Chapter 3 Linear Models for Regression

Regression (supervised learning)

$\mathbf{x}$: D-dimensional input vector

t : continuous target variable

Linear regression model : linear combination of non linear basis functions of the input variables with adjustable parameters. E.g. of basis function

$$x \xrightarrow{\Phi} \begin{pmatrix} 1 \\ x \\ x^2 \\ x^3 \end{pmatrix}$$

$$y(x, w) = w_0 + w_1 x + w_2 x^2 + w_3 x^3$$

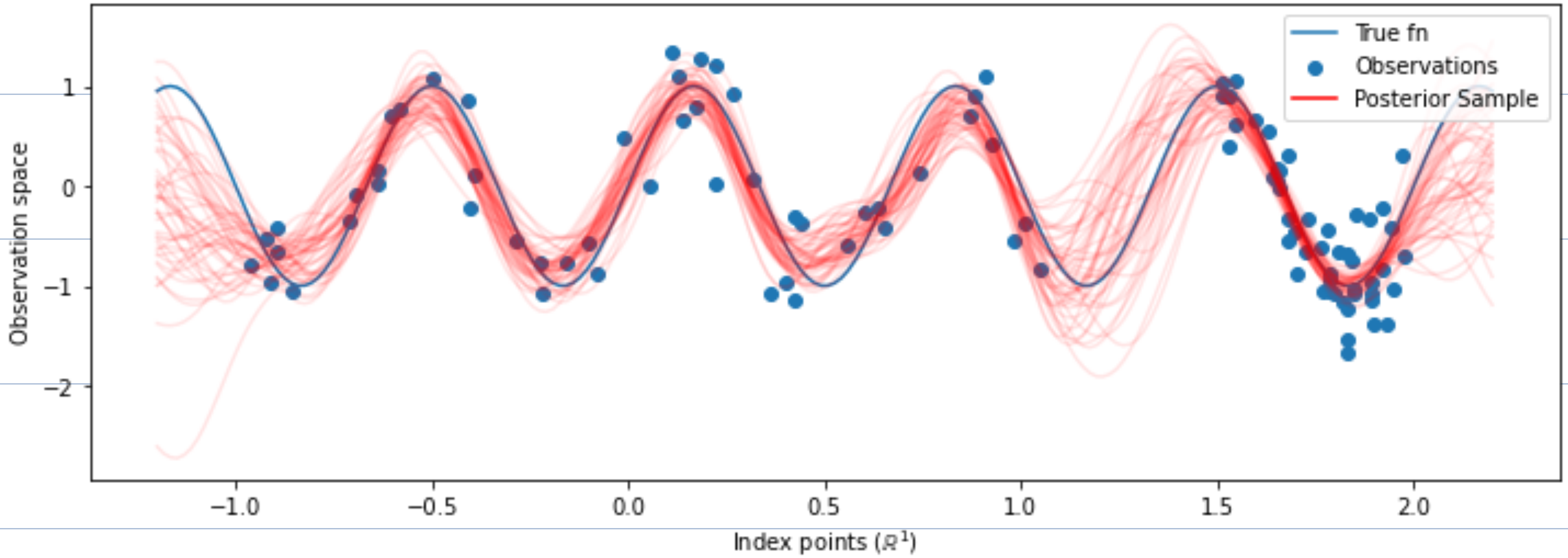
whose basis is $\{1, x, x^2, x^3\}$.

N observations $\{x_n\}$ $n = 1, 2, \dots N$, corresponding target t_n.

Goal: predict the value of t for a new x.

Simplest approach: constructing appropriate function $y(x)$

General or probabilistic perspective: modeling the predictive distribution $p(t \mid x)$ (uncertainty about t for each x)



3.1 Linear Basis Function Models

Linear combinations of fixed nonlinear functions of input $\mathbf{x}$

$$y(\mathbf{x}, \mathbf{w}) := w_0 + \sum_{j=1}^{M-1} w_j \phi_j(\mathbf{x})$$

where $\phi_j(\mathbf{x})$ are known as basis functions. w_0 allows for any fixed offset, called bias.

