Model Selection

Mark van der Wilk

Department of Computing
Imperial College London

@markvanderwilk
m.vdwilk@imperial.ac.uk

January 30, 2023

A Note on Notation

In Gaussian processes, we are building a **conditional** model of the data, with PoE:

$$p(\mathbf{y}, f(X), \mathbf{y}^*, f(X^*)|X, X^*) = \left[\prod_{n=1}^{N} p(y_n|f(\mathbf{x}_n), \mathbf{x}_n)\right]\left[\prod_{t=1}^{T} p(y_t^*|f(\mathbf{x}_t^*), \mathbf{x}_n^*)\right]$$
$$p(f(X), f(X)|X, X^*)$$

With the additional property that

$$\begin{aligned}
p(f(X), f(X^*)|X, X^*) &\overset{\text{AT}}{=} p(f(X^*)|f(X), X, X^*)p(f(X)|X, X^*) \\
&\overset{\text{AT}}{=} p(f(X)|f(X^*), X, X^*)p(f(X^*)|X, X^*) \\
&\overset{\text{MA}}{=} p(f(X^*)|f(X), X, X^*)p(f(X)|X) \\
&\overset{\text{MA}}{=} p(f(X)|f(X^*), X, X^*)p(f(X^*)|X^*)
\end{aligned}$$

(You can prove this by finding the marginal of $p(f(X), f(X^*)))$

A Note on Notation

- You are expected to be able to derive these things, if necessary (see exercise in question sheet)
- However, for conditional models, we can simplify notation.
- ⏩ If not otherwise specified, for PoEs specified conditionally, you may drop what is conditioned on:

$$p(z, x|w, y) = p(x|z, w, y)p(z|w, y) \tag{1}$$

- Since we care mostly about the interaction between observed an unobserved quantities.

For GPs:

$$p(\mathbf{y}, f(X), \mathbf{y}^*, f(X^*)) = \left[\prod_{n=1}^{N} p(y_n|f(\mathbf{x}_n))\right]\left[\prod_{t=1}^{T} p(y_t^*|f(\mathbf{x}_t^*))\right] p(f(X), f(X^*))$$

Learning objectives

How to select the right prior assumptions

- What makes a valid kernel?
- Influence of a kernel on the GP prior.
- Influence of the GP prior on the posterior.
- Bayes' rule for inferring hyperparameters.
- The maximum a-posteriori approximation (MAP).
- Some practical issues.

Kernels

We constructed two kernels from inner products:

- $k(x, y) = (xy + 1)^{M-1} = \sum_{m=0}^{M-1} \binom{M-1}{m} x^m y^m = \boldsymbol{\phi}(x)^\top \boldsymbol{\phi}(y)$
- $k(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{(\mathbf{x}-\mathbf{y})^2}{2\ell^2}\right) = \lim_{M\to\infty} \boldsymbol{\phi}_M(\mathbf{x})^\top \boldsymbol{\phi}_M(\mathbf{y})$

Property: Kernels constructed from inner products are **positive-(semi)definite** functions, i.e. for any set of input points $\boldsymbol{X}$ we have:

$$\mathbf{v}^\top k(\boldsymbol{X}, \boldsymbol{X})\mathbf{v} = \sum_i \sum_j v_i k(\mathbf{x}_i, \mathbf{x}_j) v_j \geqslant 0 \tag{2}$$

Remember: $[k(\boldsymbol{X}, \boldsymbol{Z})]_{ij} = k(\mathbf{x}_i, \mathbf{z}_j)$, where $\boldsymbol{X}$ and $\boldsymbol{Z}$ are stacked vectors $\{\mathbf{x}_i\}$ and $\{\mathbf{z}_i\}$.

Proof: We constructed the kernel as $k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\top \phi(\mathbf{z})$, so:

$$\sum_{ij} v_i \phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j) v_j = \sum_i \boldsymbol{\alpha}_i^\top \sum_j \boldsymbol{\alpha}_j = \boldsymbol{\beta}^\top \boldsymbol{\beta} \geqslant 0 \tag{3}$$

Mercer's theorem proves converse.

Using any positive semidefinite function as a covariance function for Gaussian distributions gives a valid GP (see Kolmogorov extension theorem).

