Bayesian Methods for Data Modelling (Part 1)

Mike Tipping

January 24, 2008

EPSRC Winter School:
Mathematics for Data Modelling

University of Sheffield

Outline (Part 1)

1. Prologue ...
   - "Ockham's Razor" and the "just right" model
2. Setting the Scene ...
   - A simple linear regression problem
   - Least-squares approximation
   - Complexity control & regularisation
3. Bayesian Inference
   - Likelihood, priors & inference
   - MAP estimation
   - Marginalisation
   - "Ockham's Razor" revisited

"Ockham's Razor"

- In the fourteenth century, William of Ockham proposed:

  "Pluralitas non est ponenda sine neccesitate"

  which literally translates as:

  "Entities should not be multiplied unnecessarily"

- In a data modelling context, of all potential solutions to a given problem, we would ideally choose the simplest

- Bayesian statistical inference automatically manages the trade-off between simplicity and solution accuracy

Bayesian Preference for Appropriate Simplicity

- Consider that we have a binary (■/❑) communication system
- We have a fixed dictionary of symbols (strings of bit-1):
  - ❑❑❑❑❑❑❑
  - ❑❑❑❑
  - ❑❑
  - ❑
- Messages are constructed by OR-ing an arbitrary number of bit-1 symbols in arbitrary positions within a field of bit-0's
- There may be transmission errors (independent inversion of bits)
- We receive a binary sequence: $\mathbf{t} =$ ■■❑❑❑❑■■❑❑
- What is the "best" decoding?

Decoding

Some possible decodings:

	Decoding	**Model $\mathcal{M}$**	**Error ϵ**
1		$6 \times$ ❑	-
2		$3 \times$ ❑❑	-
3		❑❑❑❑ + ❑❑	-
4		❑❑❑❑❑❑❑❑	2

- Models 1–3 predict the sequence perfectly
- Model 4 is simplest, but requires introduction of bit errors
- Without any further assumptions, we can show that Decoding 3 is most probable . . .

Decoding

- For each decoding, we calculate $p(\mathbf{t}|\mathcal{M})$, the probability assigned to the sequence by the model *with reference to all the other sequences that the model could potentially have decoded*

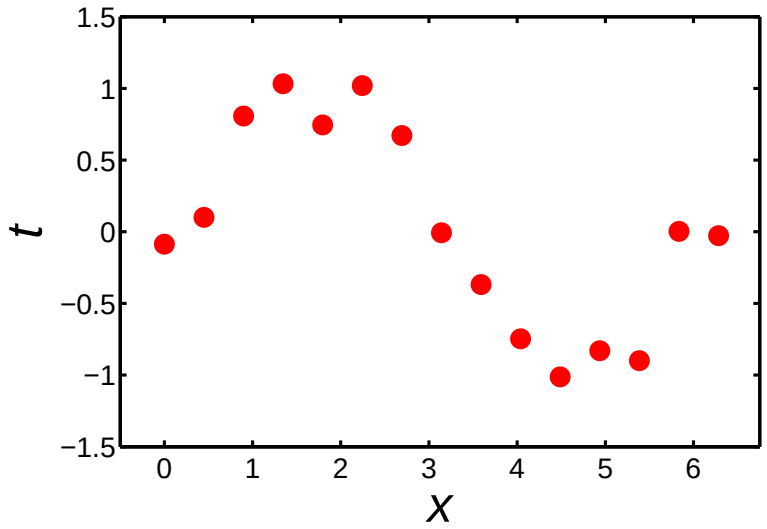
- Giving:

	Decoding	**Model $\mathcal{M}$**	ϵ	**# Sequences**	$p(\mathbf{t}|\mathcal{M})$
1	■■❑❑❑❑■■❑❑	$6 \times$ ❑	-	6! of 10^6	0.0007
2	■■❑❑❑❑■■❑❑	$3 \times$ ❑❑	-	3! of 9^3	0.0082
3	■■❑❑❑❑■■❑❑	❑❑❑❑ + ❑❑	-	1 of 7×9	0.0159
4	■■❑❑❑❑❑❑❑❑	❑❑❑❑❑❑❑❑	2	1 of $3 \times \binom{10}{2}$	0.0074

- This simple example is "Bayesian inference in disguise", and is exactly analogous to the way that Bayesian methods perform in more complex machine learning and data modelling tasks

An Example Modelling Problem

- We have a set of 'mystery' data:



- Truth: $N = 15$ samples synthesised from the function $y = \sin(x)$ with added Gaussian noise of standard deviation 0.2
- The 'input' variables are denoted x_n, $n = 1 \ldots N$
- For each x_n, there is an associated real-valued observation t_n