So it is convenient to define an additional dummy 'basis function' $\phi_0(\mathbf{x}) = 1$ so that

$$\overset{D\text{-dim}}{\mathbf{x}} \longrightarrow \overset{M\text{-dim}}{\Phi(\mathbf{x})}$$

$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\mathbf{x}) = \mathbf{w}^T \Phi(\mathbf{x})$$

where $w = (w_0, .. w_{M-1})^T$ and $\Phi(x) = (\underbrace{\phi_0}_{=1}, \phi_1(x), \dots \phi_{M-1}(x))^T$

In view of pre-processing or feature extraction, the feature can be expressed as $\{\phi_j(x)\}$

Basis functions

In Chapter 1. there is a single input x and the basis functions take the form of powers of x ($\phi_j(x) = x^j$)

One limitation of polynomial basis : global functions

Gaussian basis functions

$$\phi_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\}$$

where μ_j govern the locations of the basis functions in input space and s governs their spatial scale.

Sigmoidal basis functions

$$\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right)$$

where σ is the logistic sigmoid $\sigma(a) = \frac{1}{1+\exp(-a)}$

Equivalently, we can use the 'tanh' function. Since $\tanh(a) = 2\sigma(2a) - 1$, general linear combination of sigmoid is equivalent to a general linear combination of tanh

Most of the discussion in this chapter is independent of the particular choice of basis.

3.1.1 Maximum likelihood and least squares

We have showed SSE could be motivated as the maximum likelihood solution under an assumed Gaussian noise model. As before, we assume that target t is given by a deterministic function $y(\mathbf{x}, \mathbf{w})$ with additive Gaussian noise

$$t = y(\mathbf{x}, \mathbf{w}) + \varepsilon$$

where ε is a zero mean Gaussian random variable with precision β (inverse of variance)

I.e.

$$p(t \mid \mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}(t \mid y(\mathbf{x}, \mathbf{w}), \beta^{-1}) \tag{3.8}$$

In section 1.5.5, we showed that

$$\mathbb{E}_t[t \mid \mathbf{x}] = \int t \, p(t \mid \mathbf{x}) \, dt \tag{3.9}$$

is the optimal prediction

Consider inputs $\mathbf{X} = \{\mathbf{x}_1, .. \mathbf{x}_N\}$ with corresponding target $t_1 ... t_N$. Let $\mathbf{t} := (t_1 ..., t_N)^T$. Assume $\mathbf{X}$ and $\mathbf{t}$ are drawn independently from (3.8). Then the likelihood function of $\mathbf{w}$ and β is in the form

$$p(\mathbf{t} \mid \mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n \mid \overset{y(\mathbf{x}, \mathbf{w})}{\mathbf{w}^T \boldsymbol{\Phi}(\mathbf{x}_n)}, \beta^{-1}) \qquad (3.10)$$

We will drop the explicit $\mathbf{X}$ from expressions.

$$\ln p(\mathbf{t} \mid \mathbf{w}, \beta) = \sum_{n=1}^{N} \ln \mathcal{N}(t_n \mid \mathbf{w}^T \boldsymbol{\Phi}(\mathbf{x}_n), \beta^{-1}) \qquad (3.11)$$

$$= \frac{N}{2} \ln \beta - \frac{N}{2} \ln(2\pi) - \beta E_D(\mathbf{w})$$

where SSE is defined by

$$E_D(\mathbf{w}) := \frac{1}{2}\sum_{n=1}^{N}\{\underset{\text{실제}}{t_n} - \underset{\text{예측}}{\mathbf{w}^T\Phi(\mathbf{x}_n)}\}^2 \qquad (3.12)$$

Consider first the maximization of (3.11) w.r.t $\mathbf{w}$.

Maximization of likelihood function under a conditional Gaussian (3.10) for a linear model $\Longleftrightarrow$ Minimizing $E_D(\mathbf{w})$

$$\nabla_{\mathbf{w}} \ln p(\mathbf{t} \mid \mathbf{w}, \beta) = \beta \sum_{n=1}^{N}\{t_n - \mathbf{w}^T\Phi(\mathbf{x}_n)\}\Phi(\mathbf{x}_n)$$

Setting this gradient to zero gives

$$\mathbf{0} = \sum_{n=1}^{N} t_n \Phi(\mathbf{x}_n) - \left(\sum_{n=1}^{N} \Phi(\mathbf{x}_n) \Phi(\mathbf{x}_n)^T \right) \mathbf{w}$$

Solving for $\mathbf{w}$ we obtain

$$\Phi(\mathbf{x}_n) := \begin{pmatrix} 1 \\ \phi_1(\mathbf{x}_n) \\ \vdots \\ \phi_{M-1}(\mathbf{x}_n) \end{pmatrix}$$

$$\mathbf{w}_{ML} = (\underset{M\times N}{\Phi^T} \underset{N\times M}{\Phi})^{-1} \overset{M\times N}{\Phi^T} \overset{N\times 1}{\underline{\mathbf{t}}} \qquad O(M^3)$$

which are known as normal equations for least square problem.

