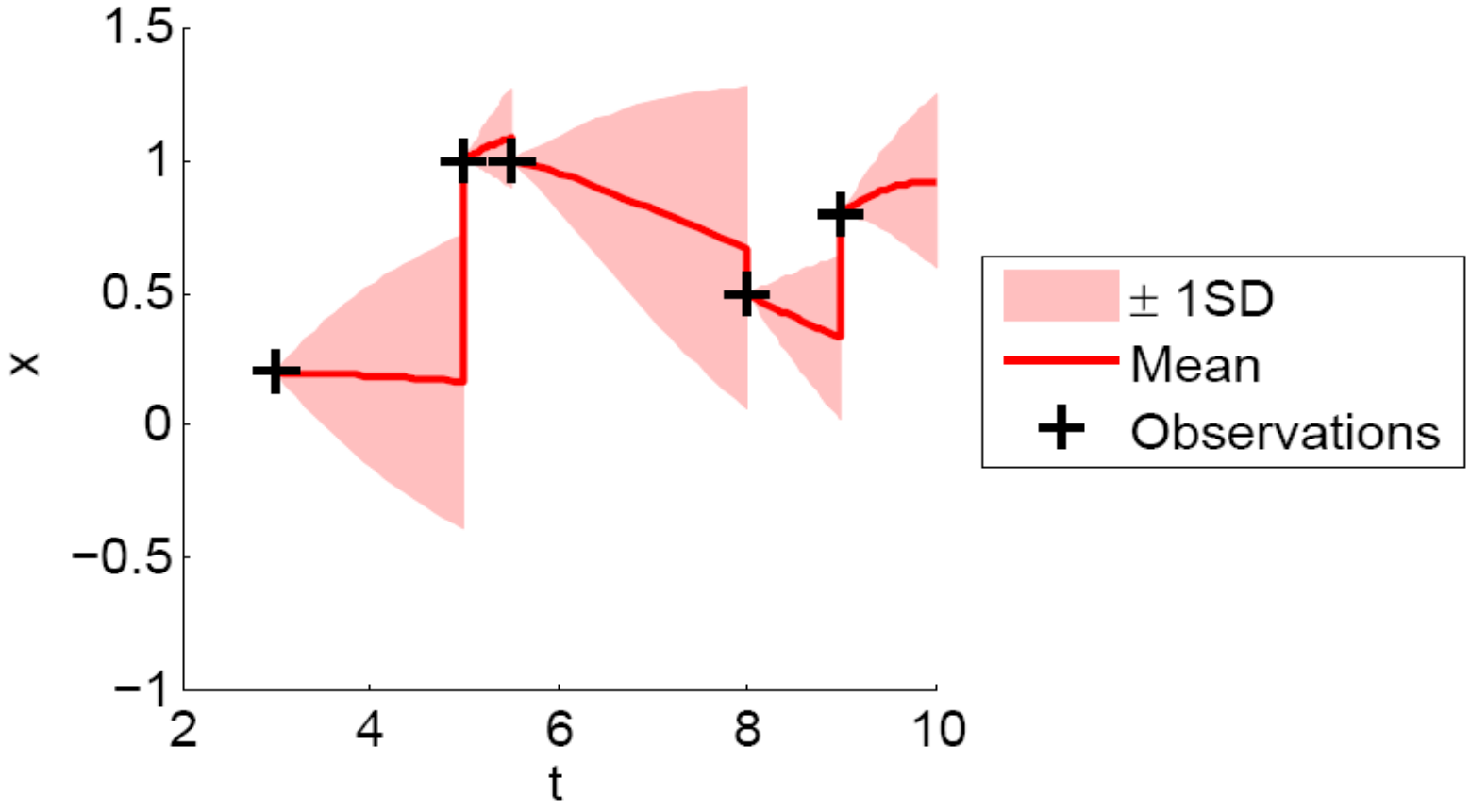
$$\begin{aligned} p(y_d \mid f_d) &= \prod_i p(y_{d,i} \mid f_{d,i}) \\ &= \prod_i \mathcal{N}(y_{d,i}\,; f_{d,i}\,, \sigma^2) \\ &= \mathcal{N}(y_d; f_d, \sigma^2 I_d), \end{aligned}$$

where $I_d$ is an identity matrix of length equal to that of $y_d$. Our predictive mean and variance become

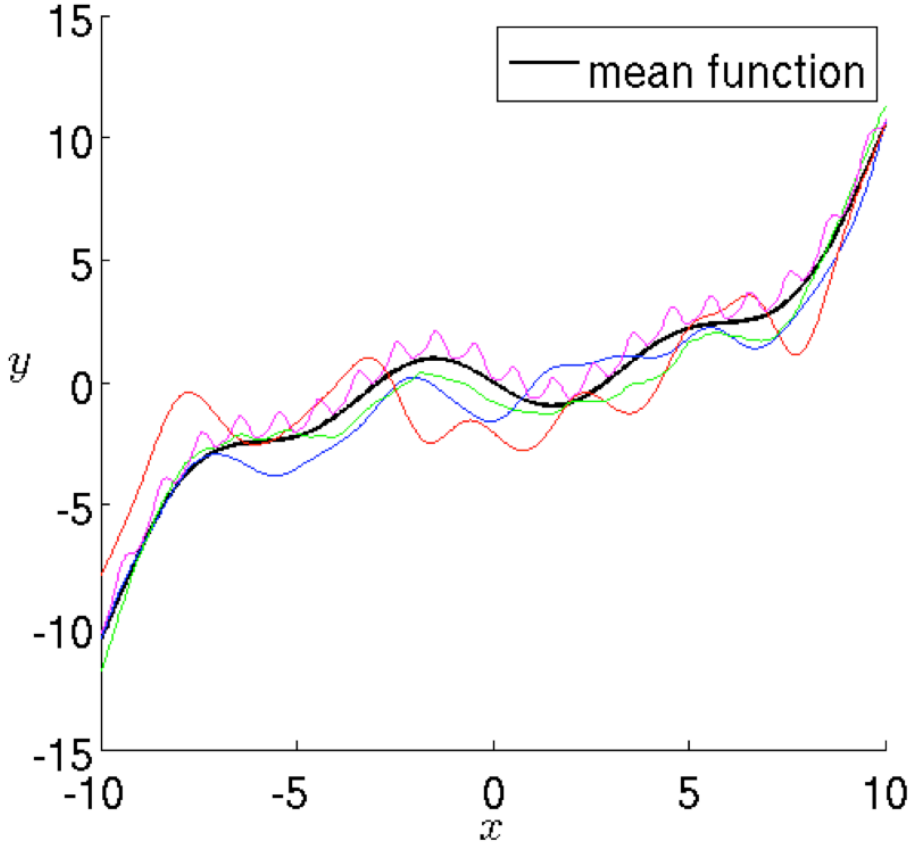
$$m_{*|d} = \mu(x_*) + K(x_*, x_d)\big(K(x_d, x_d) + \sigma^2 I_d\big)^{-1}\big(y_d - \mu(x_d)\big)$$

$$C_{*|d} = K(x_*, x_*) - K(x_*, x_d)\big(K(x_d, x_d) + \sigma^2 I_d\big)^{-1} K(x_d, x_*).$$

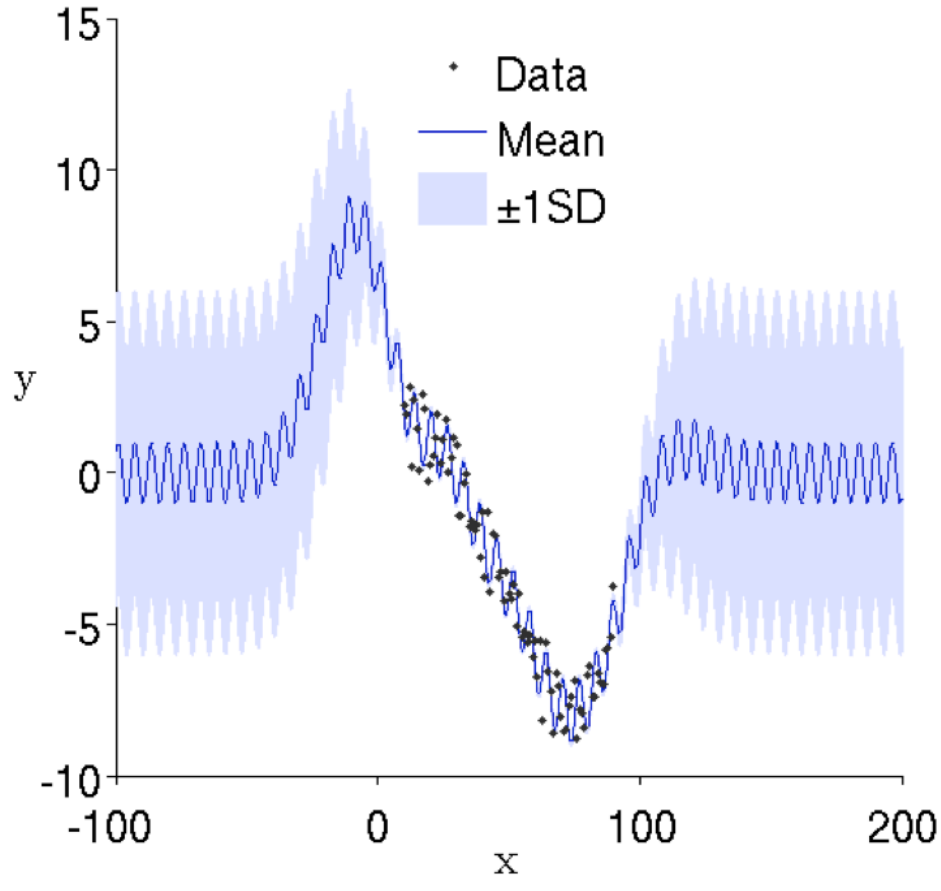
# We often want to address functions of time, using Gaussian processes for tracking.



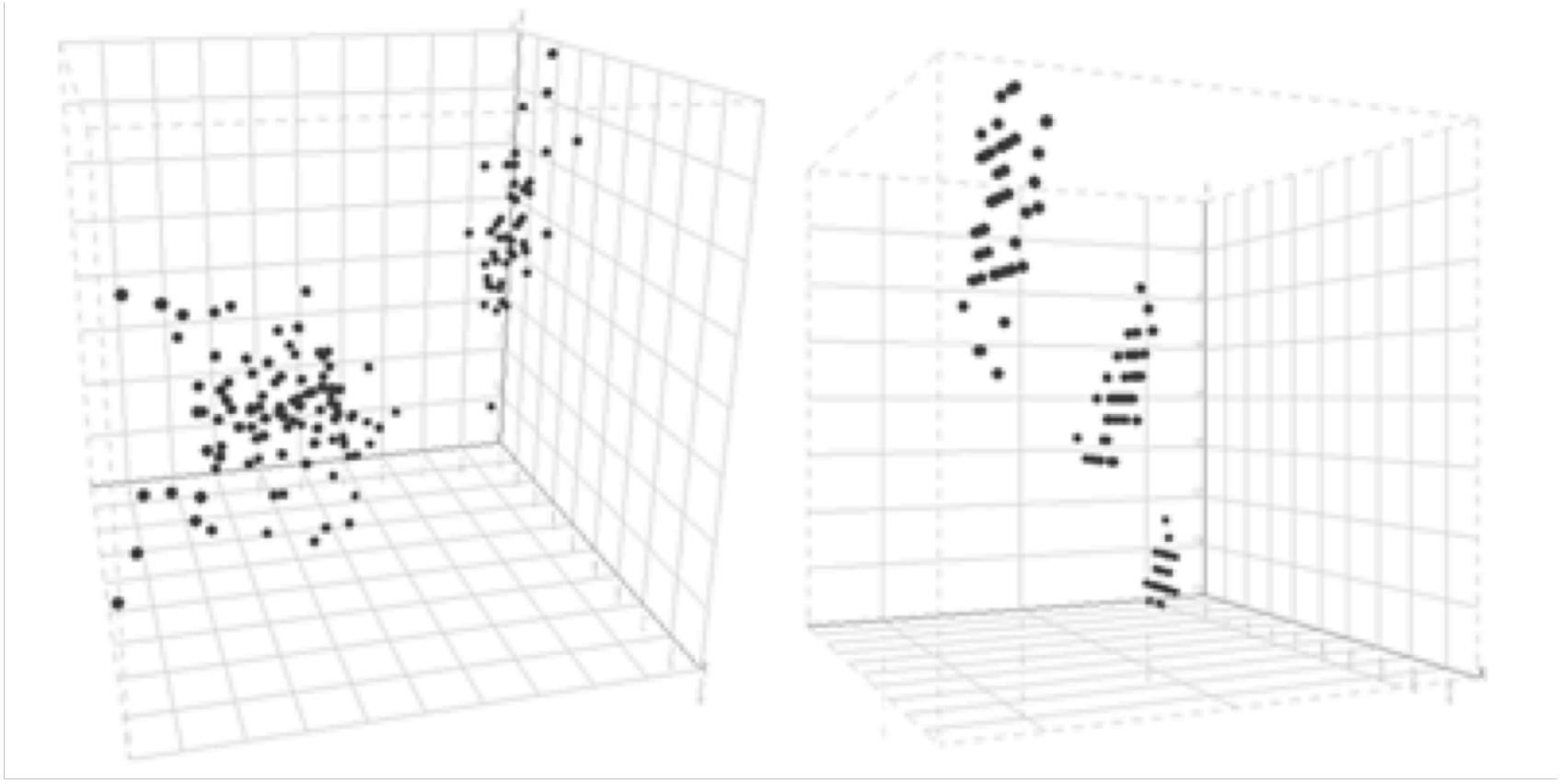
The prior mean function $\mu(x)$ should be our best guess (of any form) for the function $y(x)$ before any observations are made.