Linear (in-the-parameter) Models

- Model choice: parametric function $y(x; \mathbf{w})$
- A linearly-weighted sum of M fixed basis functions $\phi_m(x)$:

$$y(x; \mathbf{w}) = \sum_{m=1}^{M} w_m \phi_m(x)$$

- Example: Gaussian data-centred basis functions:

$$\phi_m(x) = \exp\left\{-(x - x_m)^2 / r^2\right\}$$

  - a "radial basis function" (RBF) model
  - $M = N = 15$ in this example

"Least-squares" Approximation

- Goal: find $\mathbf{w}$ such that $y(x;\mathbf{w})$ is a 'good' model
- Start with a classic approach: *least-squares*, minimising:

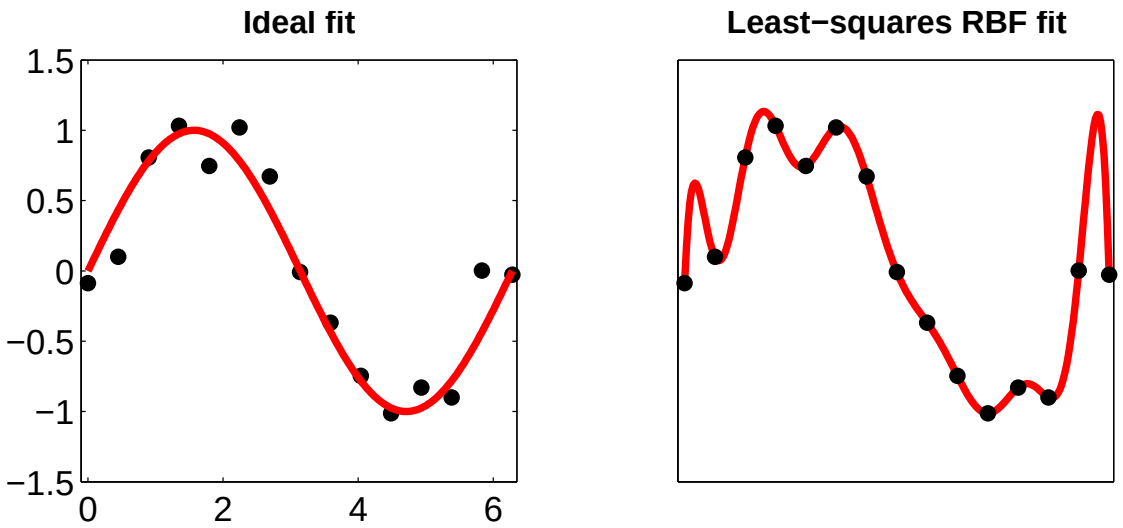
$$E_{LS}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left[t_n - \sum_{m=1}^{M} w_m \phi_m(x_n)\right]^2$$

- If $\mathbf{\Phi}$ is the 'design matrix' such that $\mathbf{\Phi}_{nm} = \phi_m(x_n)$, and $\mathbf{t} = (t_1, \ldots, t_N)^{\mathrm{T}}$ then:

$$\mathbf{w}_{LS} = (\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi})^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$$

Model Complexity?

- With $M = 15$ basis functions and only $N = 15$ examples, minimisation of squared-error leads to "over-fitting":



- How do we judge which model is "better"?

- To estimate complexity, we *must* introduce some prior knowledge, preference, expectation, prejudice ...

Complexity Control: Regularisation

- We typically prefer smoother functions, which typically have smaller weights $\mathbf{w}$
- Augment the error function with a weight penalty term:

$$E_{PLS}(\mathbf{w}) = E_{LS}(\mathbf{w}) + \lambda E_W(\mathbf{w})$$

- A conventional choice is the squared-weight penalty:

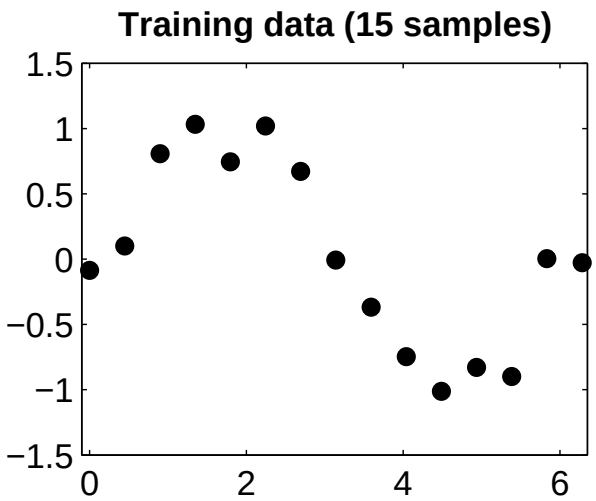
$$E_W(\mathbf{w}) = \frac{1}{2}\sum_{m=1}^{M} w_m^2$$