Here Φ is an $N \times M$ matrix called design matrix whose elements are given by $\Phi_{nj} = \underline{\phi_j(\mathbf{x}_n)}$

I.e.

$$\Phi = \begin{array}{cc} & M \\ N & \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & & \\ \phi_0(\mathbf{x}_N) & & & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix} \end{array} = \begin{pmatrix} \Phi(\mathbf{x}_1)^T \\ \Phi(\mathbf{x}_2)^T \\ \vdots \\ \Phi(\mathbf{x}_N)^T \end{pmatrix}$$

where $\Phi(\mathbf{x}) = (\phi_0(\mathbf{x}), \dots \phi_{M-1}(\mathbf{x}))^T$.

The quantity

$$\Phi^{\dagger} := (\Phi^T \Phi)^{-1} \Phi^T$$

is known as the Moor - Penrose pseudo - inverse of Φ.

If Φ is square and invertible, then $\Phi^{+} = \Phi^{-1}$.

Let us see the role of bias parameter w_0 w_0

$$E_D(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left\{t_n - w_0 - \sum_{j=1}^{M-1} w_j \phi_j(\mathbf{x}_n)\right\}^2$$ SSE

Set $\frac{\partial E_D}{\partial w_0} = 0$ and solving for w_0. Then we obtain

$$w_0 = \bar{t} - \sum_{j=1}^{M-1} w_j \bar{\phi}_j$$

where we have defined $\bar{t} := \frac{1}{N}\sum t_n$, $\bar{\phi}_j := \frac{1}{N}\sum \phi_j(\mathbf{x}_n)$

Thus, w_0 is the difference between the averages of t_n and the weighted sum of the averages of basis function values

After finding w_{ML}, we can maximize log likelihood (3.11) w.r.t noise precision parameter β,

$$\frac{1}{\beta_{ML}} = \underbrace{\frac{1}{N}\sum_{n=1}^{N}\left\{t_n - w_{ML}^T\phi(x_n)\right\}^2}_{\text{residual variance of target}}$$

3.1.2 Geometry of Least squares

N-dim

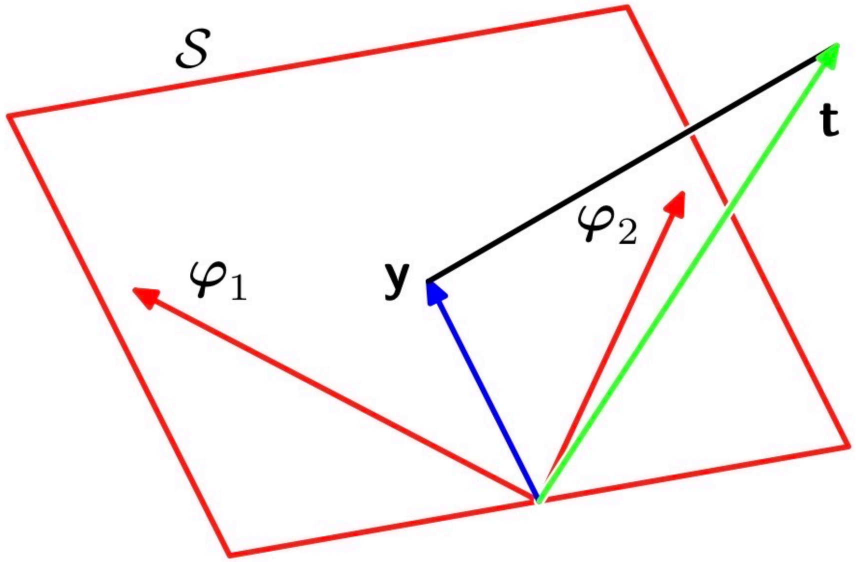


Figure 3.2 Geometrical interpretation of the least-squares solution, in an N-dimensional space whose axes are the values of $t_1, \ldots, t_N$. The least-squares regression function is obtained by finding the orthogonal projection of the data vector **t** onto the subspace spanned by the basis functions $\phi_j(\mathbf{x})$ in which each basis function is viewed as a vector φ_j of length N with elements $\phi_j(\mathbf{x}_n)$.