Properties of Kernels

For PSD kernels k, k_1, k_2 we have

$$k(\mathbf{x}, \mathbf{x}) \geqslant 0 \qquad \text{Take single point.} \tag{4}$$

$$k(\mathbf{x}, \mathbf{x}')^2 \leqslant k(\mathbf{x}, \mathbf{x}) k(\mathbf{x}', \mathbf{x}') \qquad \text{Cauchy-Schwarz} \tag{5}$$

$$\mathbf{v}^\top (k_1(\boldsymbol{X}, \boldsymbol{X}) + k_2(\boldsymbol{X}, \boldsymbol{X}))\mathbf{v} \geqslant 0 \qquad \text{i.e. } k_1 + k_2 \text{ is kernel} \tag{6}$$

$$\mathbf{v}^\top (k_1(\boldsymbol{X}, \boldsymbol{X}) \circ k_2(\boldsymbol{X}, \boldsymbol{X}))\mathbf{v} \geqslant 0 \qquad \text{i.e. } k_1 \cdot k_2 \text{ is kernel} \tag{7}$$

Also:

- $k(h(\mathbf{x}), h(\mathbf{x}'))$ is a kernel for a deterministic function $h(\cdot)$.
- $h(\mathbf{x})k(\mathbf{x}, \mathbf{x}')h(\mathbf{x}')$ is a kernel for deterministic function $h(\cdot)$.

Effect of kernel on GP prior

See Jupyter notebook `kernel-zoo.ipynb`.

Goal: Predict at new points

Remember our goal:

Use training set
to make good predictions at **new unseen inputs**.

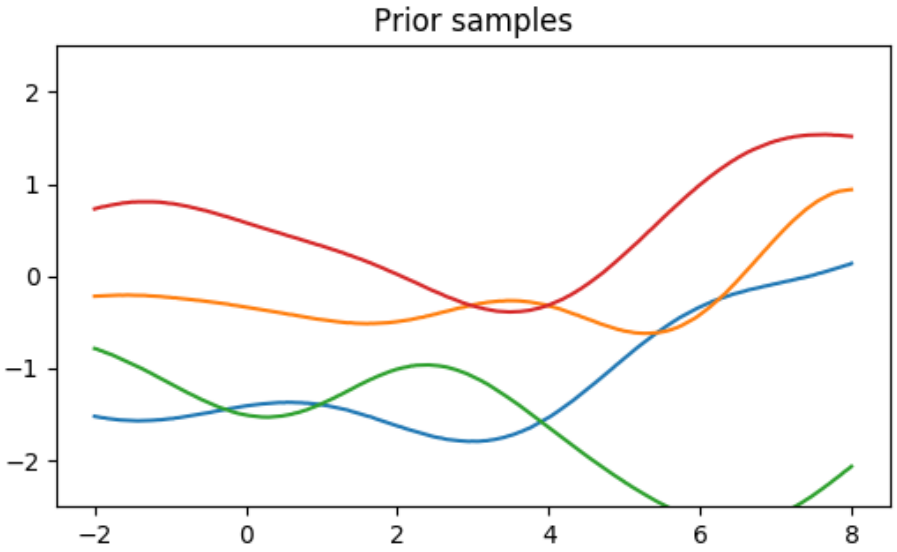
Measure generalisation accuracy using **log predictive density**, i.e. the predictive density evaluated at a point in the test set. This estimates the accuracy on future data drawn from the same distribution.