The prior mean function is the function our inference will default to far from observations, important for extrapolation.



# Test points far from observations are an important consideration for high-dimensional data.



# Here are my heuristics for selecting a mean function.

| Conditions | Mean function ($\mu$) |
|---|---|
| Exceptionally lazy | 0 |
| Ordinarily lazy | mean($y_d$) |
| Interested only in interpolation | constant, $\theta$ |
| Extrapolation required | Bespoke parametric model built using domain knowledge |

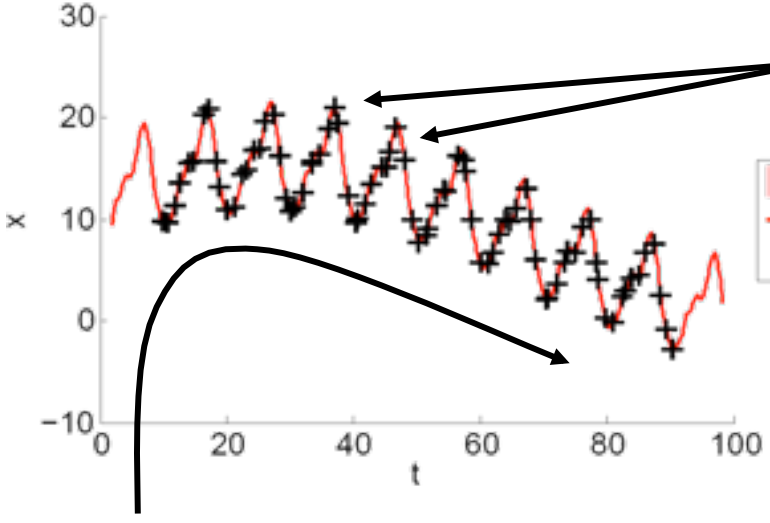
# Most mean and covariance functions are specified by a small number (maybe 10 total) of hyperparameters.

The parameters of the model are the function values, of which there are a potentially infinite number (one for every possible location).

Hence a Gaussian process's expressivity is potentially unbounded despite a few hyperparameters having snuck in through the back door.

We'll come back to discussion of how to manage these hyperparameters (for which we'll use lowercase greek), but, for now, we will collect them into a vector $\theta$.

# Covariance functions allow us to incorporate structure and correlations into our models.

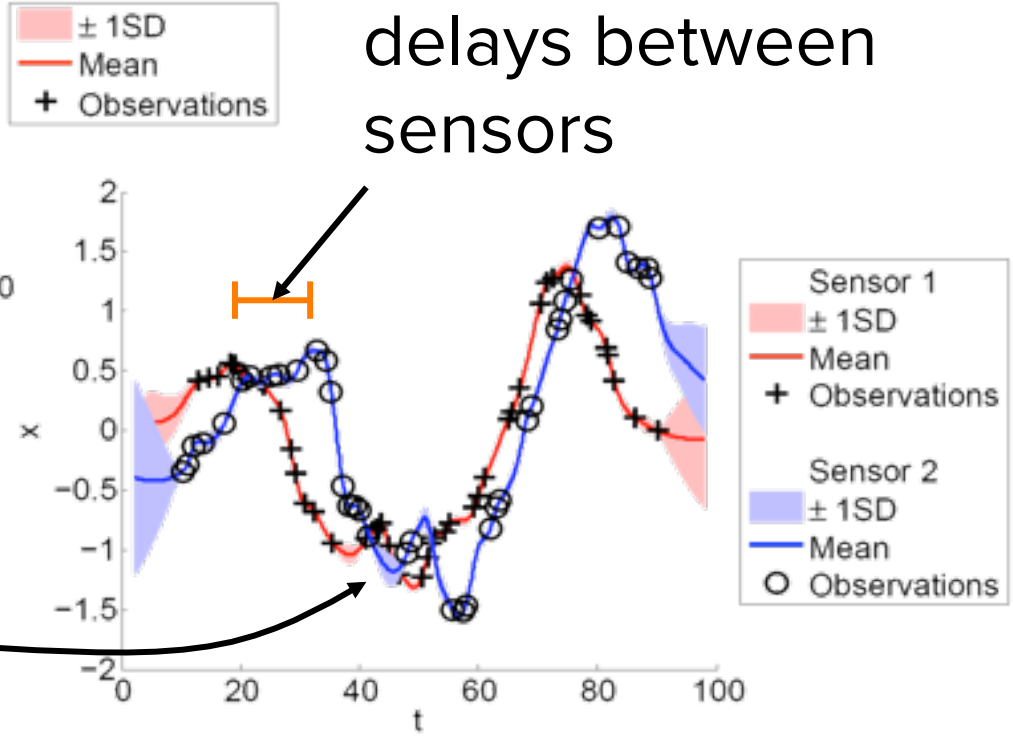



periodicity

long-term drifts

delays between sensors

correlated sensors



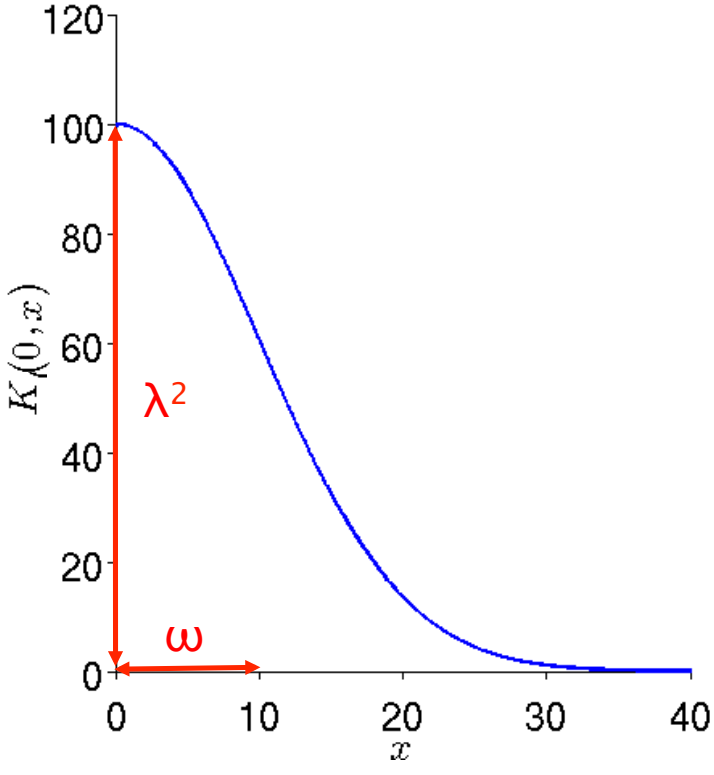
# Many common covariance functions express that covariance decreases with increasing Euclidean distance:

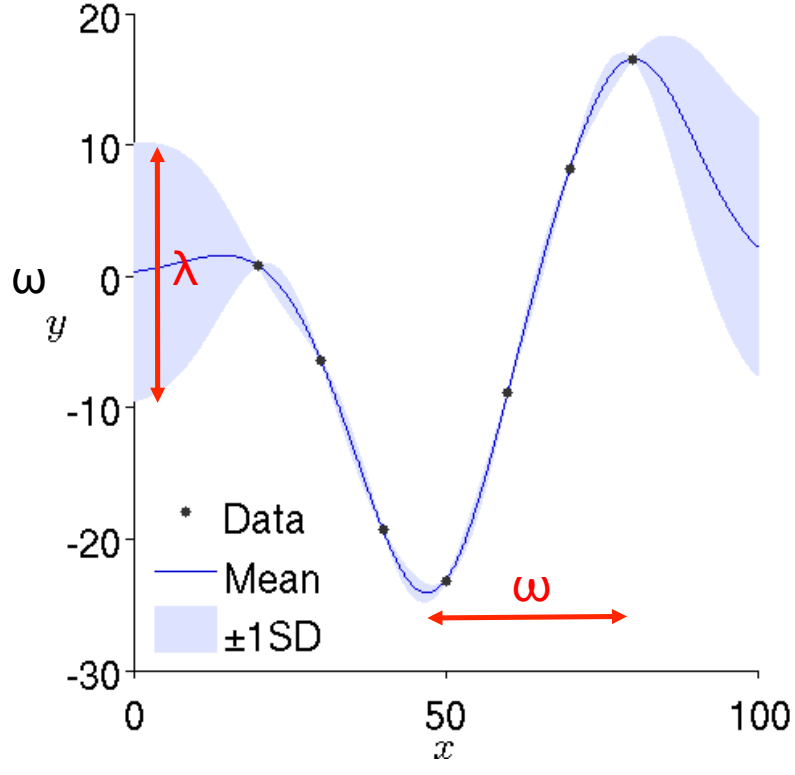
$$K(x, x'; \theta) = \lambda^2 f\left(\frac{d(x, x')}{\omega}\right),$$

where $f$ is a monotonically decreasing function, and $d$ is the Euclidean distance,

$$d(x, x') = \sqrt{\sum_i (x_i - x'_i)^2}.$$

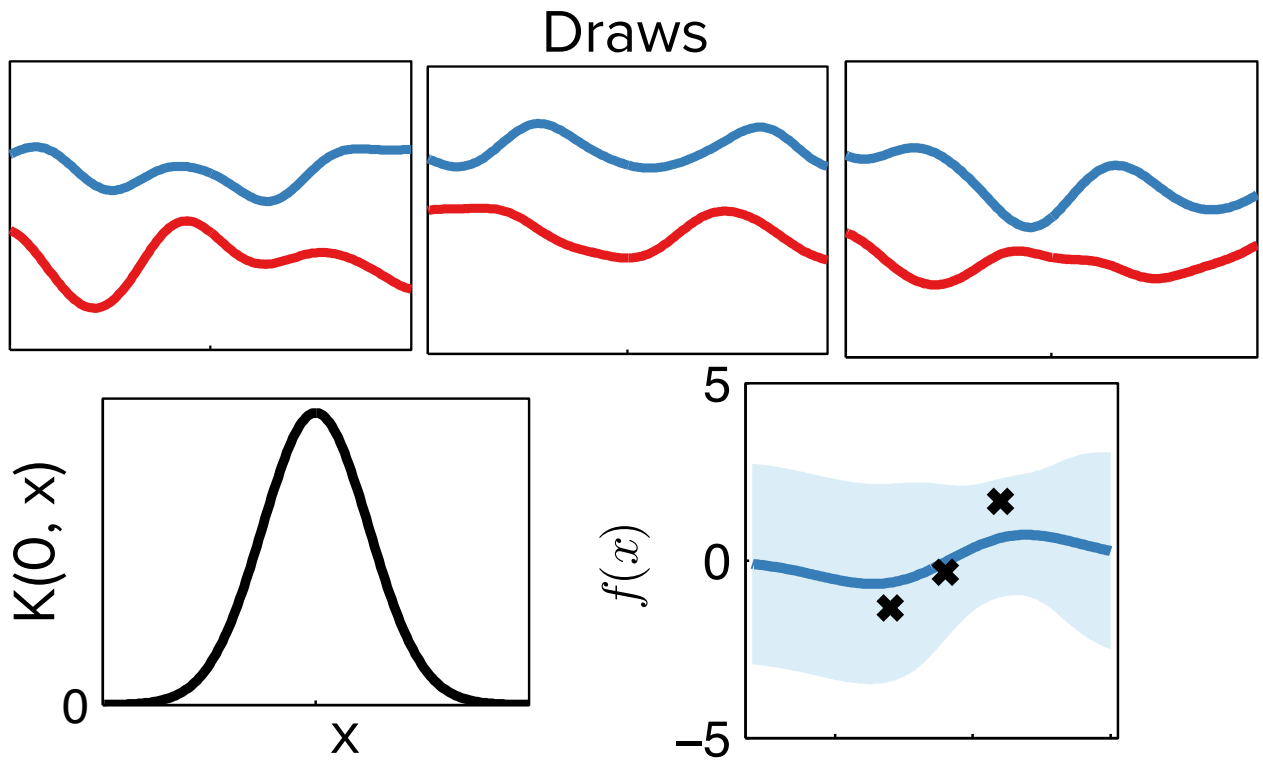
The hyperparameters $\lambda > 0$ and $\omega > 0$ specify our expected length scales of the function in output ('height') and input ('width') spaces respectively.