M 개

$$\mathbf{t} = \begin{pmatrix} t_1 \\ \vdots \\ t_N \end{pmatrix}$$

N - dimensional space whose axes are given by t_n.

For fixed j, basis function values $\phi_j(x_n)$ (N data points) can be represented as a vector in the same space

N - dim

Denote φ_j by this vector, given by

$$\varphi_j := (\phi_j(x_1), \phi_j(x_2), \dots, \phi_j(x_N))^T \quad (\ j^{th} \text{ colum of } \Phi\ )$$

where $j = 0, 1, \dots, M-1$. N 차원 M 개의 vector

Let $M < N$ and S be the M-dim subspace spanned by φ_j

Define $\mathbf{y} := (\underbrace{y(x_1, w)}_{= \sum_{j=0}^{M-1} w_j \phi_j(x_1)}, y(x_2, w), \dots y(x_N, w))^T$. Because $\mathbf{y}$

☆ $\mathbf{y} = w_0 \varphi_0 + w_1 \varphi_1 + \dots w_{M-1} \varphi_{M-1}$ ☆

is an arbitrary linear combination of φ_j, $\mathbf{y}$ live anywhere in the M-dimensional subspace S.

$\mathbf{y}$ 의 모든 component는 동일한 $w_0, \dots w_{M-1}$ 을 공유

SSE (3.12) is equal (up to a factor $\frac{1}{2}$) to $\|\mathbf{y} - \mathbf{t}\|^2$ (squared Euclidean distance)

Thus the least square solution $\mathbf{w}$ corresponds to that choice of $\mathbf{y}$ lying in subspace S and that is closest to $\mathbf{t}$

3.1.3 Sequential learning

A.k.a on-line algorithm

Applying the technique of stochastic gradient descent, also Known as sequential gradient descent

If the error function $E = \Sigma_n E_n$, then after presentation of pattern n, SGD updates w using

$$w^{(\tau+1)} := w^{(\tau)} - \eta \nabla \boxed{E_n}$$

where τ : iteration number, η : learning rate parameter.

For the case of SSE (3.12), $E_n = \frac{1}{2}\left(t_n - w^T \Phi(x_n)\right)^2$

$$w^{(\tau+1)} = w^{(\tau)} + \eta\,(\underbrace{\overset{\text{실제}}{t_n} - w^{(\tau)T}\Phi_n}_{\text{예측}})\,\Phi_n$$

where $\Phi_n := \Phi(x_n) = \left(\phi_0(x_n), \phi_1(x_n), \dots \phi_{M-1}(x_n)\right)^T$

3.1.4 Regularized least squares

To prevent over-fitting, we added regularization term so that

$$\overset{SSE}{E_D(w)} + \lambda E_W(w)$$

where λ is the regularization coefficient.

Simple for of regularizer is as follow

$$E_W(w) := \frac{1}{2} w^T w$$

weight decay

parameter shrinkage

If we also consider SSE, then the total error function becomes

$$\frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^T \Phi(\mathbf{x}_n)\}^2 + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w} \qquad (3.27)$$

Set the gradient of (3.27) w.r.t $\mathbf{w}$ to zero and solve for $\mathbf{w}$. Then we obtain the solution $\mathbf{w}$

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

More general regularizer is used as follows

$$\frac{1}{2}\sum_{n=1}^{N}\{t_n - w^T\Phi(x_n)\}^2 + \frac{\lambda}{2}\sum_{j=1}^{M}|w_j|^q \quad (3.29)$$

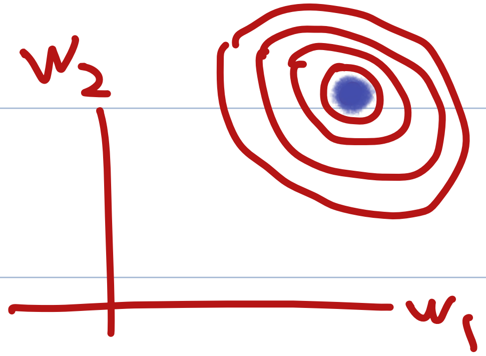
where $q=2$ corresponds to the quadratic regularizer (3.27)

The case of $q=1$ is known as lasso.