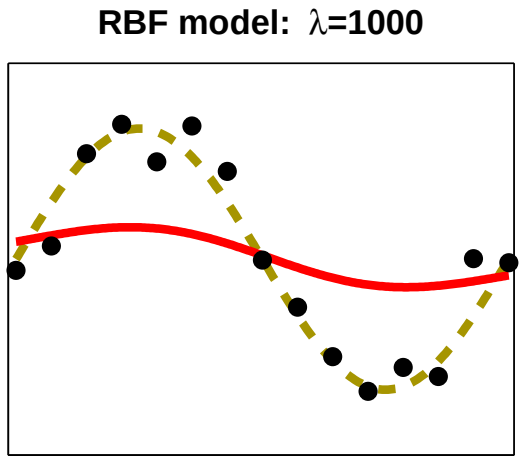
- This conveniently gives the "penalised least-squares" estimate:

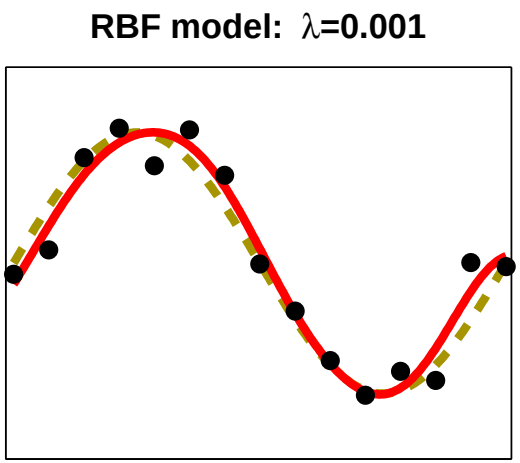
$$\mathbf{w}_{PLS} = (\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi} + \lambda\mathbf{I})^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$$

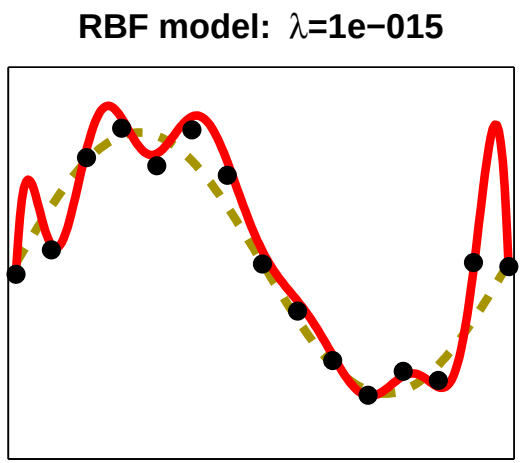
The Regularisation Hyperparameter

- The *hyperparameter* λ controls the trade-off between quality of fit, $E_{LS}(\mathbf{w})$, and smoothness, $E_W(\mathbf{w})$