# The exponentiated quadratic covariance is useful for smooth data:

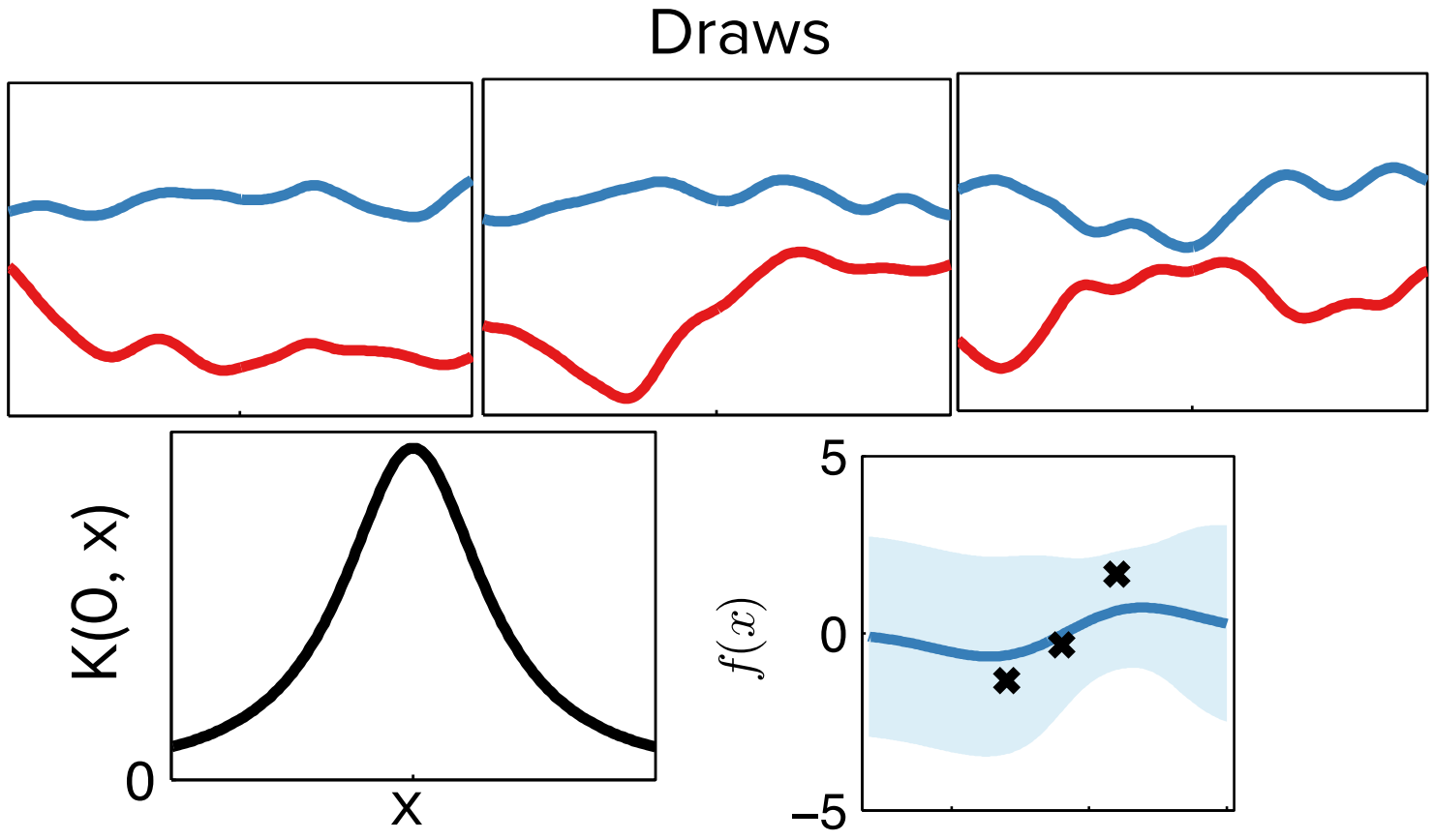
$$K(x, x'; \theta) = \lambda^2 \exp -\frac{1}{2}\left(\frac{d(x, x')}{\omega}\right)^2.$$




*All plots courtesy of David Duvenaud.*

# Use the rational quadratic covariance for data that is smooth across a range of length scales:

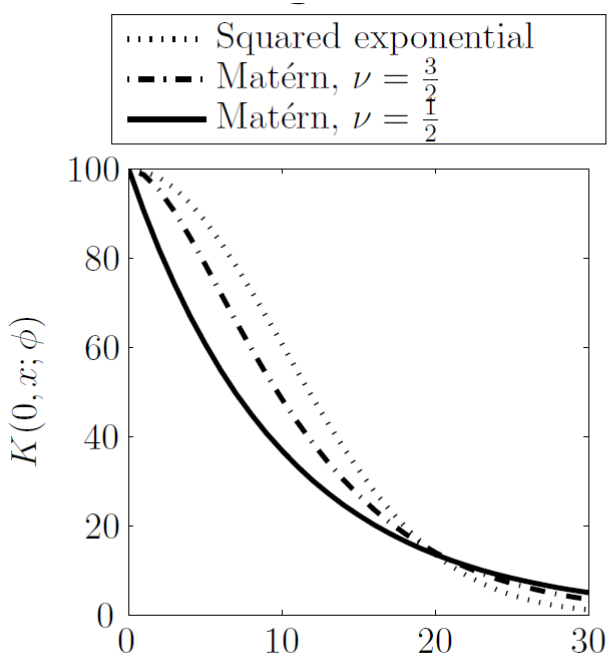
$$K(x, x'; \theta) = \lambda^2 \left(1 + \frac{1}{2\alpha}\left(\frac{d(x, x')}{\omega}\right)^2\right)^{-\alpha}.$$

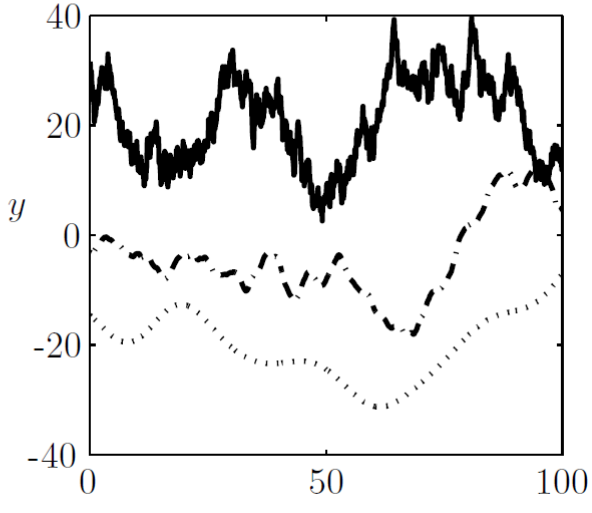


# The Matérn class of covariances are useful for functions of variable smoothness:

$$K(x, x'; \theta, \nu = 1/2) = \lambda^2 \exp\left(-\frac{d(x, x')}{\omega}\right);$$

$$K(x, x'; \theta, \nu = 3/2) = \lambda^2 \left(1 + \sqrt{3}\frac{d(x, x')}{\omega}\right) \exp\left(-\sqrt{3}\frac{d(x, x')}{\omega}\right).$$

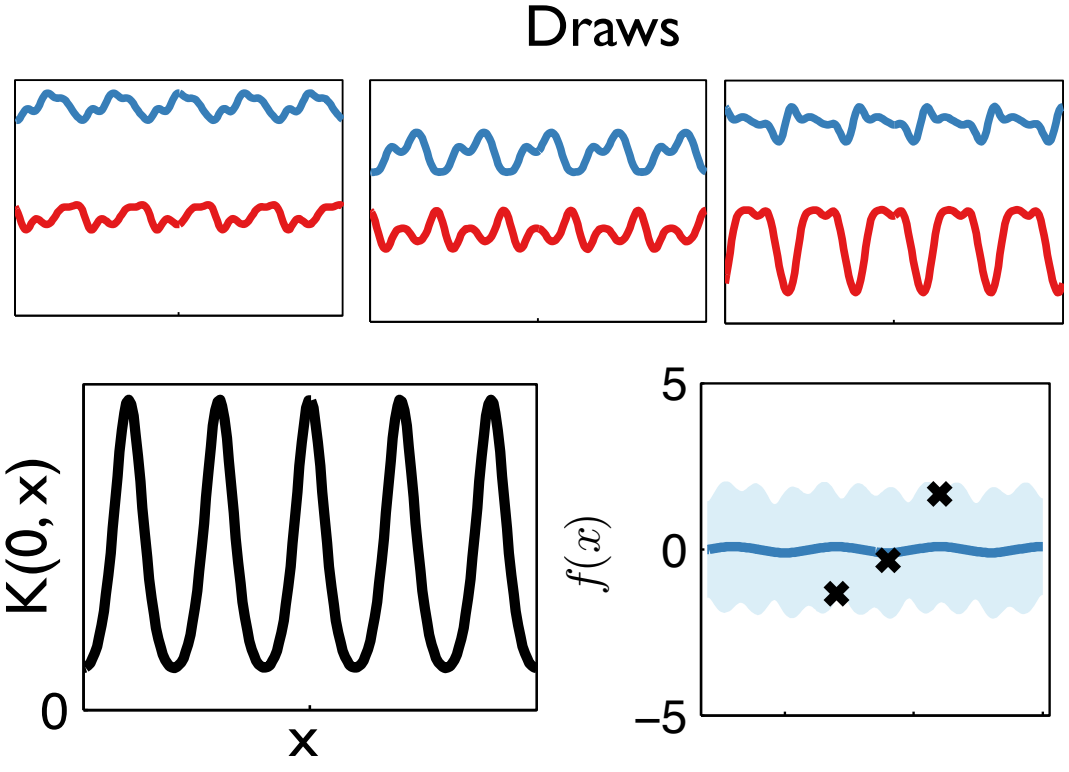


# The periodic covariance is useful for periodic data:

$$K(x, x'; \theta) = \lambda^2 \exp\left( -\frac{2\left(\sin\left(\pi\, d(x, x')/\rho\right)\right)^2}{\omega} \right);$$

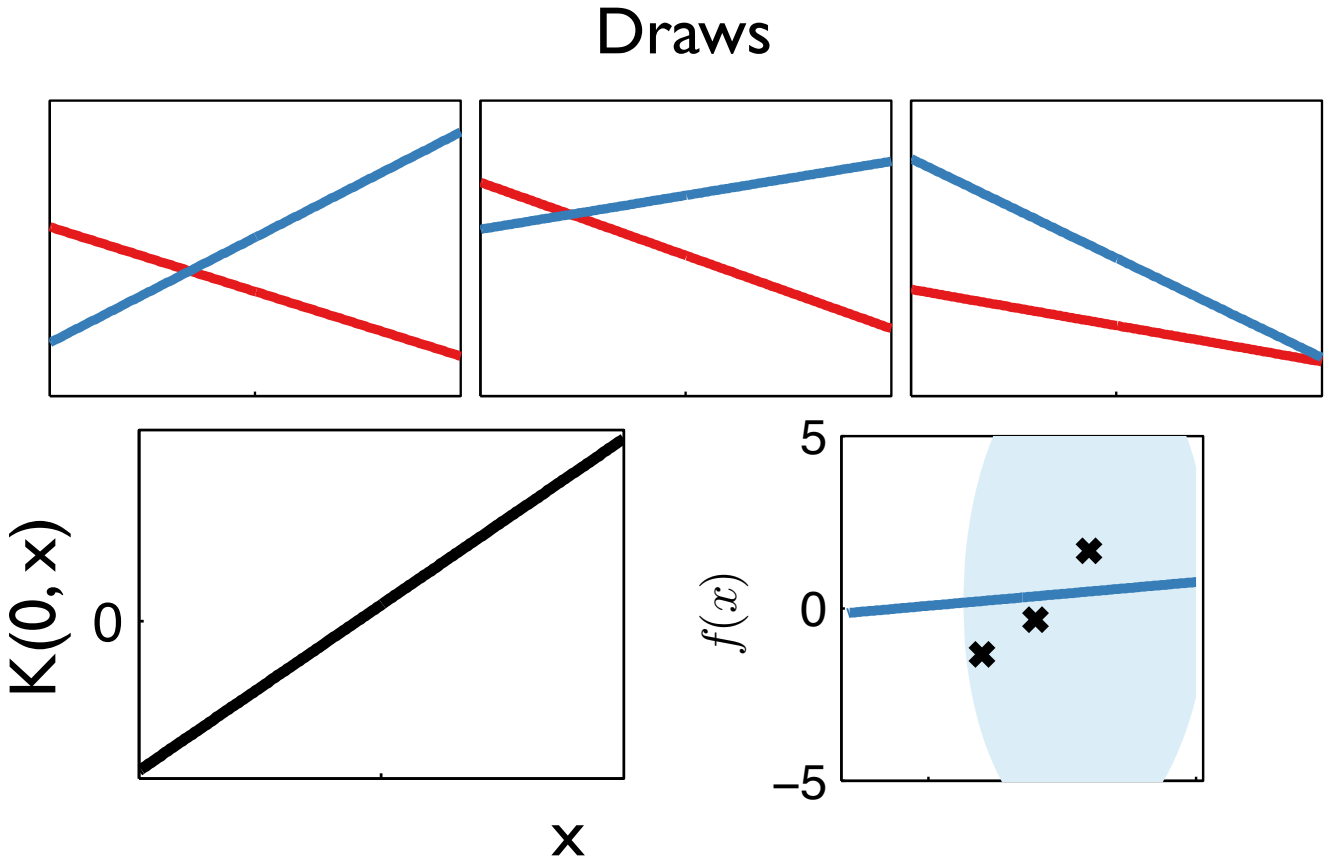
where $\rho$ is the period and $\omega$ controls the roughness.



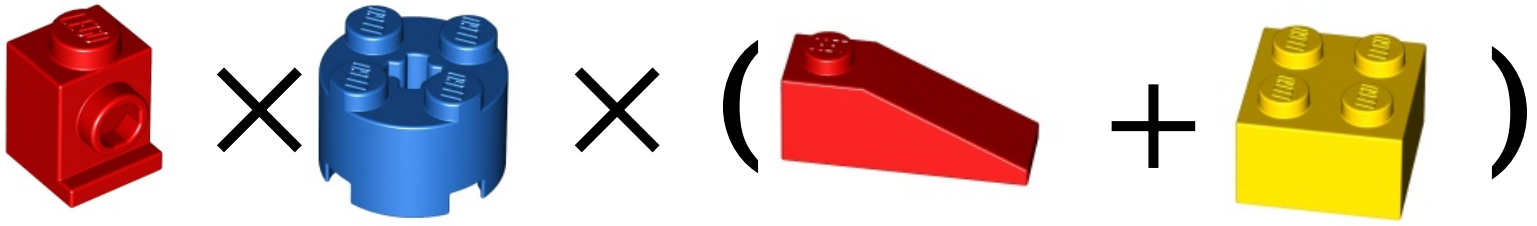
# The affine covariance exactly corresponds to an affine model with Gaussian priors:

$$K(x, x'; \theta) = \alpha^2 + \beta^2(x - \gamma)(x' - \gamma);$$

recall our derivation at the start of this topic.