Exercise 3.5 and Appendix E

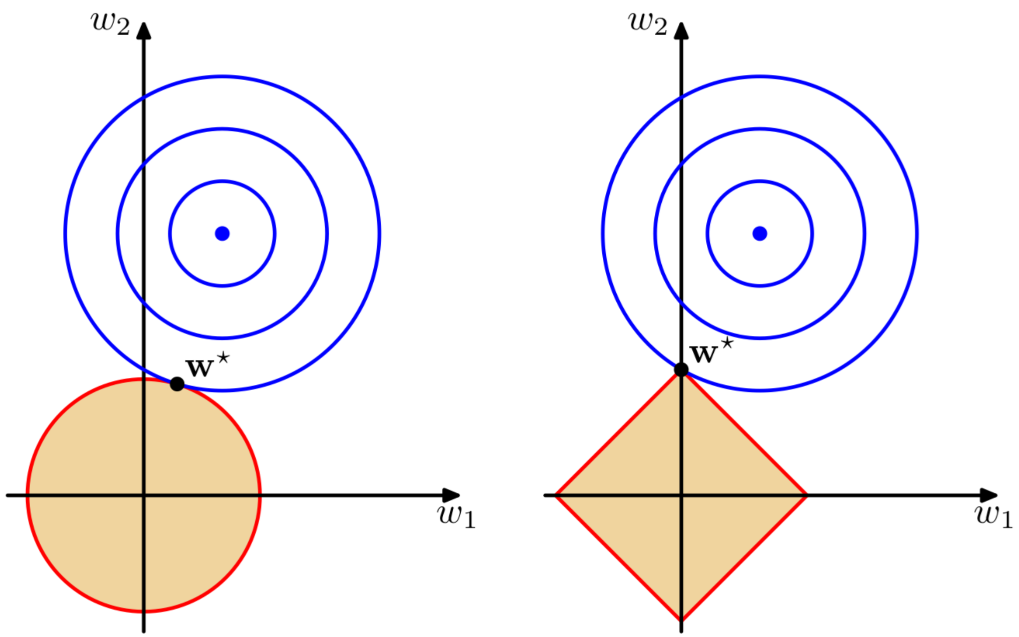
Minimize (3.29) $\iff$ Minimize $E_D(w)$ subject to the constraints

$$\sum_{j=1}^{M} |w_j|^q \le \eta$$



for some appropriate value of $\eta(\lambda)$

Figure 3.4 Plot of the contours of the unregularized error function (blue) along with the constraint region (3.30) for the quadratic regularizer $q = 2$ on the left and the lasso regularizer $q = 1$ on the right, in which the optimum value for the parameter vector $\mathbf{w}$ is denoted by $\mathbf{w}^\star$. The lasso gives a sparse solution in which $w_1^\star = 0$.



3.1.5 Multiple outputs

$\mathbf{x} \longrightarrow \Phi(\mathbf{x}) \longrightarrow W^T \Phi(\mathbf{x})$

$\boxed{K > 1}$ dimensional target vector $\mathbf{t} = \begin{pmatrix} t_1 \\ \vdots \\ t_K \end{pmatrix}$

Our approach is to use the same basis functions to model all of the components of target vector

$$\mathbf{y}(\mathbf{x}, W) := W^T \underset{M\text{-dim}}{\Phi(\mathbf{x})} = \begin{pmatrix} y_1 \\ \vdots \\ y_K \end{pmatrix}$$

where $\mathbf{y}$ is a K-dim vector, W is an $M \times \boxed{K}$ matrix of parameters and $\Phi(\mathbf{x})$ is an M-dim vector with elements $\phi_j(\mathbf{x})$

Suppose conditional distribution of the target vector to be an istropic Gaussian

$$p(\mathbf{t} \mid \mathbf{x}, W, \overset{\text{single value}}{\underline{\beta}}) = \mathcal{N}(\mathbf{t} \mid W^{T} \Phi(\mathbf{x}), \boxed{\beta^{-1} I})$$

Given k-dim N observations $\mathbf{t}_1, \mathbf{t}_2, .. \mathbf{t}_N$. we can combine these into $N \times K$ matrix $\mathbb{T}$. (target)

Similarly combine the input vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_N$ into $N \times D$ matrix $\mathbb{X}$.