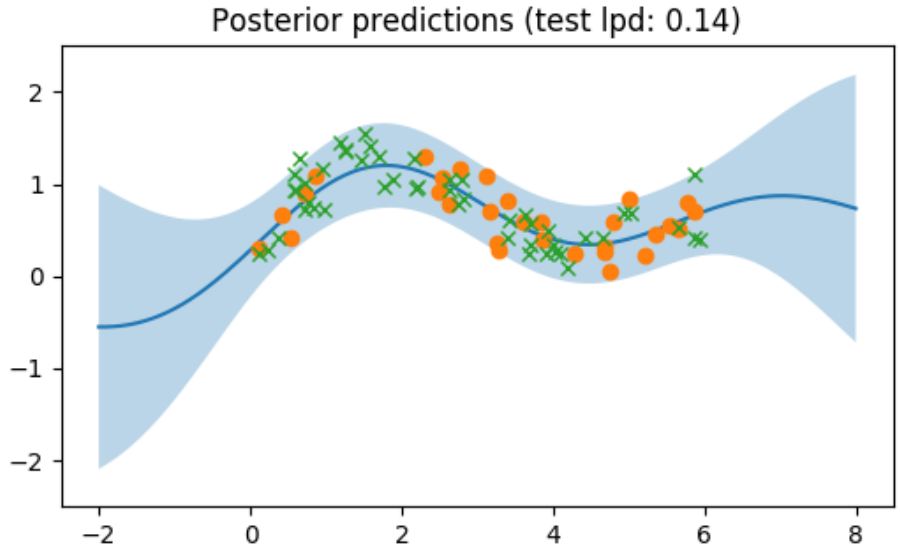
$$\text{lpd} = \sum_{n=1}^{N_t} \log p(y_n^* \mid \mathbf{x}_n^*, X, \mathbf{y}, \theta)\,, \qquad \text{for test set } \{\mathbf{x}_n^*, y_n^*\}_{n=1}^{N_t} \tag{8}$$

$$p(y_n^* \mid \mathbf{x}_n^*, X, \mathbf{y}, \theta) = \int \underbrace{p(y_n^* \mid f(\mathbf{x}_n^*), \mathbf{x}_n^*, \theta)}_{\text{likelihood}}\, p(f(\mathbf{x}_n^*) \mid X, \mathbf{x}_n^*, \mathbf{y}, \theta)\, \mathrm{d}f(\mathbf{x}_n^*) \tag{9}$$

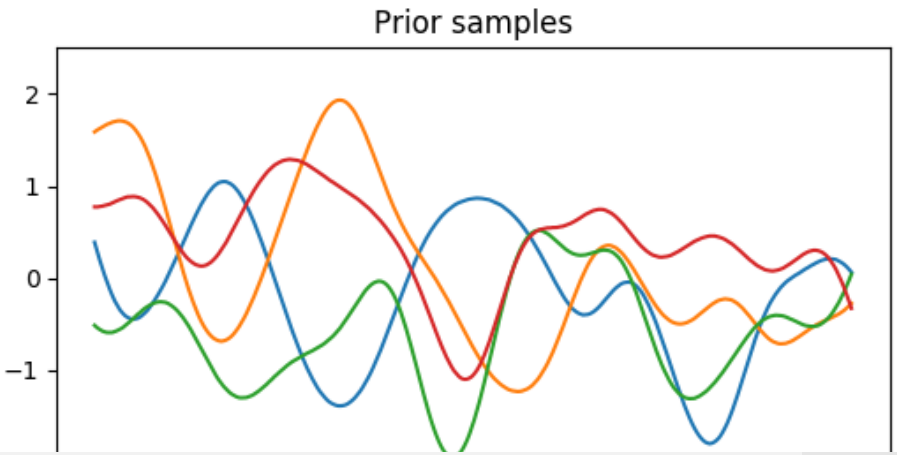
Influence of prior on posterior

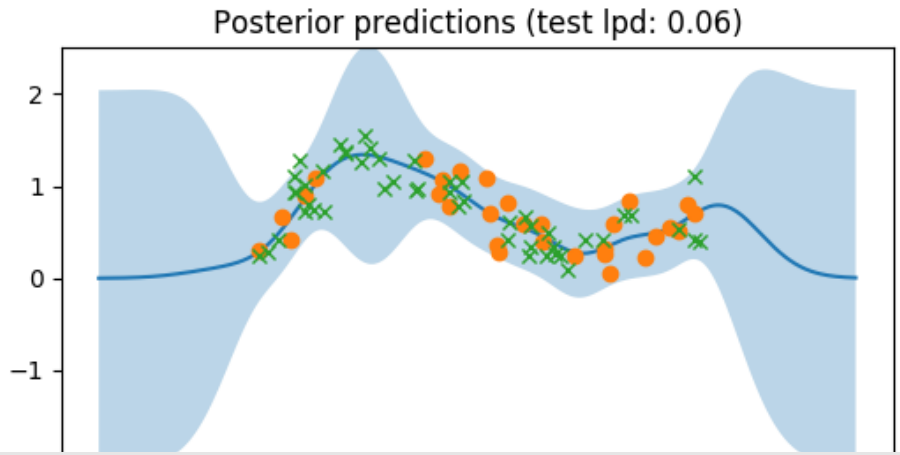
Dataset 1:





Dataset 1:





What is model selection

- Given a prior, we can make predictions with uncertainty.
- Different priors make different predictions of different quality.
- Different tasks need different priors.