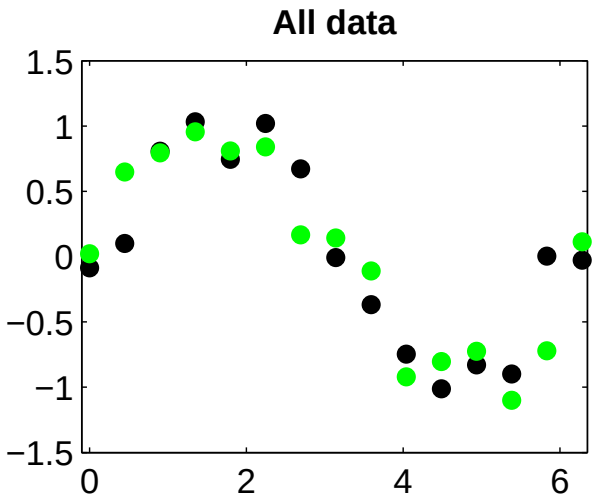


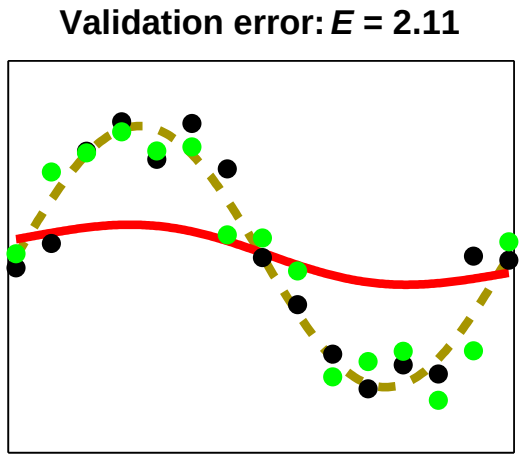


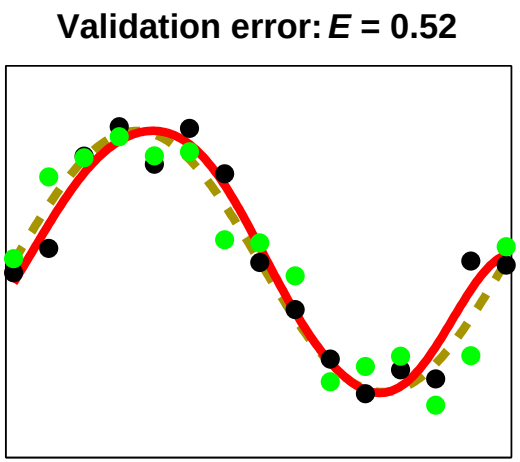


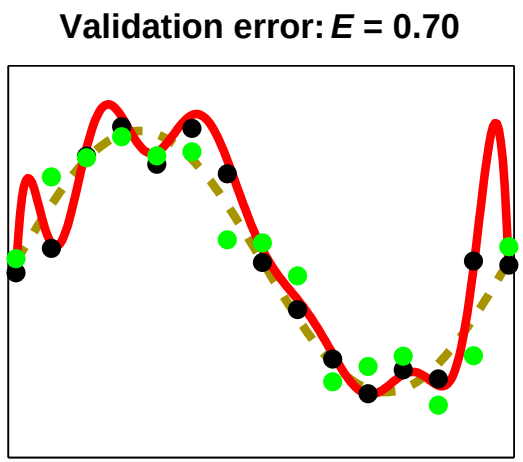
Estimating λ via Validation (1)

- Assess candidate values of λ according to validation set data error

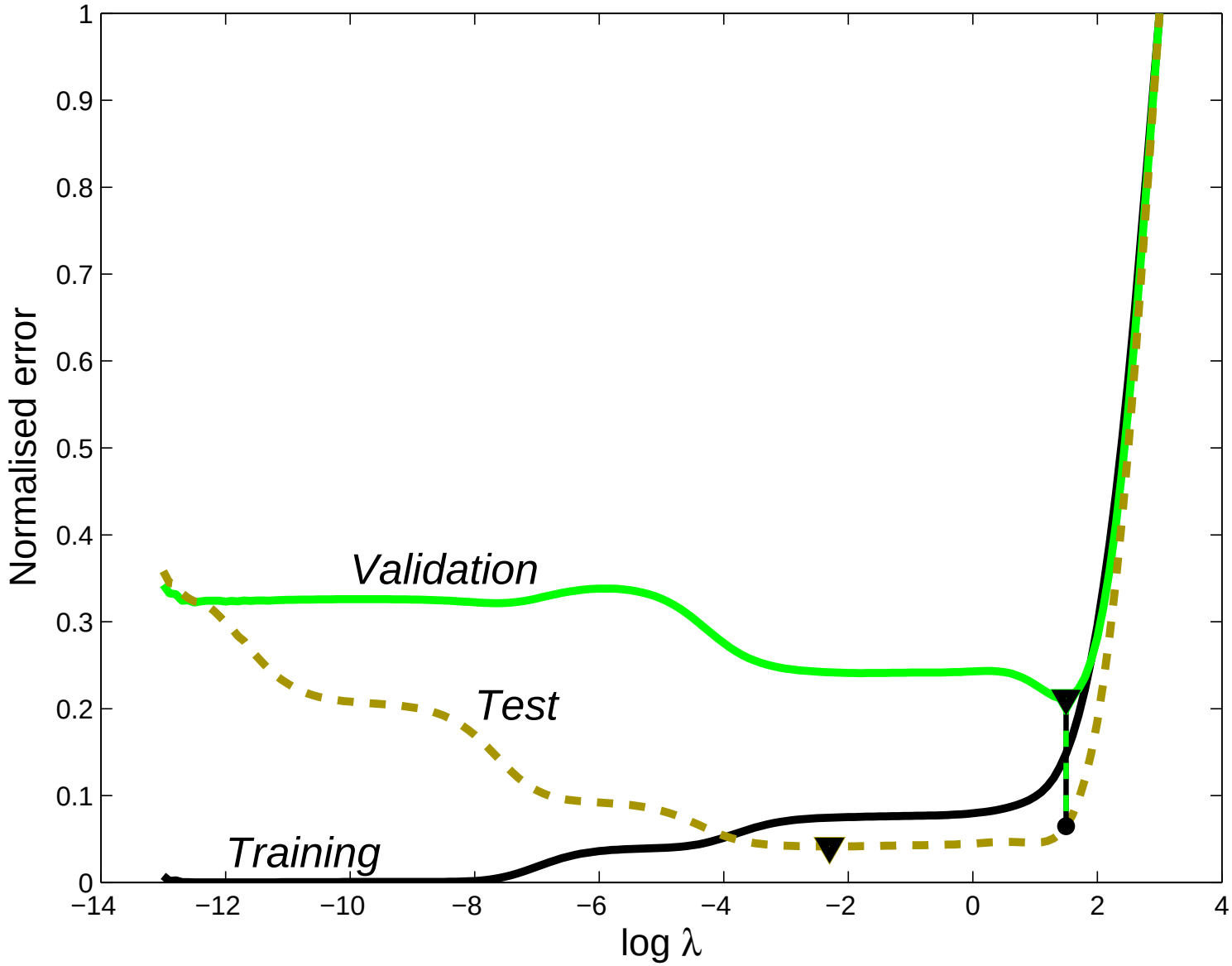








Estimating λ via Validation (2)