The log likelihood

$$\ln p(\mathbb{T} | \mathbb{X}, \mathbb{W}, \beta) = \sum_{n=1}^{N} \ln \mathcal{N}(\mathbf{t}_n | \mathbb{W}^T \Phi(\mathbf{x}_n), \beta^{-1} I)$$

$$= \frac{NK}{2} \ln\left(\frac{\beta}{2\pi}\right) - \frac{\beta}{2} \sum_{n=1}^{N} \| \mathbf{t}_n - \mathbb{W}^T \Phi(\mathbf{x}_n) \|^2$$

Maximization solution for $\mathbb{W}$ is given by

$$\mathbb{W}_{ML} = (\underset{M\times N}{\Phi^T} \underset{N\times M}{\Phi})^{-1} \overset{M\times N}{\Phi^T} \overset{N\times K}{\mathbb{T}} \qquad M \times K \text{ matrix}$$

If we examine this result for each target variable t_k,

$$\mathbf{w}_k = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{t}_k = \overbrace{\Phi^{\dagger}}^{\text{basis function values of observations}} \underbrace{\mathbf{t}_k}_{\text{target of observations}}$$

where $\mathbf{t}_k$ is an N - dim column vector (k^{th} column of $\mathbf{T}$)

Thus, the solutions decouples between the different targets.

From now on, we will consider single target variable t.

$\mathbf{x} \in \mathbb{R}^D$ input, basis function $\Phi(\cdot) = \begin{pmatrix} 1 \\ \phi_1(\mathbf{x}) \\ \vdots \\ \phi_{M-1}(\mathbf{x}) \end{pmatrix}$

target t, determine $\overset{M\times 1}{\mathbf{w}}$

1-dim Gaussian

$y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \Phi(\mathbf{x}) \approx t$ $\quad (t \sim \mathcal{N}(t \mid y(\mathbf{x},\mathbf{w}), \beta))$

target $\mathbf{t}$ (K - dim), determine $\overset{M\times K}{W}$

K-dim Gaussian

$\mathbf{y}(\mathbf{x}, W) = W^T \Phi(\mathbf{x}) \approx \mathbf{t}$ $\quad (\mathbf{t} \sim \mathcal{N}(\mathbf{t} \mid \mathbf{y}(\mathbf{x},\mathbf{w}), \beta I))$

3.2 The Bias - Variance Decomposition (모델 평가)

Frequentist view of model complexity

Bias - Variance trade-off

When error function is SSE, the optimal prediction is given by

$$h(x) = \mathbb{E}_t[t \mid x] = \int t \, p(t \mid x) \, dt$$

SSE $L(x,t)$

We showed in Section 1.5.5 that the expected squared loss can be written in the form

$$\mathbb{E}[L] = \int \{ \overset{\text{Prediction}}{y(x)} - \overset{\text{optimal solution}}{h(x)} \}^2 p(x) \, dx + \iint \{ h(x) - t \}^2 p(x,t) \, dx \, dt \qquad (3.37)$$

$$\mathbb{E}[L] = \int \{ \underset{\underset{y(\mathbf{x},\mathbf{w})}{\uparrow}}{\overset{\text{Prediction}}{y(\mathbf{x})}} - \overset{\text{optimal solution}}{h(\mathbf{x})} \}^2 p(\mathbf{x})\, d\mathbf{x} + \int \{ h(\mathbf{x}) - t \}^2 p(\mathbf{x},t)\, d\mathbf{x}\, dt$$

The second term arises from the intrinsic noise and is the minimum expected loss.

The first term depends on our choice of $y(\mathbf{x})$

Our goal is to seek $y(\mathbf{x})$ making the first term a minimum.

유한한 N개의 관측

$$\Rightarrow \quad D \rightarrow \underset{\text{매개변수}}{\mathbf{w}_{ML}} \longrightarrow \underset{y(\mathbf{x},\mathbf{w}_{ML})}{\overset{y(\mathbf{x};D)}{\parallel}}$$

modeling $h(\mathbf{x})$ using $y(\mathbf{x},\mathbf{w})$

Bayesian : uncertainty is expressed through a posterior distribution over $\mathbf{w}$

Frequentist : point estimate of $\mathbf{w}$ based on D.