# We can create new covariance functions by adding and/or multiplying other covariance functions.



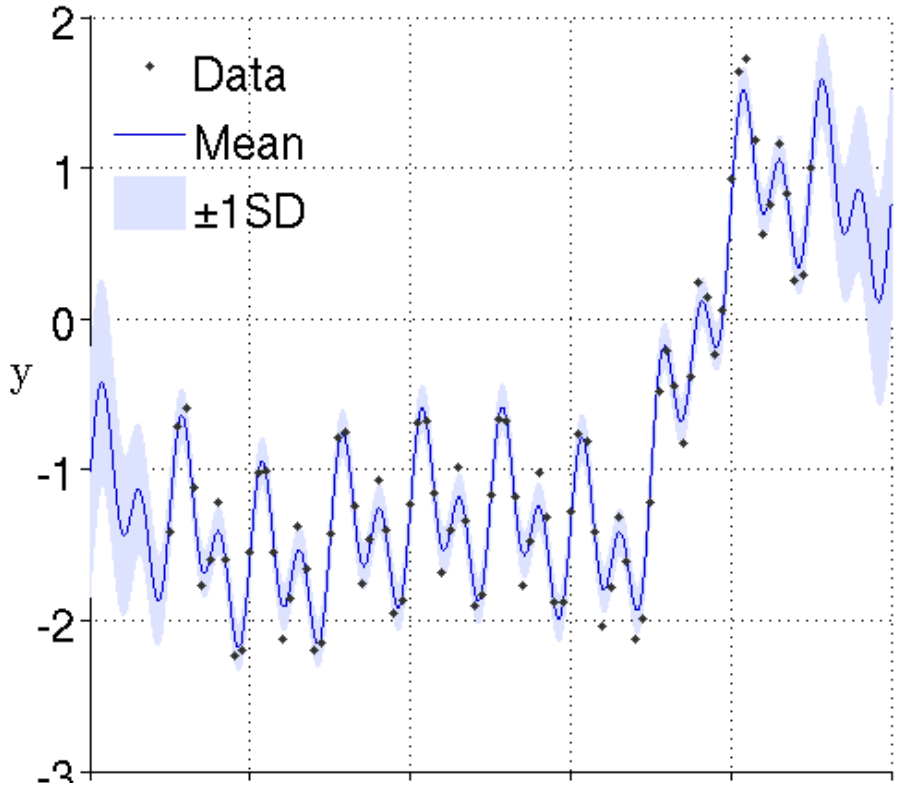
Multiplication (and) corresponds to covariance being dependent on similarity under both terms.

Addition (or) corresponds to covariance being dependent on similarity under either term.

# When a function is the sum of two independent functions:

use a covariance that is the sum of the covariances for those two functions,

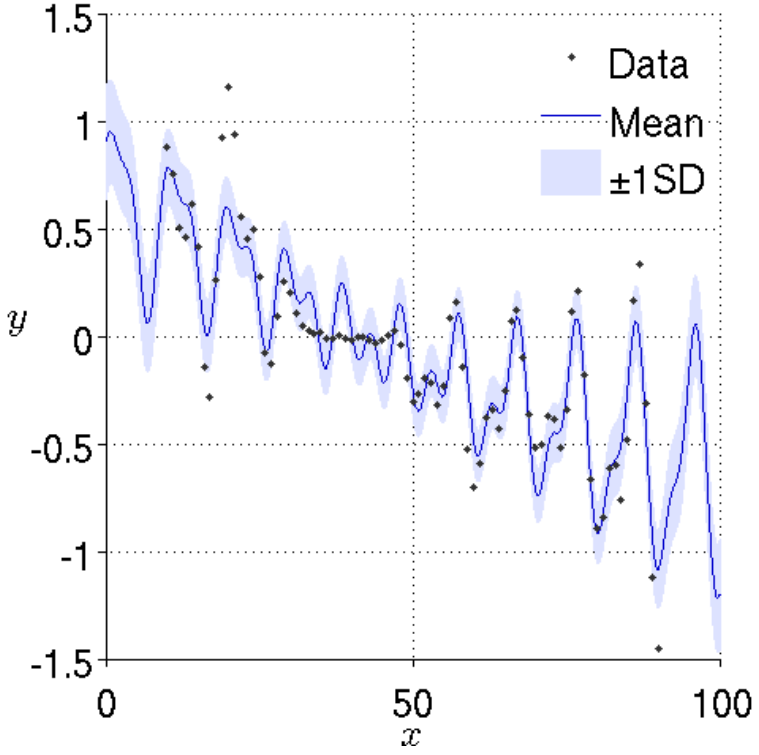
$$K_a(x, x') + K_b(x, x').$$



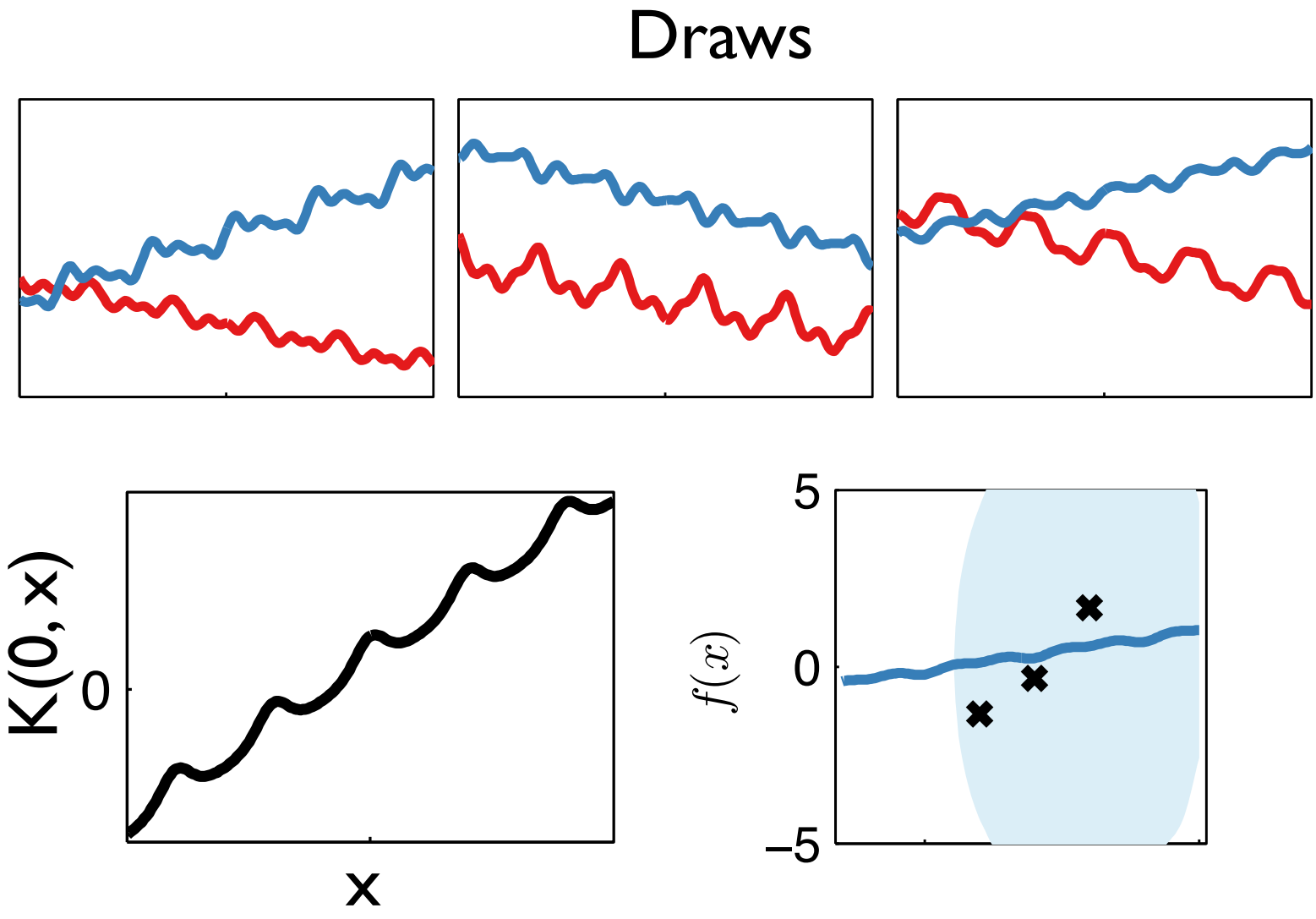
# When a function is the product of two independent functions:

use a covariance that is the product of the covariances for those two functions,

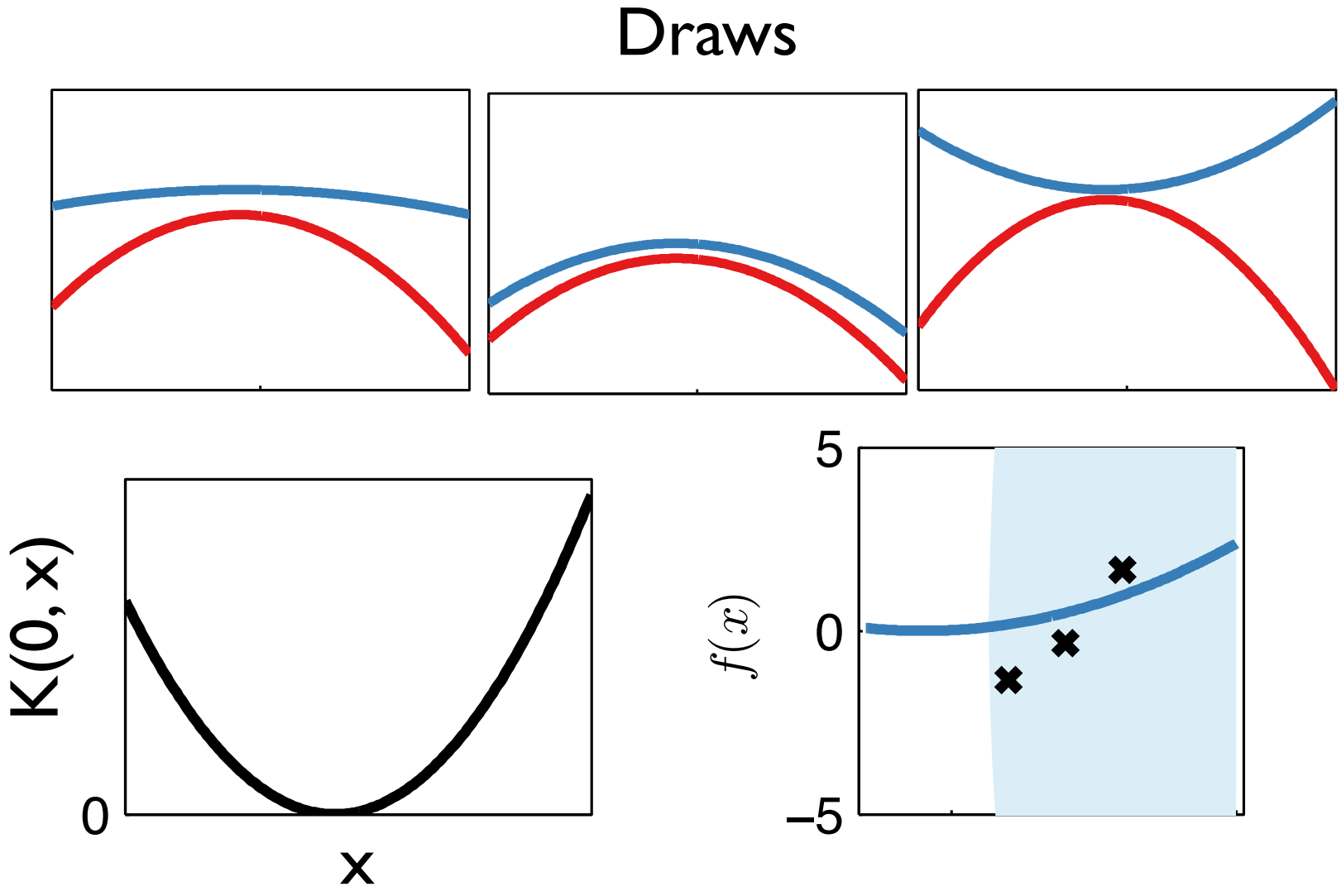
$$K_a(x, x')K_b(x, x') + K_a(x, x')\mu_b(x)\mu_b(x') + K_b(x, x')\mu_a(x)\mu_a(x').$$