Bayesian Inference: Basic Principles

- Define *prior* probability distributions over *all* model variables:
  - Inherently stochastic quantities (*e.g.* the observations $\mathbf{t}$)
  - All *parameters* (*e.g.* $\mathbf{w}$, σ, λ)
  - The model $\mathcal{M}$ itself (*e.g.* its type, structure, basis choice *etc*)
- Update these distributions in light of the data (Bayes' rule!)
- *Integrate out* variables which are not directly of interest
  - Most required integrations are analytically intractable ✘
- Key features of the Bayesian approach:
  - A consistent way to deal with all sources of uncertainty ✔
  - An explicit framework for encoding prior knowledge ✔
  - Automatic implementation of "Ockham's Razor" ✔

Bayesian Inference: Likelihood Model

- Data is a noisy sample from the underlying function:

$$t_n = y(x_n; \mathbf{w}) + \epsilon_n$$

- Gaussian zero-mean noise model with variance σ^2:

$$p(\epsilon_n|\sigma^2) = N(0, \sigma^2)$$

- Assuming independence, the *likelihood* $p(\mathbf{t}|\mathbf{w}, \sigma^2)$ of the data is:

$$\prod_{n=1}^{N} p(t_n|\mathbf{w}, \sigma^2) = \prod_{n=1}^{N} (2\pi\sigma^2)^{-1/2} \exp\left[-\frac{\{t_n - y(x_n; \mathbf{w})\}^2}{2\sigma^2}\right]$$

- So "maximum-likelihood" $\equiv$ "least-squares" here

Bayesian Inference: Prior Distributions

- Model complexity is controlled by specifying a *prior* distribution which expresses a "degree of belief" regarding appropriate values for $\mathbf{w}$ *before observing the data*
- A conventional choice is a zero-mean Gaussian:

$$p(\mathbf{w}|\alpha) = \left(\frac{\alpha}{2\pi}\right)^{M/2} \prod_{m=1}^{M} \exp\left\{-\frac{\alpha}{2} w_m^2\right\}$$

- This expresses a preference for smoother models by declaring smaller weights to be *a priori* more probable
- The strength of this preference is controlled by the shared inverse-variance hyperparameter α

Bayesian Inference: Bayes' Rule!

- Given the likelihood and the prior, we compute the *posterior distribution* over $\mathbf{w}$ via Bayes' rule:

$$p(\mathbf{w}|\mathbf{t},\alpha,\sigma^2) = \frac{p(\mathbf{t}|\mathbf{w},\sigma^2)\,p(\mathbf{w}|\alpha)}{p(\mathbf{t}|\alpha,\sigma^2)} = \frac{\text{likelihood} \times \text{prior}}{\text{normalising factor}}$$

- Here, the posterior is Gaussian: $p(\mathbf{w}|\mathbf{t},\alpha,\sigma^2) = N(\boldsymbol{\mu},\boldsymbol{\Sigma})$ with

$$\boldsymbol{\mu} = (\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} + \sigma^2\alpha\mathbf{I})^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}$$
$$\boldsymbol{\Sigma} = \sigma^2(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} + \sigma^2\alpha\mathbf{I})^{-1}$$

Rules of probability

MAP Estimation: a 'Bayesian' Short-cut

- The "maximum *a posteriori*" (MAP) estimate for $\mathbf{w}$ is the single most probable value under the posterior distribution $p(\mathbf{w}|\mathbf{t}, \alpha, \sigma^2)$
- For a Gaussian posterior, the maximum is equal to the mean:

$$\mathbf{w}_{MAP} = \boldsymbol{\mu} = (\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} + \sigma^2\alpha\mathbf{I})^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}$$

- Recall: $\mathbf{w}_{PLS} = (\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} + \lambda\mathbf{I})^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}$
- The MAP estimate is therefore identical to the penalised least-squares estimate re-parameterised with $\lambda = \sigma^2\alpha$

Demo: Bayesian Posterior Inference

- Over to Matlab …

Marginalisation

- The MAP/PLS equivalence does *not* mean that the Bayesian framework is simply a re-interpretation of classical methods!
- The key element of Bayesian inference is *marginalisation*, where we seek to integrate out all 'nuisance' variables, including $\mathbf{w}$
- This integration procedure automatically implements "Ockham's Razor": the intrinsic assignment of higher probability to "appropriately complex" models
- We'll exploit this to:
  - robustly estimate the hyperparameter α/λ (next)
  - selection of the model itself (later)

The True Bayesian Path

- We should define priors over *all* variables, not just the weights $\mathbf{w}$
- Having defined priors $p(\alpha)$ and $p(\sigma^2)$, we apply Bayes' rule:

$$p(\mathbf{w}, \alpha, \sigma^2|\mathbf{t}) = \frac{p(\mathbf{t}|\mathbf{w}, \sigma^2)\, p(\mathbf{w}|\alpha)\, p(\alpha)\, p(\sigma^2)}{p(\mathbf{t})}$$