N observations D are independently drawn from $p(t, \mathbf{x})$

For a given D, we can obtain a prediction function $y(\mathbf{x}; D)$

$y(\mathbf{x}; D)$ and its squared error depend on D

The performance of learning algorithm is assessed by taking the average over ensemble of data sets

Consider the first term in (3.37)

$$\{ \underbrace{y(\mathbf{x};D)}_{\substack{\text{prediction by } D \\ \text{depented by ML algorithm}}} - \underbrace{h(\mathbf{x})}_{\text{optimal solution}} \}^2$$

which depends on D.

$$\{ y(\mathbf{x};D) \pm \mathbb{E}_D[y(\mathbf{x};D)] - h(\mathbf{x}) \}^2$$

$$= \{ y(\mathbf{x};D) - \mathbb{E}_D[y(\mathbf{x};D)] \}^2 + \{ \mathbb{E}_D[y(\mathbf{x};D)] - h(\mathbf{x}) \}^2$$

$$+ 2\{ y(\mathbf{x};D) - \mathbb{E}_D[y(\mathbf{x};D)] \}\{ \mathbb{E}_D[y(\mathbf{x};D)] - h(\mathbf{x}) \}$$

Take the expectation w.r.t D.

$$\mathbb{E}_D[\{y(\mathbf{x};D)-h(\mathbf{x})\}^2] = \underbrace{\{\mathbb{E}_D[y(\mathbf{x};D)]-h(\mathbf{x})\}^2}_{\text{bias}^2}$$

$$+ \underbrace{\mathbb{E}_D[\{y(\mathbf{x};D)-\mathbb{E}_D[y(\mathbf{x};D)]\}^2]}_{\text{variance}}$$

D

bias : extent to which the average prediction over all data sets differs from the desired regression function

variance : extent to which the solutions for individual data sets vary around thier average.

(sensitivity of $y(\mathbf{x};D)$ w.r.t the choice of D)

We can obtain the following decomposition of expected squared loss

$$\text{expected squared loss} = (\text{bias})^2 + \text{variance} + \text{noise}$$

where $(\text{bias})^2 = \int \{ \mathbb{E}_D[y(\mathbf{x};D)] - h(\mathbf{x}) \}^2 p(\mathbf{x})\, d\mathbf{x}$

$$\text{variance} = \int \mathbb{E}_D\left[\{ y(\mathbf{x};D) - \mathbb{E}_D[y(\mathbf{x};D)] \}^2 \right] p(\mathbf{x})\, d\mathbf{x}$$

$$\text{noise} = \int \{ h(\mathbf{x}) - t \}^2 p(\mathbf{x},t)\, d\mathbf{x}\, dt$$

Our goal is to minimize the expected loss.