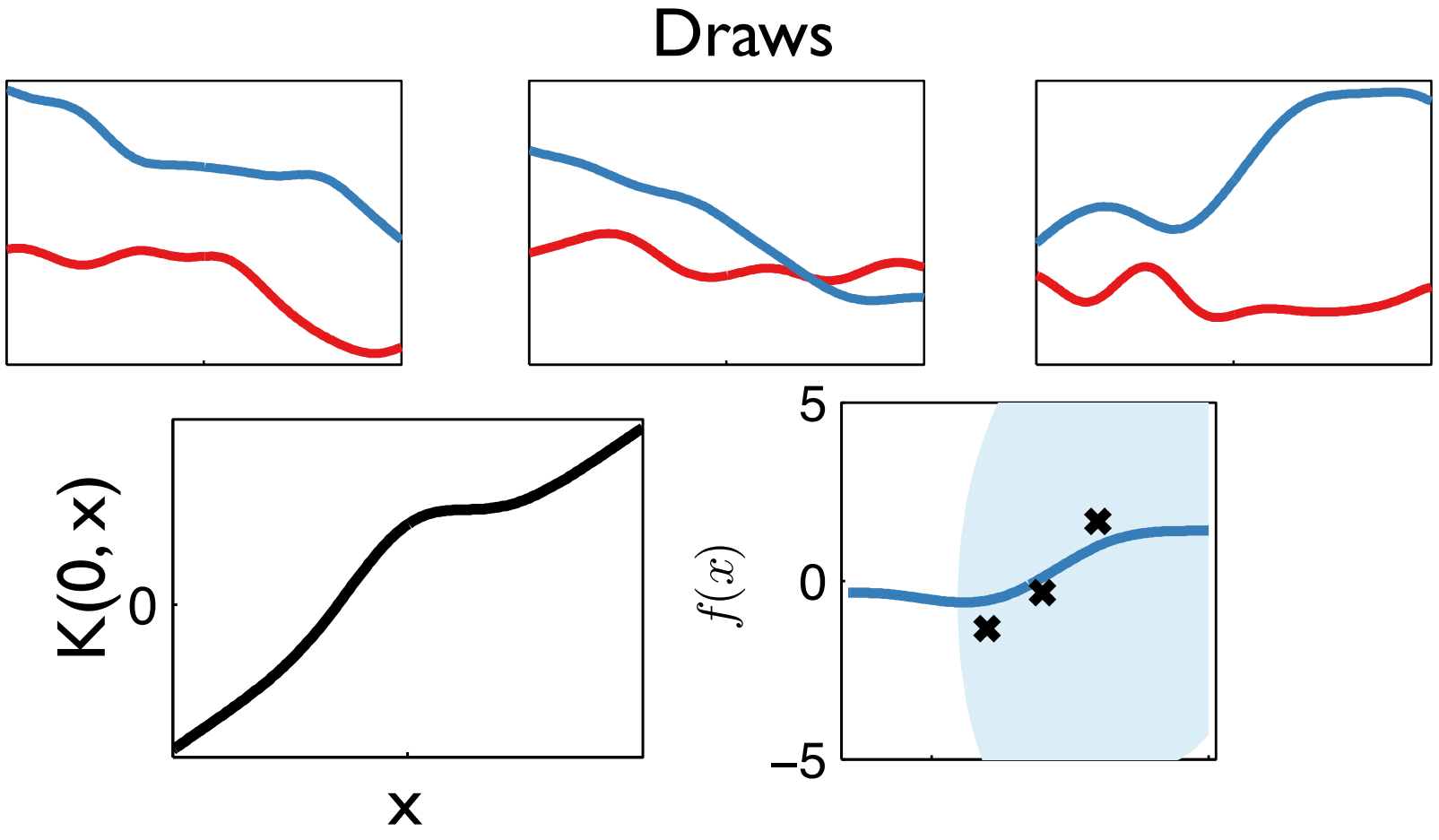
# The linear plus periodic covariance:



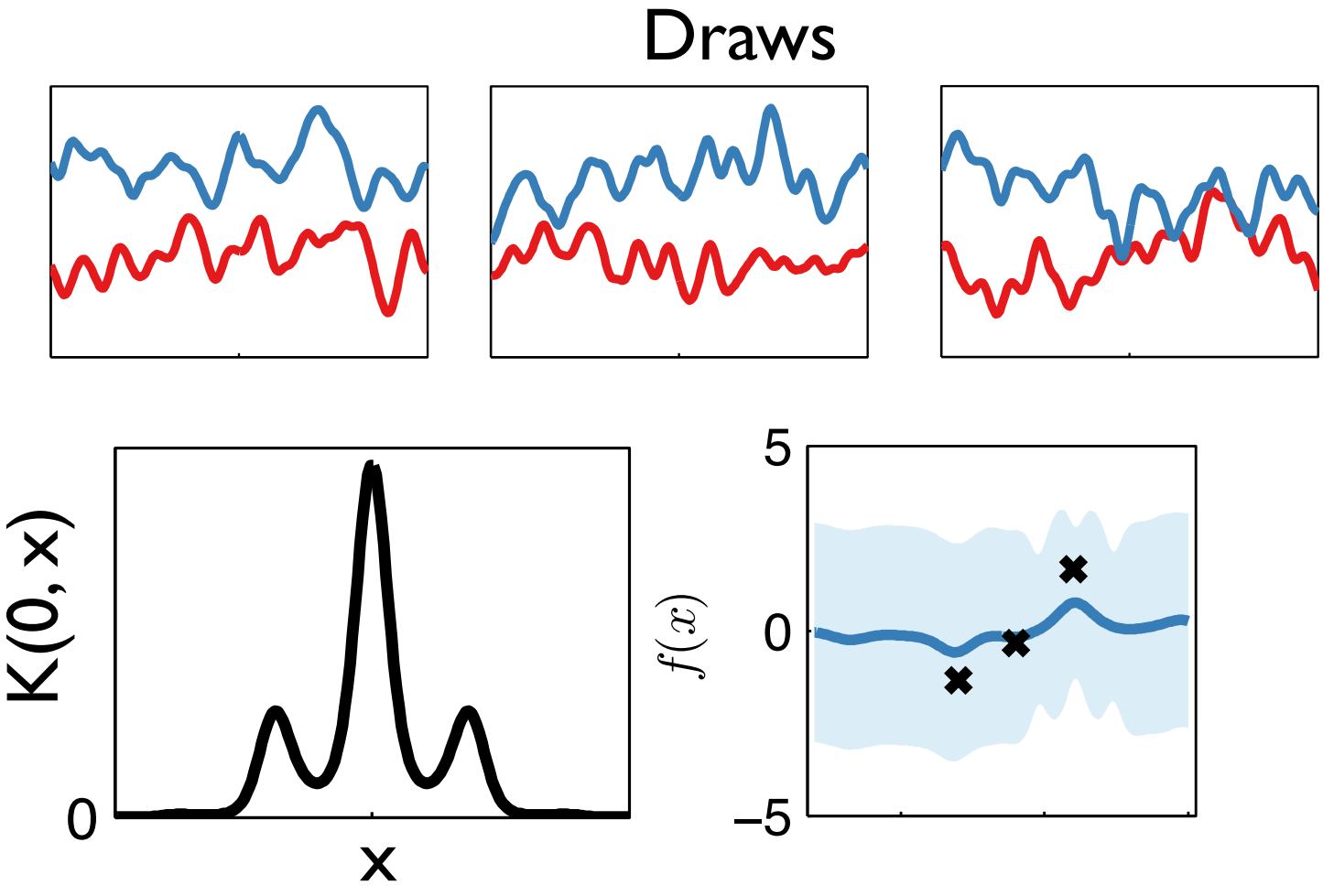
# The linear times linear covariance:



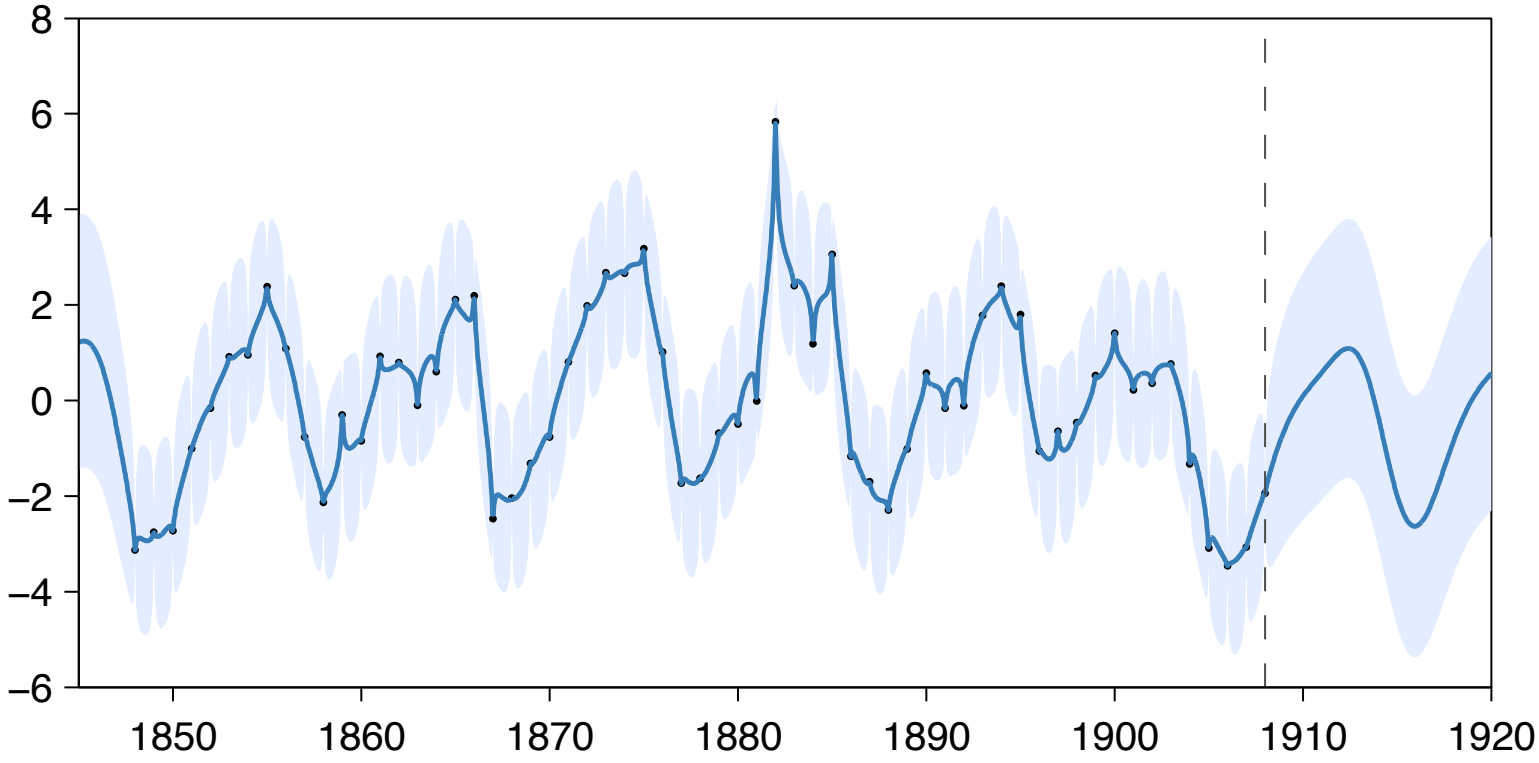
# The linear times exponentiated quadratic covariance:



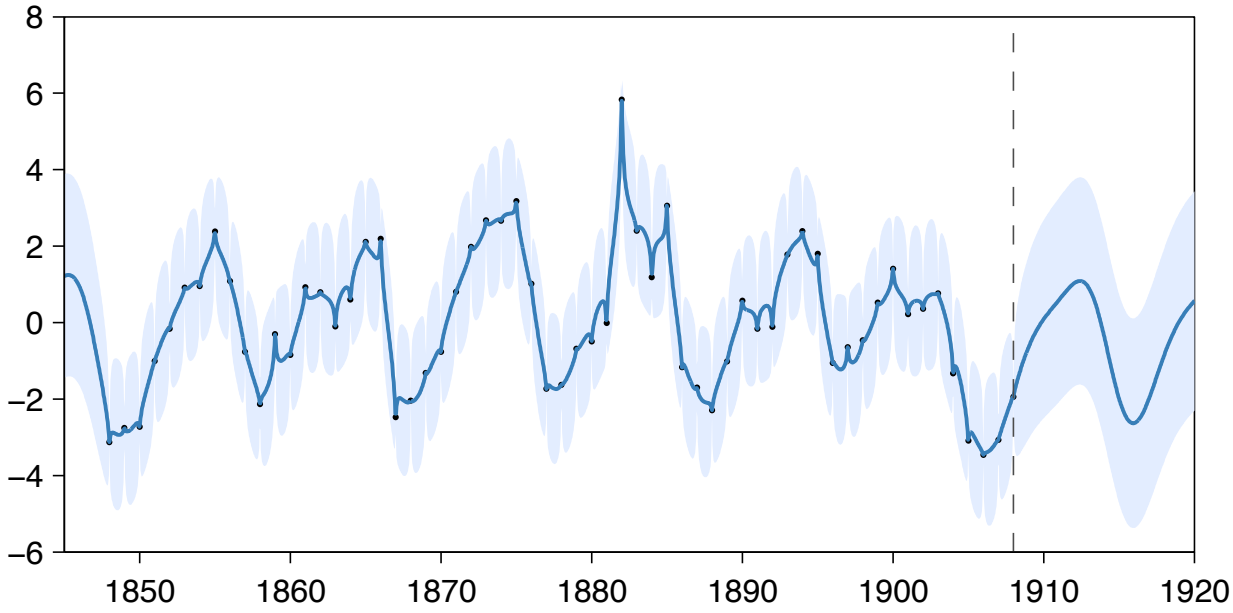
# The periodic times exponentiated quadratic covariance:



# Which covariance function might you use to model historical fur sales in the US?

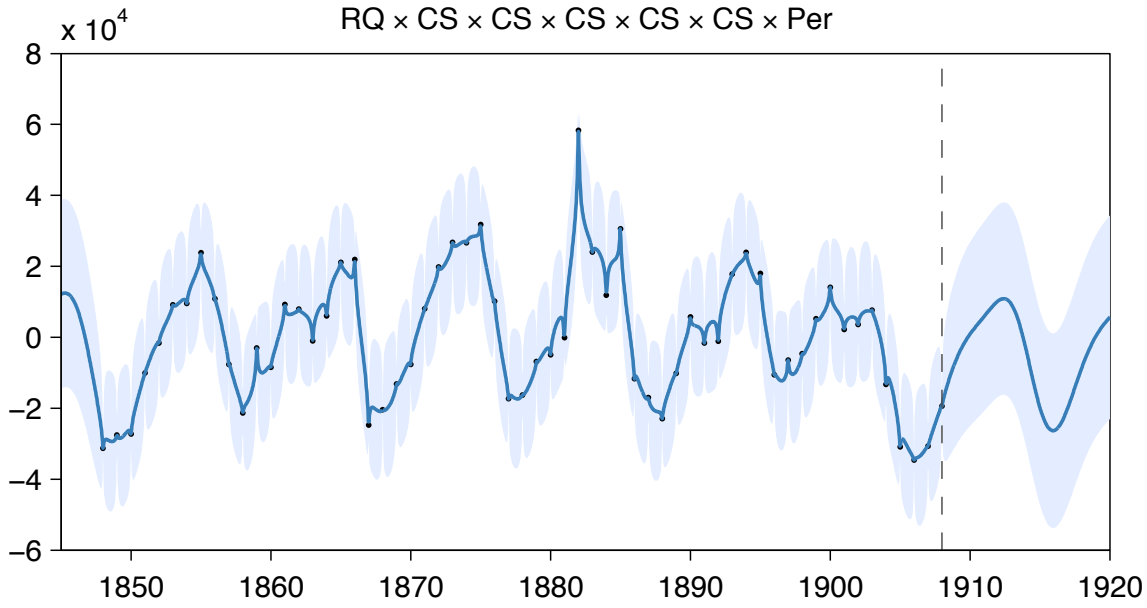



Plot courtesy of David Duvenaud.




Which covariance function should you choose?

1. Rational quadratic times periodic.
2. Exponentiated quadratic.
3. Exponentiated quadratic plus periodic.
4. Matérn, $\nu = \frac{1}{2}$.



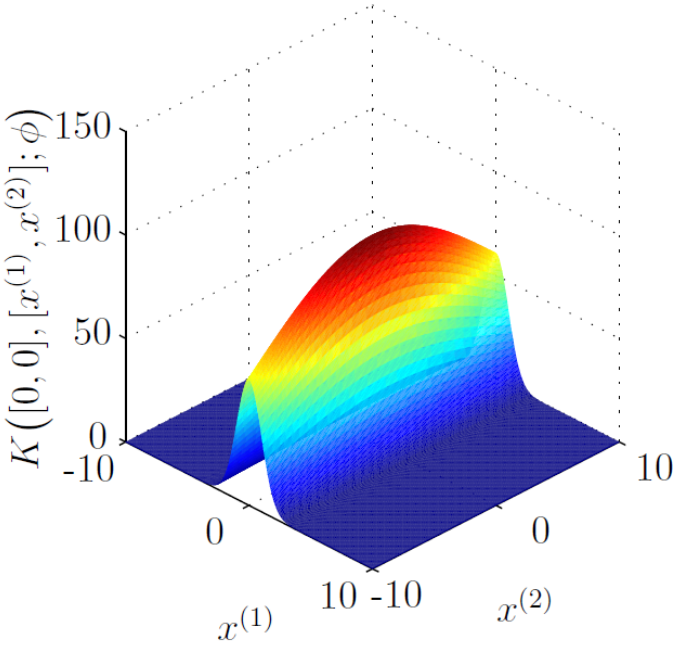

Which covariance function should you choose?
NB: there is no 'right' answer!

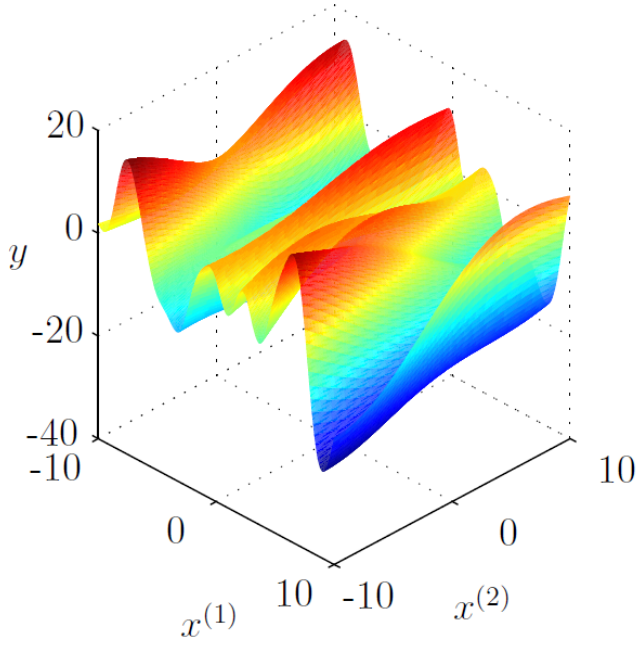
1. Rational quadratic times periodic.
2. Exponentiated quadratic.
3. Exponentiated quadratic plus periodic.
4. Matérn, $\nu = \frac{1}{2}$.

# We can create covariances for functions of many dimensions.

1. We can simply use the Euclidean distance in $\mathbb{R}^n$;
2. Alternatively, we can take a covariance that is the product or sum of covariances, one for each dimension:

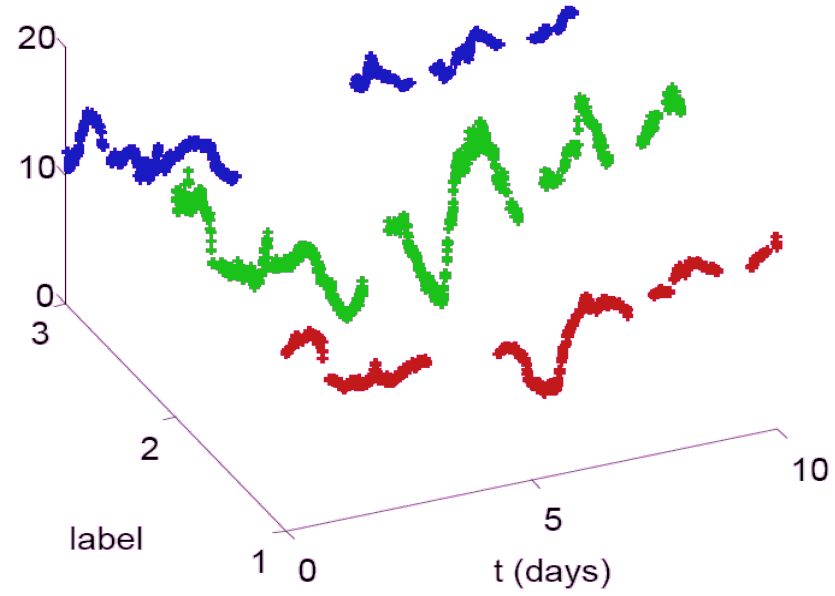
$$K_1(x_1, x_1')K_2(x_2, x_2') \quad \text{or} \quad K_1(x_1, x_1') + K_2(x_2, x_2').$$



# We tackle multiple outputs with co-regionalisation.

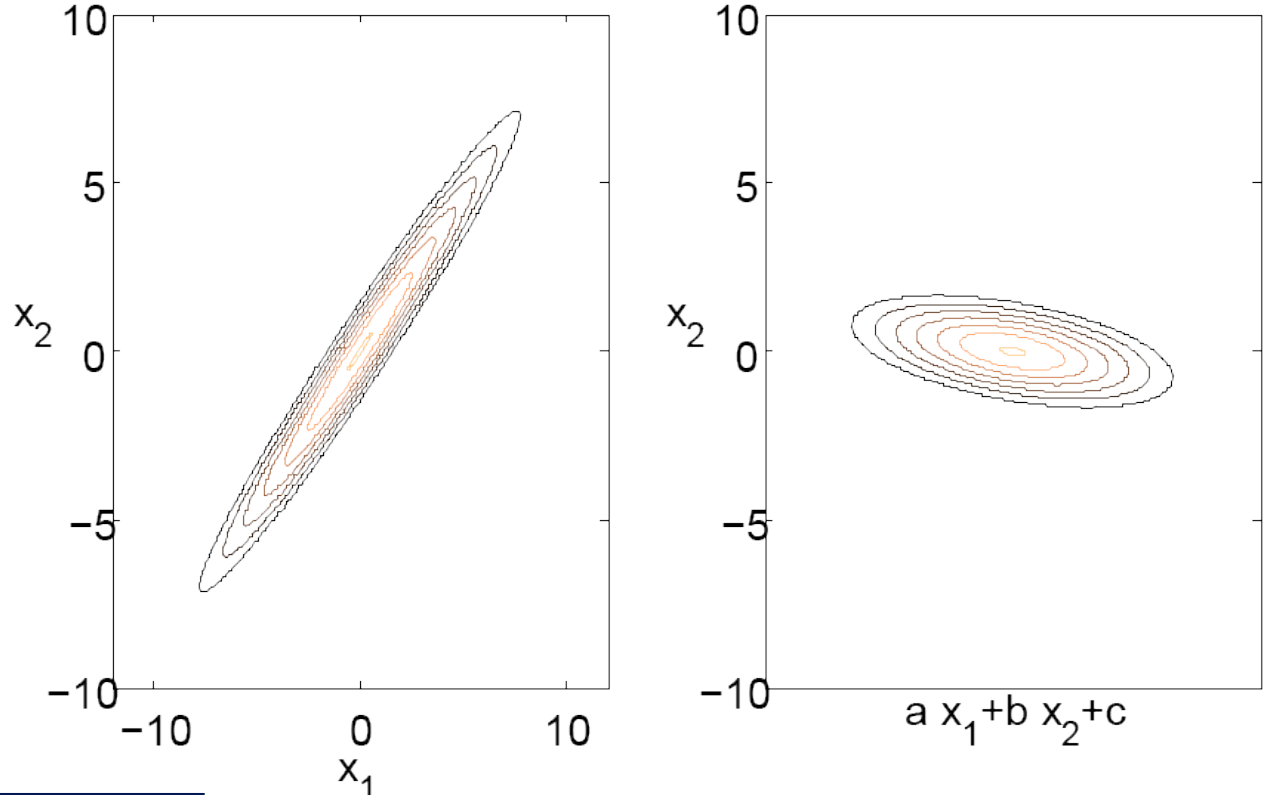
If the $l$th output is $f_l(x)$, rewrite as $g(l, x) = f_l(x)$. Then we can take a covariance for $g$ of the form $K_L(l, l')K_X(x, x')$.




Note that if there are $3$ outputs, $K_L$ is specified by a $3$ by $3$ positive semi-definite matrix.

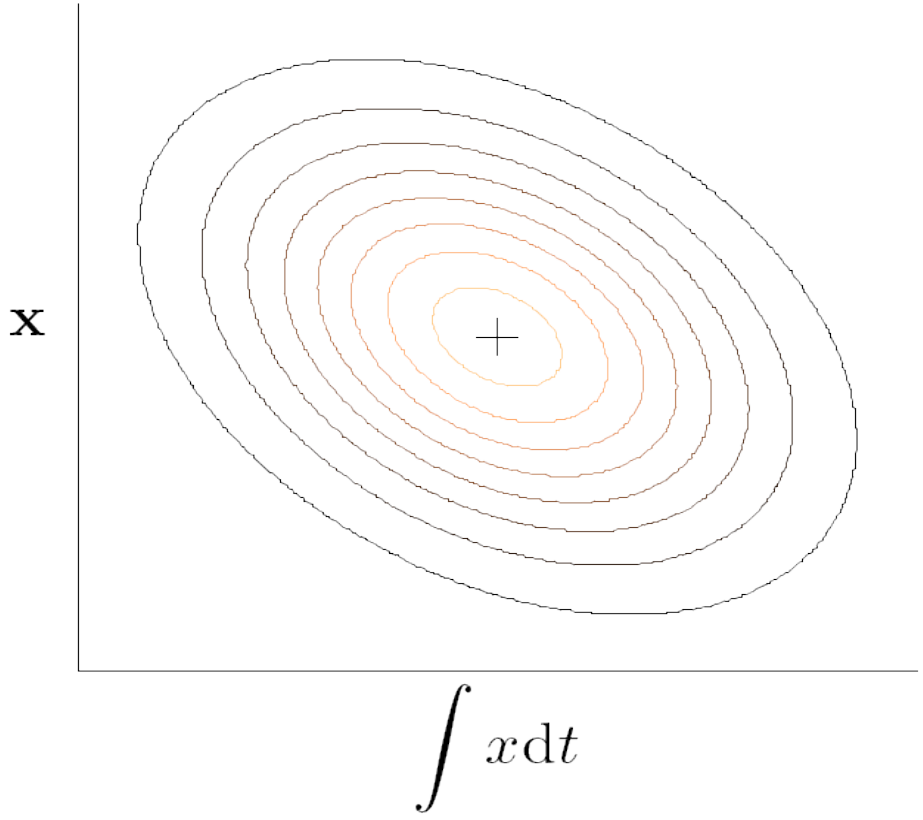
# Recall Gaussian distributed variables are joint Gaussian with any affine transform of them;

this includes rotations, scalings and translations.