How do we select the right prior for the task?

→ Model selection

Bayesian approach

Let's follow the Bayesian approach.

Hyperparameters are simply
yet another **unobserved quantity**
which we can infer with **Bayes' rule**.

$$p(f_{X,\mathbf{x}_*}|\mathbf{y},\theta) = \frac{p(\mathbf{y}, f_{X,\mathbf{x}_*}|\theta)}{p(\mathbf{y}|\theta)} = \frac{p(\mathbf{y}|f_X,\theta)p(f_{X,\mathbf{x}_*}|\theta)}{p(\mathbf{y}|\theta)} \tag{10}$$

- I use $f_{X,\mathbf{x}_*}$ as shorthand for $\begin{bmatrix} f(\boldsymbol{X})^\intercal & f(\mathbf{x}_*) \end{bmatrix}^\intercal \in \mathbb{R}^{N+1}$.
- Here, I drop the conditioning on the inputs.
- If *explicitly asked* on an exam, you must be able to correctly specify what inputs a distribution depends on.

Bayes for hyperparameters

Bayes' rule for everything:

$$p(f_{X,\mathbf{x}_*}, \theta \,|\, \mathbf{y}) = \frac{p(\mathbf{y}, f_{X,\mathbf{x}_*}, \theta)}{p(\mathbf{y})} = \frac{p(\mathbf{y} \,|\, f_X, \theta) p(f_{X,\mathbf{x}_*} | \theta) p(\theta)}{p(\mathbf{y})} \tag{11}$$

$$= \underbrace{\frac{p(\mathbf{y} \,|\, f_X, \theta) p(f_{X,\mathbf{x}_*} | \theta)}{p(\mathbf{y} \,|\, \theta)}}_{p(f_{X,\mathbf{x}_*} | \boldsymbol{\theta}, \mathbf{y})} \underbrace{\frac{p(\mathbf{y} \,|\, \theta) p(\theta)}{p(\mathbf{y})}}_{p(\theta \,|\, \mathbf{y})} \tag{12}$$

Posterior over f and θ consists of two parts

1. The original posterior over f,
2. A posterior over θ using the **marginal likelihood**:

$$p(\mathbf{y}|X, \theta) = \int p(\mathbf{y}|f(X), X, \theta) p(f(X)|\theta) \mathrm{d}f(X) \tag{13}$$