- Not computable in closed form since the integral:

$$p(\mathbf{t}) = \int p(\mathbf{t}|\mathbf{w}, \sigma^2)\, p(\mathbf{w}|\alpha)\, p(\alpha)\, p(\sigma^2)\, d\mathbf{w}\, d\alpha\, d\sigma^2$$

  is not analytically tractable

A Pragmatic Deviation from the Path

- We can't compute $p(\mathbf{w}, \alpha, \sigma^2|\mathbf{t})$ analytically, so we desire a workable approximation
- We decompose the joint posterior as:

$$p(\mathbf{w}, \alpha, \sigma^2|\mathbf{t}) \equiv p(\mathbf{w}|\mathbf{t}, \alpha, \sigma^2)\, p(\alpha, \sigma^2|\mathbf{t})$$

- The 'weight posterior' distribution $p(\mathbf{w}|\mathbf{t}, \alpha, \sigma^2)$ is tractable
- The 'hyperparameter posterior' $p(\alpha, \sigma^2|\mathbf{t})$ must be approximated

Type-II Maximum Likelihood

- Find the single 'most probable' values α_{MP} and σ^2_{MP} under the posterior distribution:

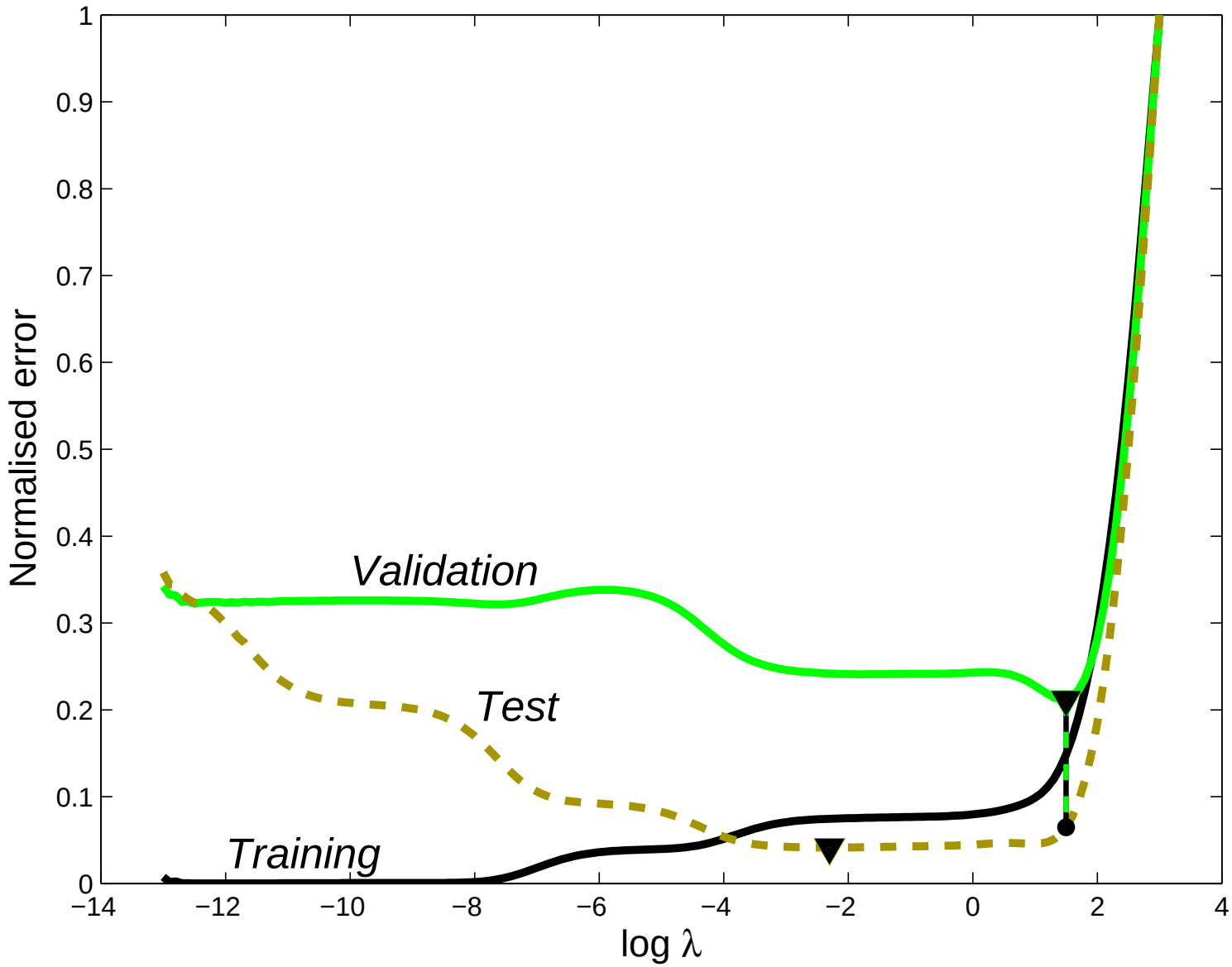
$$p(\alpha, \sigma^2|\mathbf{t}) = \frac{p(\mathbf{t}|\alpha, \sigma^2)\, p(\alpha)\, p(\sigma^2)}{p(\mathbf{t})}$$