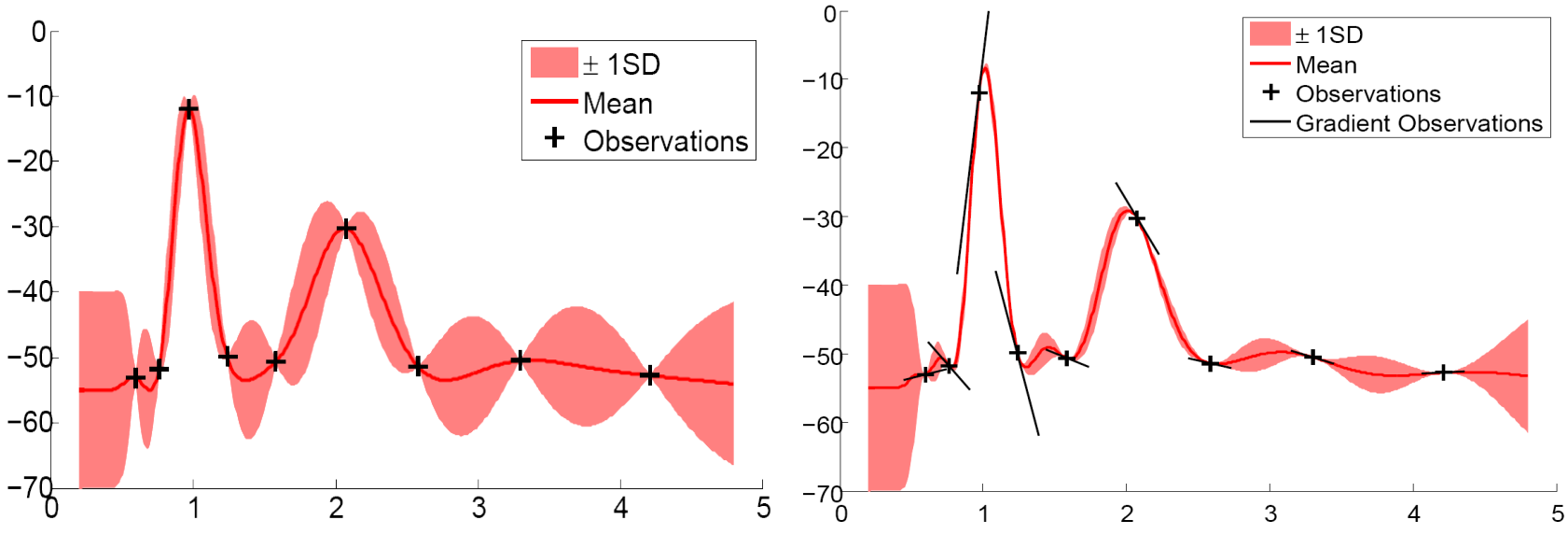



This means we can use observations of e.g. $ax_1 + bx_2 + c$ to infer e.g. $x_1$ in closed form.

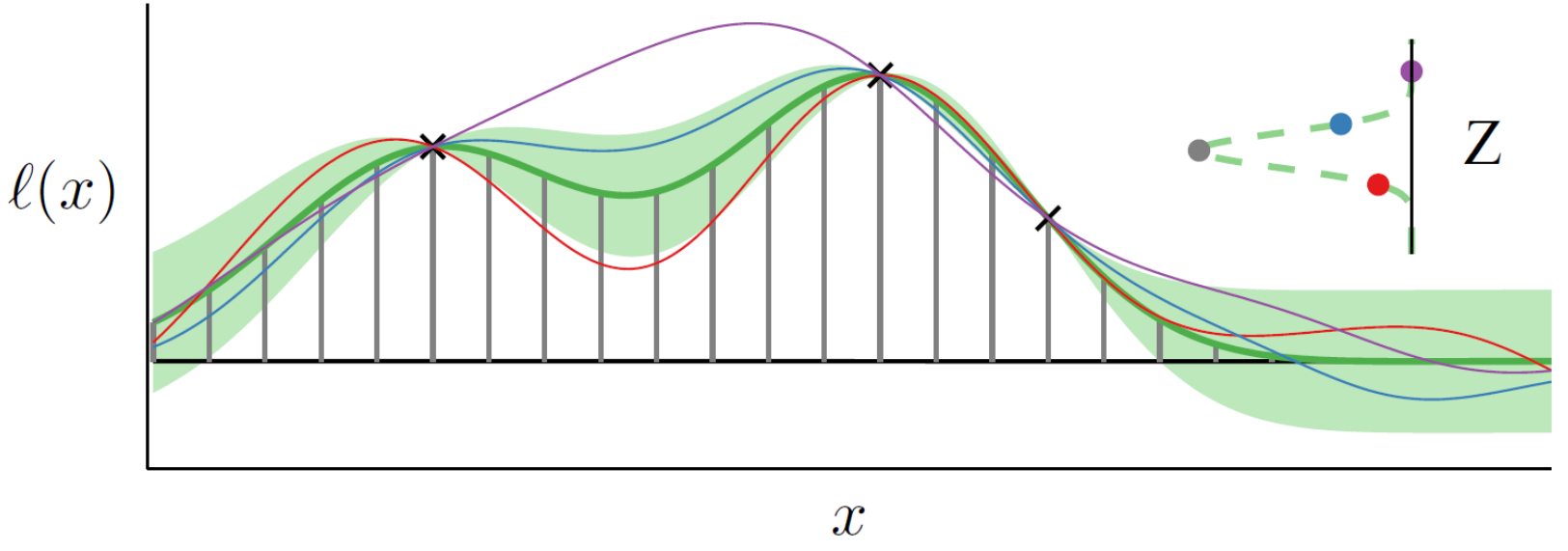
A function over which we have a Gaussian process is joint Gaussian with any integral or derivative of it, as integration and differentiation are affine.



# Hence we can include derivative observations into our Gaussian process.



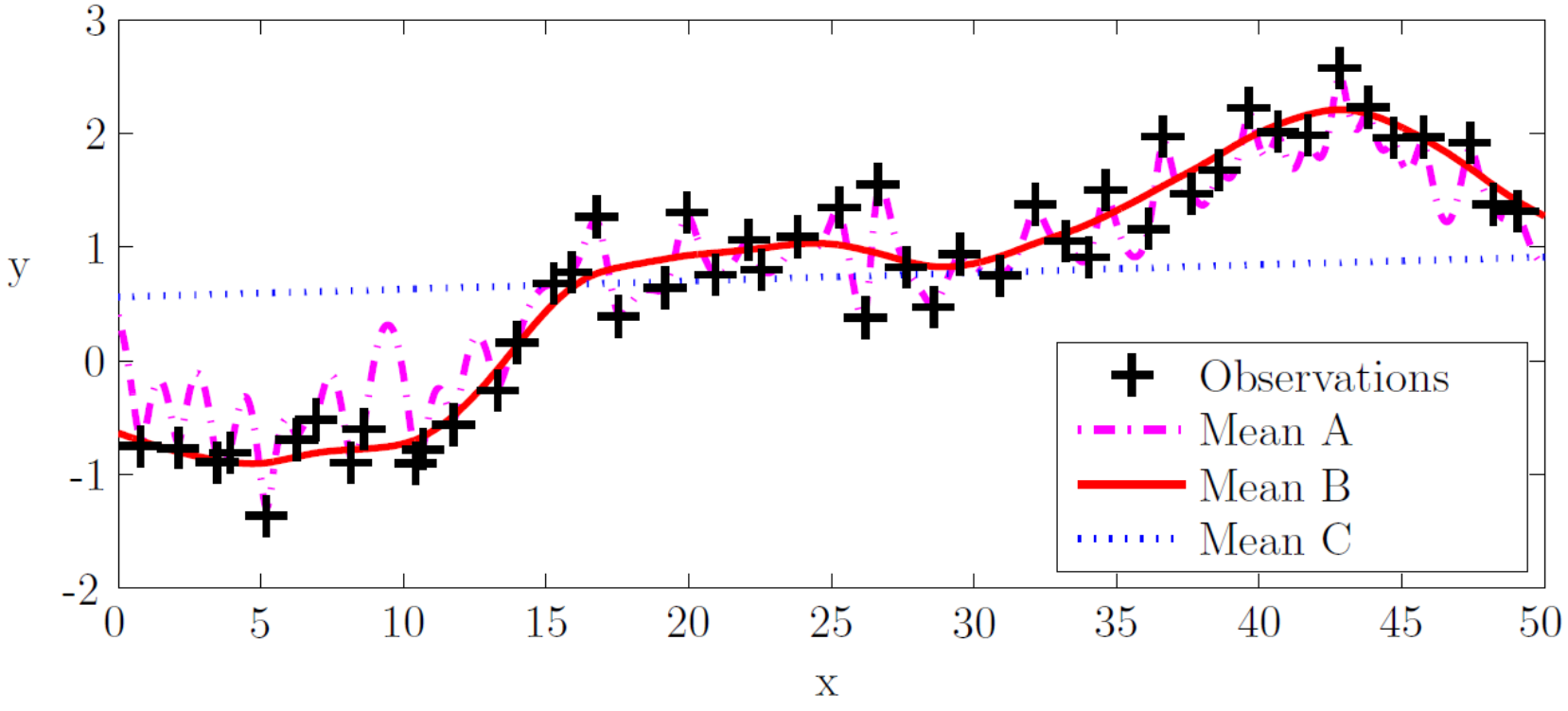
We can also use observations of an integrand $\ell$ to perform inference for its integral, $Z$: this is known as Bayesian quadrature.




UNIVERSITY OF OXFORD

# Hyperparameters can have a significant influence on inference.

The input scale for $A$ is short, longer for $B$ and longest for $C$.



# Maximum likelihood and MAP are popular approaches to setting hyperparameters.

Aren't maximum likelihood and MAP bad? Yes!

However, here we are using them for hyperparameters: our parameters (the function values) are properly marginalised.

The further up the hyper-chain you go, the less influence your approximations should have: let's justify the use of maximum likelihood for this setting.

# We need to approximate the usual ratio of integrals:

$$p(f_* \mid y_d) = \frac{\int p(f_* \mid y_d, \theta)\, p(y_d \mid \theta)\, p(\theta)\, \mathrm{d}\theta}{\int p(y_d \mid \theta)\, p(\theta)\, \mathrm{d}\theta},$$

where $p(f_* \mid y_d, \theta)$ are the predictions, $p(y_d \mid \theta)$ is the likelihood and $p(\theta)$ is the prior.

Maximum likelihood and maximum a-posteriori (MAP) approximate these integrals be approximating the likelihood ($p(y_d \mid \theta)$) and posterior (proportional to $p(y_d \mid \theta)\, p(\theta)$), respectively, as delta functions of $\theta$.

# Maximum likelihood and MAP are popular approaches to setting hyperparameters.

The GP log-likelihood is (where $n_d$ is the number of data $y_d$)

$$\log p(y_d \mid \theta) = \underbrace{-\frac{1}{2}\big(y_d - \mu(x_d)\big)^\top \big(K(x_d, x_d) + \sigma^2 I_d\big)^{-1}\big(y_d - \mu(x_d)\big)}_{\text{penalises mismatch between prior and data}}$$

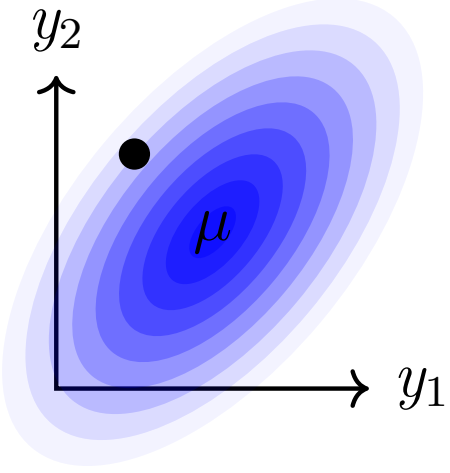
$$\underbrace{-\frac{1}{2}\log\det\big(K(x_d, x_d) + \sigma^2 I_d\big)}_{\text{penalises model complexity}} \underbrace{-\frac{n_d}{2}\log 2\pi}_{\text{constant}};$$

$\mu$ and $K$ are both functions of $\theta$. Note that many of our hyperparameters $\theta$ (e.g length scales $\omega$ and $\lambda$) are strictly positive: it is common to reparameterise to optimise over the logarithms e.g $\tilde{w} = \log \omega$.

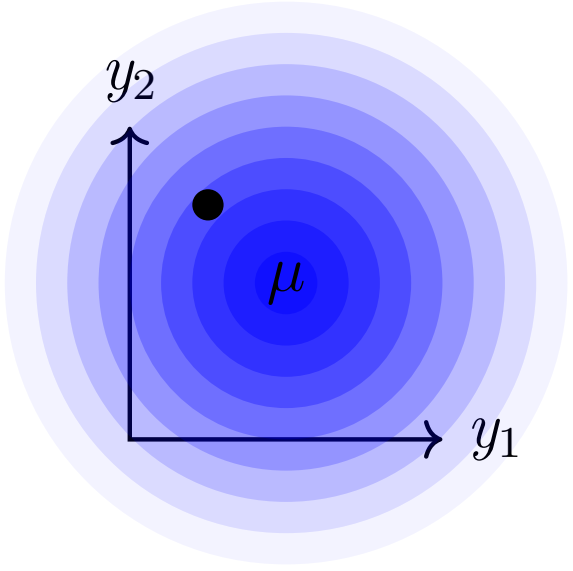
# The first term in the log-likelihood will deform the prior to match the data.