Marginal likelihood surface

1. To predict f, we need to take into account all uncertainty over both f and θ

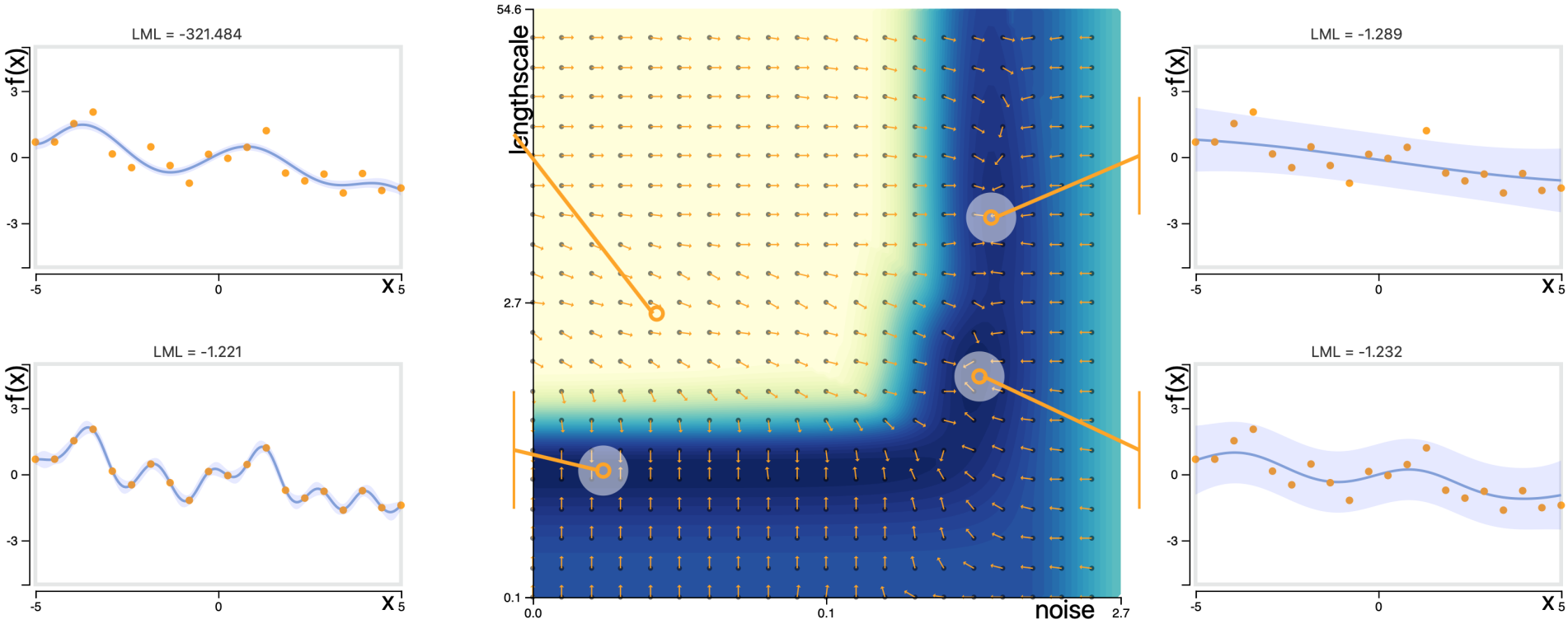
$$p(f(\mathbf{x}^*)|\mathbf{y}, X) = \int p(f(\mathbf{x}^*) \mid \mathbf{y}, X, \theta) p(\theta \mid \mathbf{y}, X) \mathrm{d}\theta \tag{14}$$

2. We take a $p(\theta)$ which is uniform over a large range of values

$$p(\theta|\mathbf{y}, X) \approx \frac{1}{Z} p(\mathbf{y} \mid X, \theta) \tag{15}$$

Marginal likelihood surface

Visualisation of hyperparameter posterior $p(\theta|\mathbf{y}, X) \approx p(\mathbf{y}|X, \theta)$:



- Several plausable hyperparameters
- Predictions should take posterior uncertainty into account!

Try for yourself: `https://drafts.distill.pub/gp/`

Intractable inference

To make a prediction, we need to compute

$$p(f(\mathbf{x}^*)|\mathbf{y}, X) = \int p(f(\mathbf{x}^*) \,|\, \mathbf{y}, X, \theta) p(\theta \,|\, \mathbf{y}, X) \mathrm{d}\theta \tag{16}$$

No closed-form solution for this integral. Inference is **intractable** :(

$$p(\theta \,|\, \mathbf{y}, X) = \frac{p(\mathbf{y} \,|\, X, \theta) p(\theta)}{p(\mathbf{y}|X)} = \frac{p(\mathbf{y} \,|\, X, \theta) p(\theta)}{\int p(\mathbf{y}|\theta, X) p(\theta) \mathrm{d}\theta} \tag{17}$$

- We can compute the **relative** plausibility of a finite number of hyperparameters,
- but the prediction needs to know the weight relative to the total volume of **all** hyperparameters.