- Assume log-uniform *hyperpriors* over $p(\alpha)$ and $p(\sigma^2)$
- Maximise $p(\mathbf{t}|\alpha, \sigma^2)$, the *marginal likelihood* of the training data:

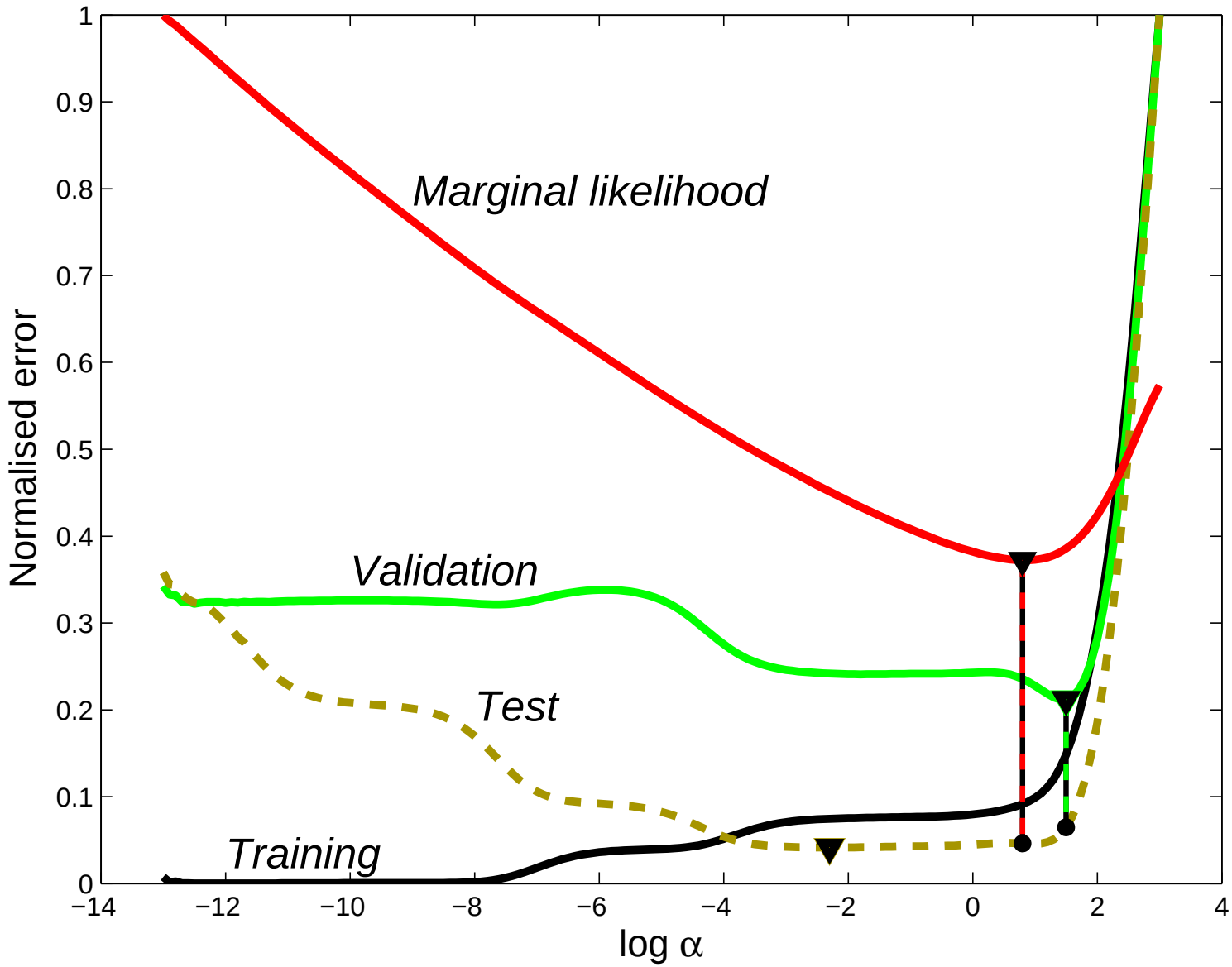
$$p(\mathbf{t}|\alpha, \sigma^2) = \int p(\mathbf{t}|\mathbf{w}, \sigma^2)\, p(\mathbf{w}|\alpha)\, d\mathbf{w}$$

$$= (2\pi)^{-N/2} |\sigma^2\mathbf{I} + \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}|^{-1/2} \exp\left\{-\frac{1}{2}\mathbf{t}^{\mathrm{T}}(\sigma^2\mathbf{I} + \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}})^{-1}\mathbf{t}\right\}$$