The first term is the log-density of (an un-normalised version of) the Gaussian prior evaluated at the data,

$$-\frac{1}{2}\big(y_d - \mu(x_d)\big)^\top \big(K(x_d, x_d) + \sigma^2 I_d\big)^{-1}\big(y_d - \mu(x_d)\big).$$




log-density low
$\Rightarrow$ likelihood term low




log-density high
$\Rightarrow$ likelihood term high

# The second term in the log-likelihood will penalise excessive complexity.

Let's write use the eigenvector decomposition

$$K(x_d, x_d) + \sigma^2 I_d = R\Lambda^2 R^\top$$

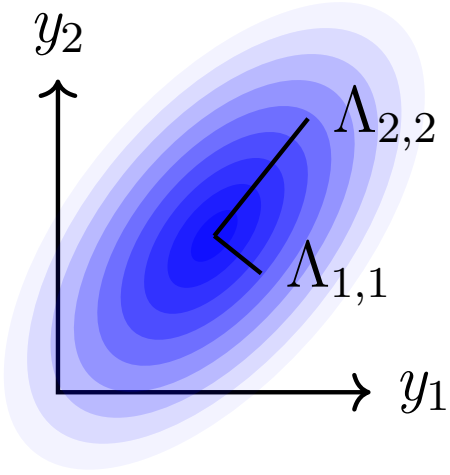
where $R$ is an orthogonal (rotation) matrix and $\Lambda$ is the diagonal eigenvalue matrix. The second term can be written as

$$-\frac{1}{2}\log \mathsf{det} K(x_d, x_d) + \sigma^2 I_d = -\frac{1}{2}\log \prod_i \Lambda_{i,i}.$$

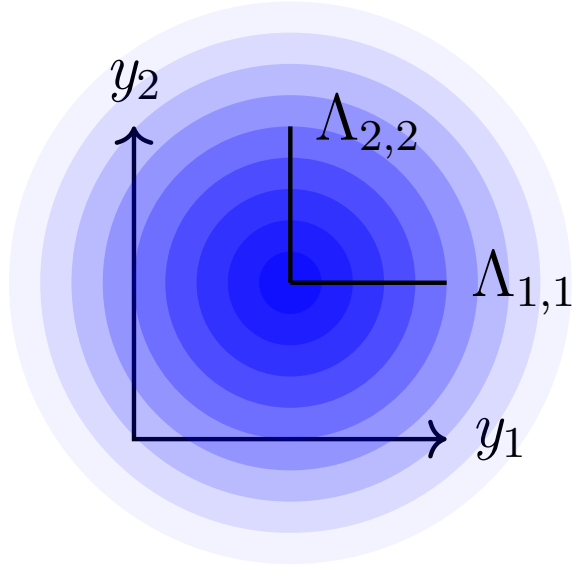
That is, this term is largest when the product of eigenvalues of the covariance matrix is smallest.

# The second term in the log-likelihood will penalise excessive complexity.

This term is largest when the product of eigenvalues of the covariance matrix is smallest.

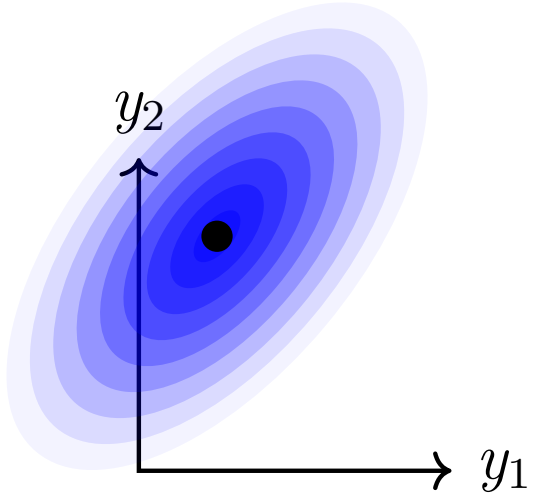



product of eigenvalues small
$\Rightarrow$ likelihood term high




product of eigenvalues large
$\Rightarrow$ likelihood term low

# Optimising the likelihood will push the prior mean $\mu$ to fit the data $y_d$ exactly.




The Gaussian process is bolted onto a parametric model: the prior mean. Doing maximum likelihood for a parametric model will lead to exactly this kind of over-fitting, as we know! We get around this in practice by taking only very simple prior mean functions that can't do much damage if we use maximum likelihood.

# The key computational bottleneck associated with Gaussian processes is resolving $K^{-1}v$.

Solving such equations crops up in many places:

$$m_{*|d} = \mu(x_*) + K(x_*, x_d)\big(K(x_d, x_d) + \sigma^2 I_d\big)^{-1}\big(y_d - \mu(x_d)\big);$$

$$C_{*|d} = K(x_*, x_*) - K(x_*, x_d)\big(K(x_d, x_d) + \sigma^2 I_d\big)^{-1} K(x_d, x_*);$$

$$\log p(y_d \mid \theta) = -\frac{1}{2}\big(y_d - \mu(x_d)\big)^\top \big(K(x_d, x_d) + \sigma^2 I_d\big)^{-1}\big(y_d - \mu(x_d)\big) - \frac{1}{2}\log \det(K(x_d, x_d) + \sigma^2 I_d) - \frac{n_d}{2}\log 2\pi.$$

# Note that computing $K^{-1}v$ is equivalent to solving $v = Ku$ for $u$.

You should never explicitly compute a matrix inverse.

Matrix inversion is $\mathcal{O}(n_d^3)$ for a matrix of size $n_d$.

Worse, matrix inversion is relatively unstable and susceptible to conditioning errors.

Conditioning issues emerge when some of your data is excessively self-similar, leading to high covariances.

$$\text{e.g.} \quad K(x_d, x_d) = \begin{bmatrix} K(x_1,x_1) & K(x_1,x_2) & K(x_1,x_3) & \cdots \\ K(x_2,x_1) & K(x_2,x_2) & K(x_2,x_3) & \cdots \\ K(x_3,x_1) & K(x_3,x_2) & K(x_3,x_3) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

$$= \begin{bmatrix} 1.00000 & 0.99999 & 0.99998 & \cdots \\ 0.99999 & 1.00000 & 0.99999 & \cdots \\ 0.99998 & 0.99999 & 1.00000 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

where the first two rows might be identical to four significant figures.

# Conditioning is ameliorated by noise.

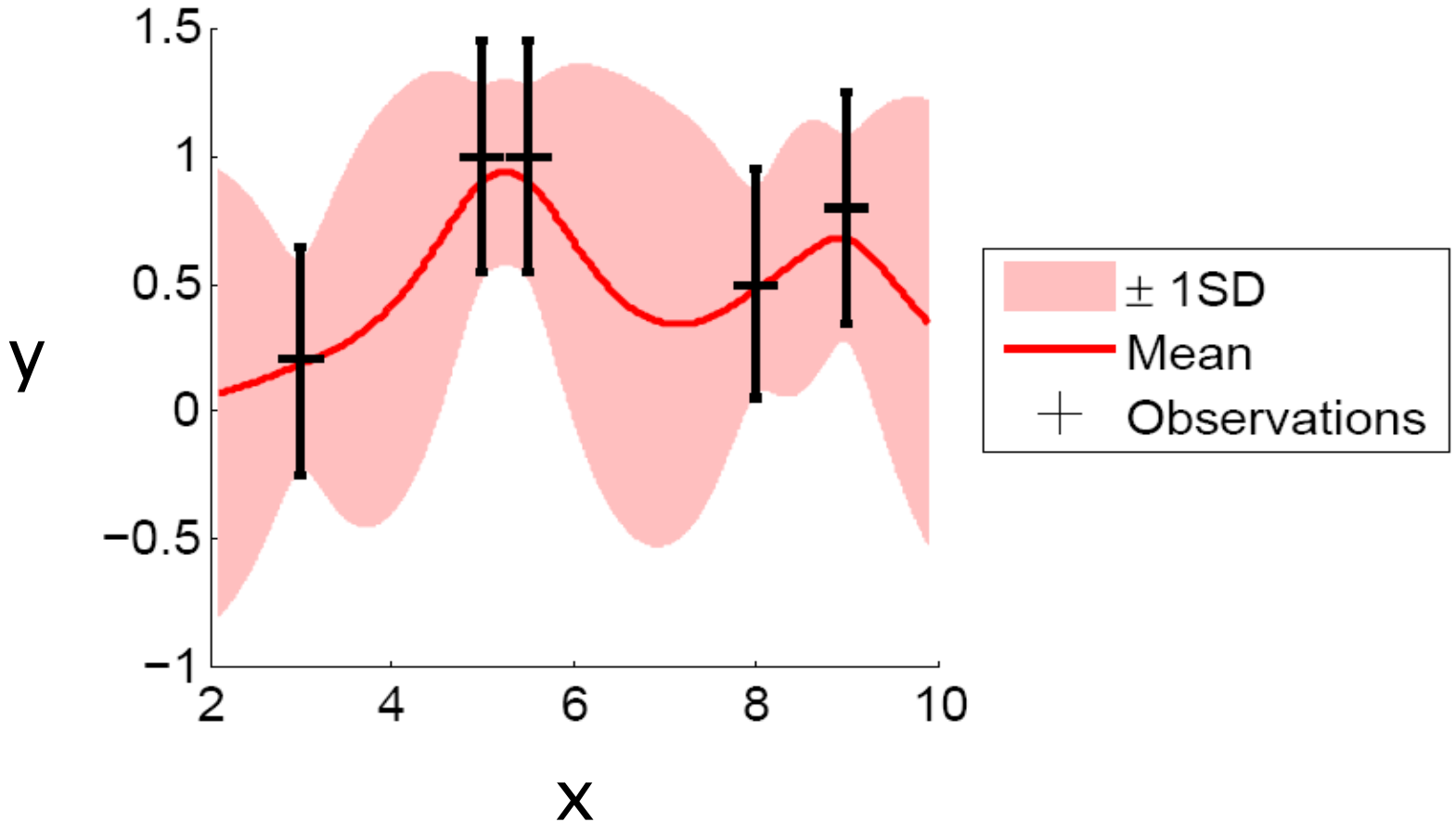
e.g. $K(x_d, x_d) + \sigma^2 I_d$

$$= \begin{bmatrix} K(x_1,x_1)+\sigma^2 & K(x_1,x_2) & K(x_1,x_3) & \cdots \\ K(x_2,x_1) & K(x_2,x_2)+\sigma^2 & K(x_2,x_3) & \cdots \\ K(x_3,x_1) & K(x_3,x_2) & K(x_3,x_3)+\sigma^2 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

$$= \begin{bmatrix} 1.00000+\sigma^2 & 0.99999 & 0.99998 & \cdots \\ 0.99999 & 1.00000+\sigma^2 & 0.99999 & \cdots \\ 0.99998 & 0.99999 & 1.00000+\sigma^2 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

For this reason, we often assume noise where there isn't, or add on a small amount to $\sigma$ (known as jitter), so as to combat conditioning.

As jitter is, essentially, artificial noise, it artificially dilutes the informativeness of data.



The Cholesky factorisation of a positive semi-definite matrix is faster than inversion ($1/3\mathcal{O}(n_d^3)$) and is more numerically stable.

$$K(x_d, x_d) + \sigma^2 I_d = R^\top R$$

where

$$R = \text{chol}\Big(K(x_d, x_d) + \sigma^2 I_d\Big)$$
$$= \begin{bmatrix} R_{11} & R_{12} & R_{13} & \cdots \\ 0 & R_{22} & R_{23} & \cdots \\ 0 & 0 & R_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

# The Cholesky factor $R$ can be used to solve $v = Ku$ using $\mathcal{O}(n_d^2)$ back-substitution.

Given $v$ and $R$ (where $K = R^\top R$), we find $u$ as

$$v = Ku$$

$$\Rightarrow v = R^\top u' \quad \text{and} \quad u' = Ru$$

$$\Rightarrow \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \end{bmatrix} = \begin{bmatrix} R_{11} & 0 & 0 & \cdots \\ R_{12} & R_{22} & 0 & \cdots \\ R_{13} & R_{23} & R_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} u'_1 \\ u'_2 \\ u'_3 \\ \vdots \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} u'_1 \\ u'_2 \\ u'_3 \\ \vdots \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & \cdots \\ 0 & R_{22} & R_{23} & \cdots \\ 0 & 0 & R_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \end{bmatrix}.$$

# The Cholesky factor can also be used to compute the determinant.

Recall

$$\log p(y_d \mid \theta) = -\frac{1}{2}\big(y_d - \mu(x_d)\big)^\top \big(K(x_d, x_d) + \sigma^2 I_d\big)^{-1}\big(y_d - \mu(x_d)\big) - \frac{1}{2}\log \det\big(K(x_d, x_d) + \sigma^2 I_d\big) - \frac{n_d}{2}\log 2\pi.$$

Now
$$\begin{aligned}\log \det R^\top R &= \log(\det R^\top \det R)\\ &= \log(\det R \det R)\\ &= 2\log \det R\\ &= 2\log \prod_i R_{ii}\\ &= 2\sum_i \log R_{ii}.\end{aligned}$$

# In summary,

1. The Gaussian process is a flexible, non-parametric, distribution for functions.
2. The mean function controls extrapolation.
3. The covariance function encodes structure about the function.
4. Covariance functions can be combined and modified.
5. Maximum likelihood can be used for Gaussian process hyperparameters.
6. Cholesky factorisation helps with the computational challenges for Gaussian processes.