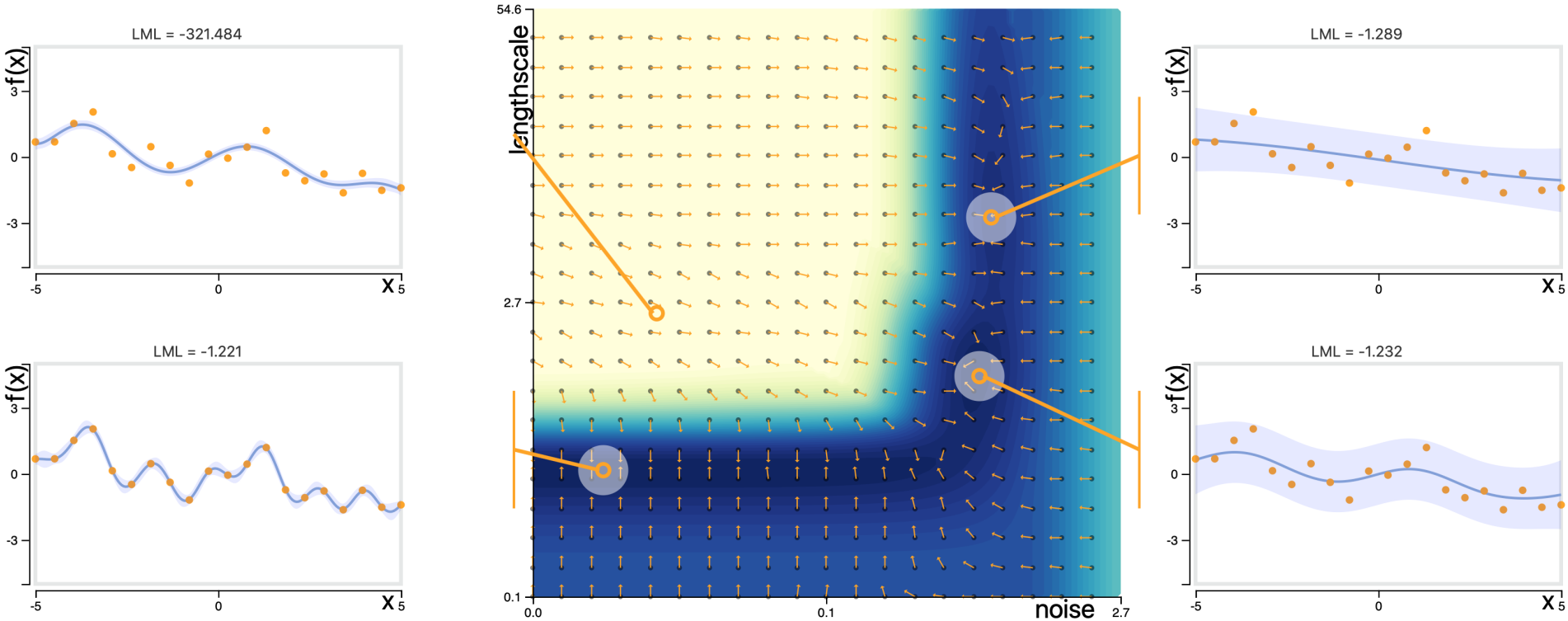
Practical solution

- Many approximations exist when closed-form solutions don't (variational, MCMC, ...)
- One pragmatic approximation is to **ignore uncertainty in θ**.

$$p(\theta|\mathbf{y}, X) \approx \delta(\theta - \hat{\theta}), \qquad \hat{\theta} = \underset{\theta}{\text{argmax}}\, p(\mathbf{y} \mid \theta, X)p(\theta) \tag{18}$$

- Maximum a-posteriori (MAP) approximation
- Found by numerically optimising $p(\mathbf{y}|\theta, X)p(\theta)$, using **gradients**