Estimating α via Validation

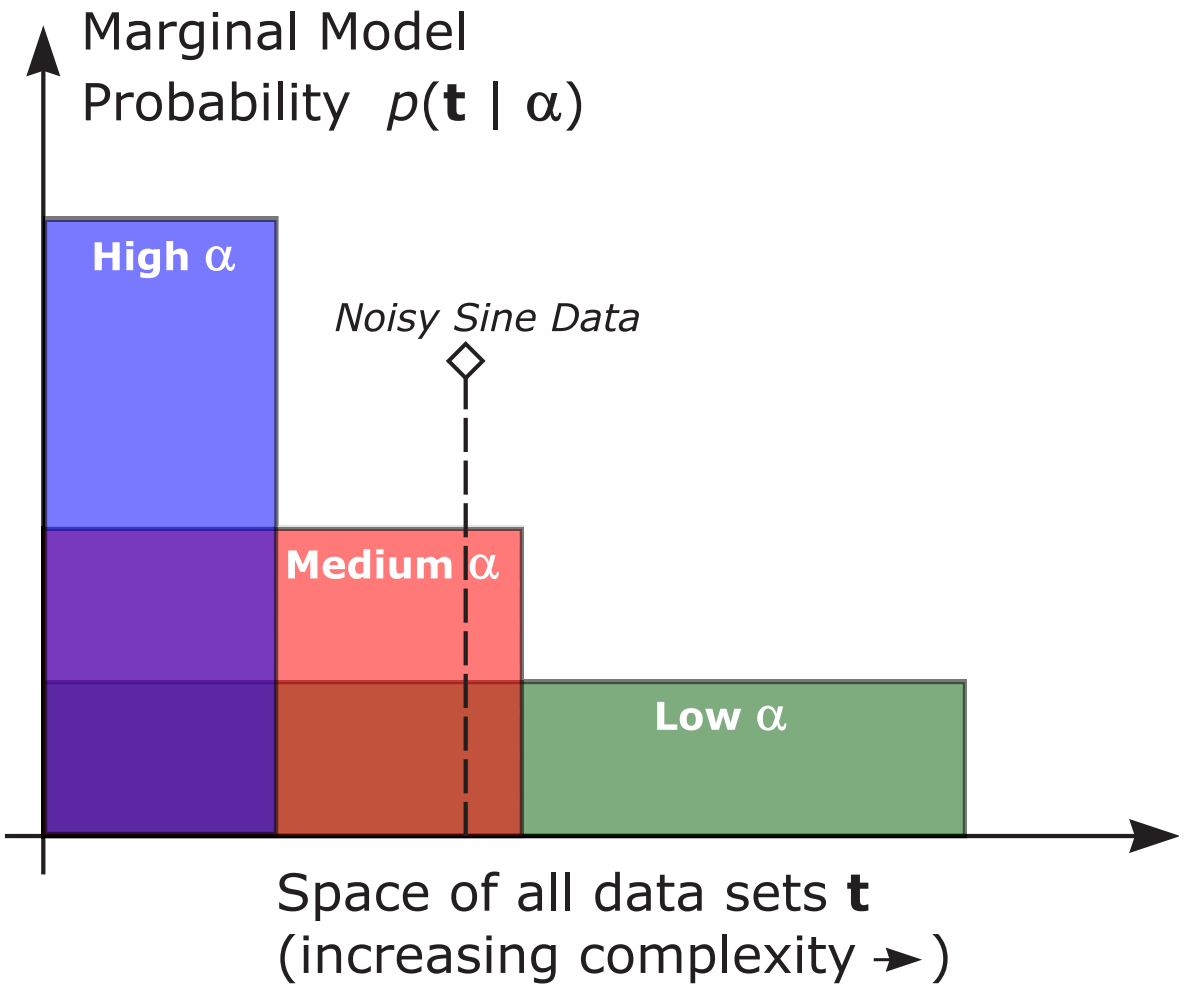


Estimating α via Marginal Likelihood



Ockham's Razor revisited

- Marginalisation over $\mathbf{w}$ implements "Ockham's Razor" by rejecting models that are both too simple and too complex



End of Part One . . .

Don't go away, we'll be right back

Rules of Probability

- Product rule:

$$p(a,b) = p(a|b)\,p(b) = p(b|a)\,p(a)$$

- More generally:

$$p(a,b|c) = p(a|b,c)\,p(b|c) = p(b|a,c)\,p(a|c)$$

- Rearranging gives Bayes' rule:

$$p(a|b,c) = \frac{p(b|a,c)\,p(a|c)}{p(b|c)}$$

- Sum (integral) rule:

$$p(b|c) = \int_{-\infty}^{\infty} p(a,b|c)\,da$$

Return