Numerical optimisation

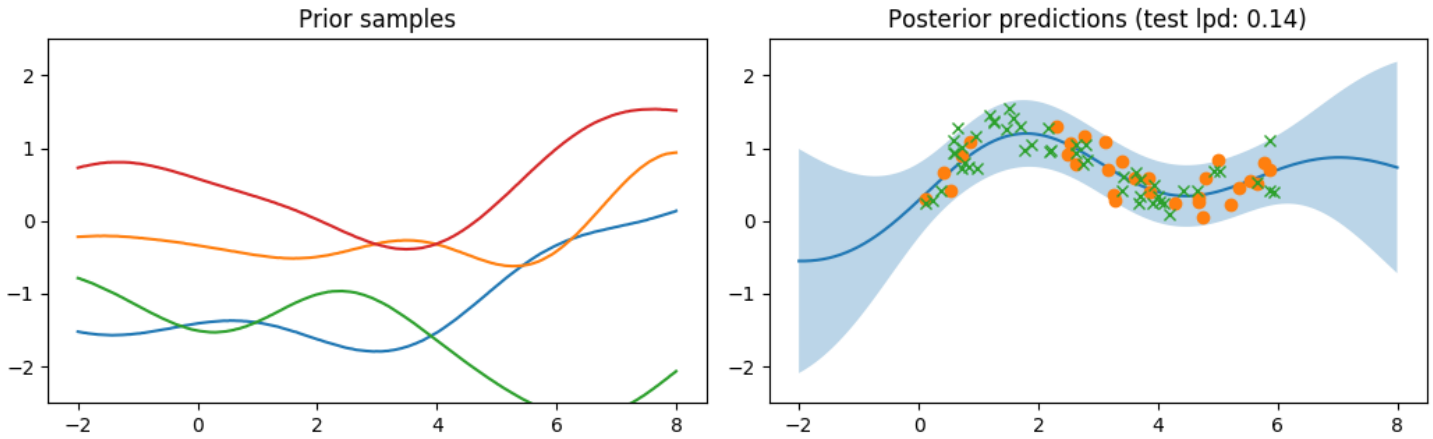


- Gradients indicated on image push you towards optima
- Surface is non-convex, so we can end up in multiple solutions
- Which one we end up in, depends on starting point

How to optimise

We are searching for $\text{argmax}_\theta\, p(\mathbf{y} \mid \theta, X)p(\theta)$, so

- Random re-starts at different locations
- Pick the θ with the highest value of $p(\mathbf{y} \mid \theta, X)p(\theta)$
- Pick a good initialisation based on your data

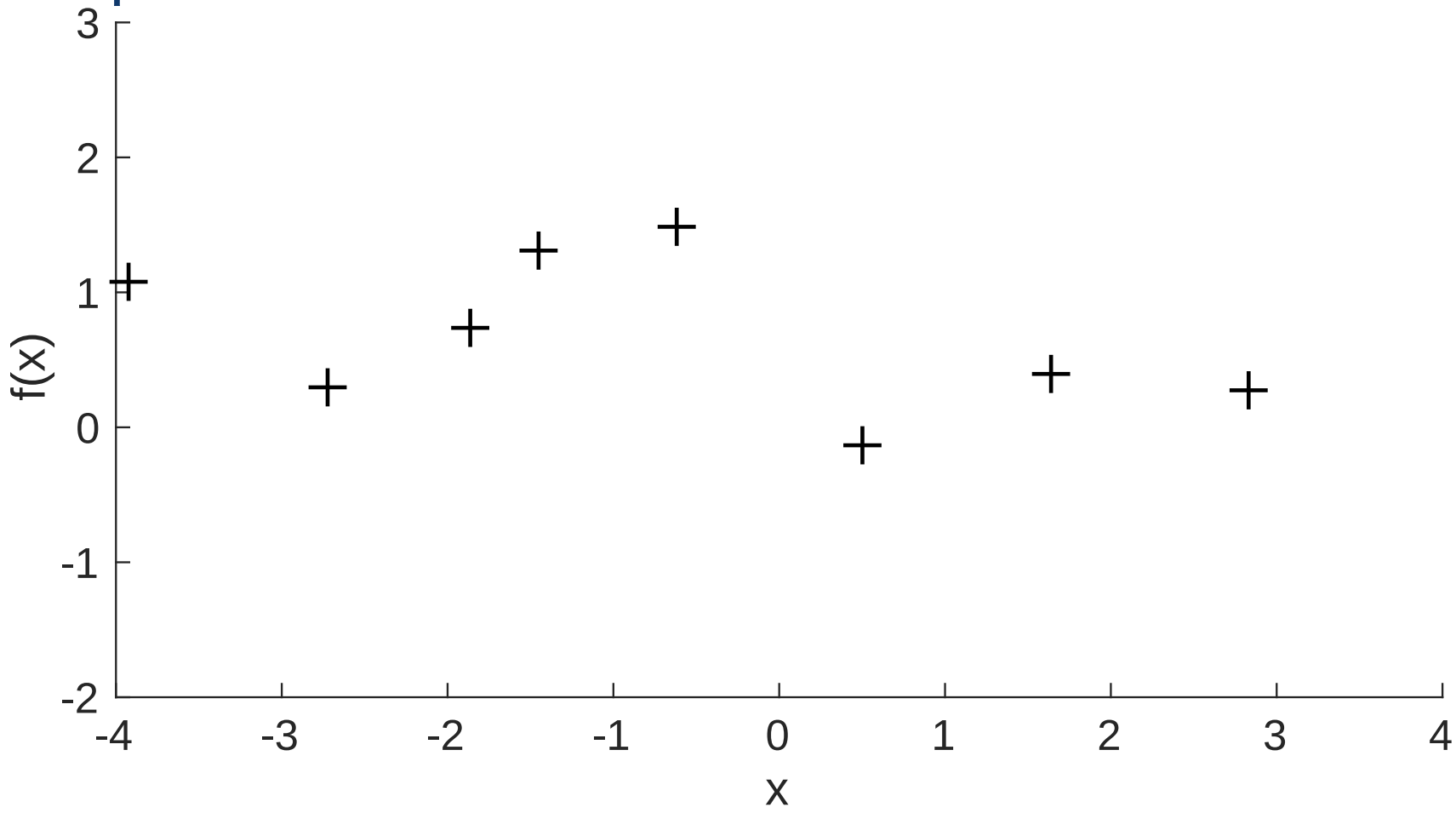


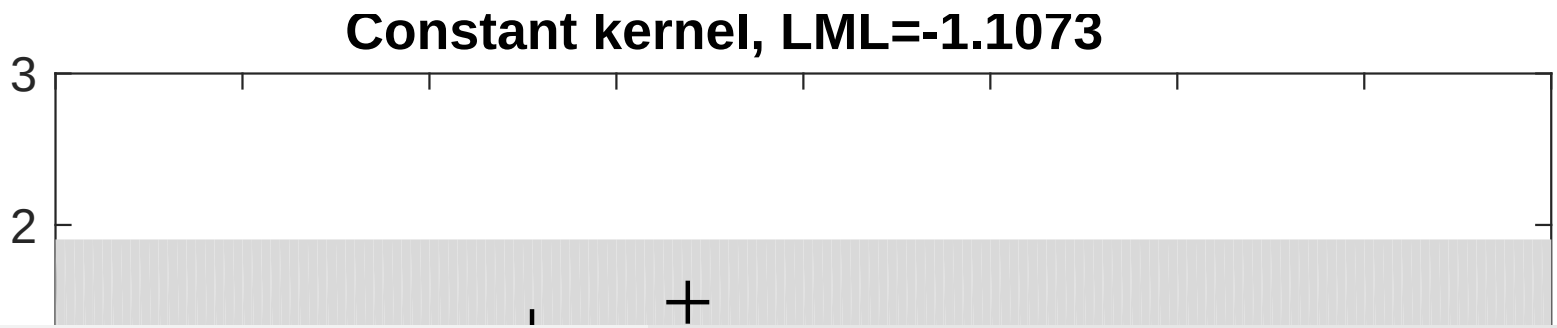
- Lengthscale appropriate to input range
- Variance appropriate to output range
- Noise scale based on how "predictable" you think the dataset is

When is MAP ok?

- More data → less uncertainty in θ
  → delta more appropriate
- More data → fewer local optima
  → optimisation more likely to work
- More parameters in θ, same data → uncertainty increases
  → delta less appropriate

Example





Fitting a real dataset

See Jupyter notebook `mauna.ipynb`.

Conclusion

- The assumptions in the prior distribution affect the posterior, and its generalisation characteristics
- We can apply Bayes rule to find the posterior over hyperparameters
- Bayesian integrals are hard, but maximising the posterior (MAP) can be reasonable

Further reading

- Rasmussen & Williams. *Gaussian Processes for Machine Learning*, chapter 5.

References I

[1] C. E. Rasmussen and C. K. Williams. *Gaussian processes for machine learning*. MIT press, Cambridge, MA, USA, 2006.