Trade - off between bias and variance

Flexible models having high variance and low bias

Rigid models having low variance and high bias

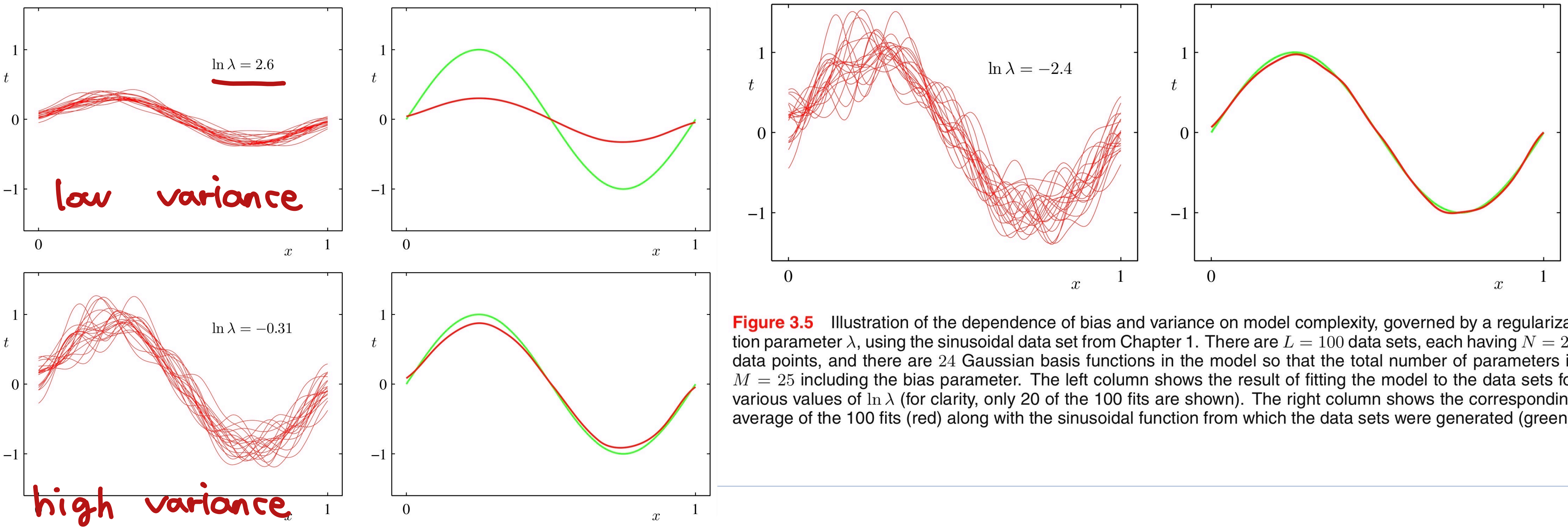


Figure 3.5 Illustration of the dependence of bias and variance on model complexity, governed by a regularization parameter λ, using the sinusoidal data set from Chapter 1. There are $L = 100$ data sets, each having $N = 25$ data points, and there are 24 Gaussian basis functions in the model so that the total number of parameters is $M = 25$ including the bias parameter. The left column shows the result of fitting the model to the data sets for various values of $\ln \lambda$ (for clarity, only 20 of the 100 fits are shown). The right column shows the corresponding average of the 100 fits (red) along with the sinusoidal function from which the data sets were generated (green).

Examine the bias - variance trade-off quantitavely

L prediction models $y^{(l)}$, $l = 1, \dots L$

The average prediction $D_1 \dots D_L$

$$\bar{y}(x) := \frac{1}{L} \sum_{l=1}^{L} y^{(l)}(x)$$

and integrated squared bias and integrated variance

$$(\text{bias})^2 = \frac{1}{N} \sum_{n=1}^{N} \{ \bar{y}(x_n) - h(x_n) \}^2$$

(approximated by sum of x_n)

$$\text{variance} = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{L} \sum_{l=1}^{L} \{ y^{(l)}(x_n) - \bar{y}(x_n) \}^2$$

3.3 Bayesian Linear regression

Bayesian linear regression can avoid the overfitting problem of maximum likelihood and lead to automatic methods of determining model complexity.

$$t \sim \mathcal{N}(t|\underline{y(x,w)}, \beta^{-1})$$

$$\begin{tikzcd}
x^* \arrow[rr] \arrow[dr] & & \text{predict } t^* \\
& \text{Parameter distribution} \arrow[ur] &
\end{tikzcd}$$

3.3.1 Parameter distribution

Consider the prior probability distribution over w

Noise precision parameter β is assumed to be known

First, we noted likelihood $p(t|w)$ is the exponential of quadratic of w

So the corresponding conjugate prior is given by

$$p(\mathbf{w}) := \mathcal{N}(\mathbf{w} \mid \mathbf{m}_0, S_0)$$

$\exp(w^2)$ w 가우시안

posterior $\propto$ likelihood · prior

$$\prod_{n=1}^{N} \mathcal{N}(t_n \mid \mathbf{w}^T \Phi(\mathbf{x}_n), \beta^{-1})$$

where mean $\mathbf{m}_0$, covariance S_0.

Thus the posterior distribution in the form (see (2.116))

$$p(\mathbf{w} \mid \mathbf{t}) = \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, S_N)$$

where

$$\mathbf{m}_N = S_N (S_0^{-1} \mathbf{m}_0 + \beta \Phi^T \mathbf{t})$$

$$S_N^{-1} = S_0^{-1} + \beta \Phi^T \Phi$$

Since the posterior is Gaussian (unimodal), its mode = mean

$\Rightarrow \quad \mathbf{w}_{MAP} = \mathbf{m}_N$

If we consider $S_0 := \alpha^{-1} I$ and $\alpha \to 0$ i.e. infinitely broad prior, then the mean $\mathbf{m}_N$ reduces to $\mathbf{w}_{ML}$

Similarly, if $N = 0$ (without observation), posterior = prior

For simplicity, consider a zero - mean isotropic Gaussian with single precision parameter α as a prior distribution

$$p(\mathbf{w} | \alpha) = \mathcal{N}(\mathbf{w} | \mathbf{0}, \alpha^{-1} I) \quad \text{(simple version)}$$

So the corresponding posterior

$$p(\mathbf{w} | \mathbf{t}) = \mathcal{N}(\mathbf{w} | \mathbf{m}_N, S_N)$$

where

$$\mathbf{m}_N = \beta S_N \Phi^T \mathbf{t}$$

$$S_N^{-1} = \alpha I + \beta \Phi^T \Phi$$

Log of posterior is the sum of log of likelihood and log of prior

$$\ln \underset{\text{posterior}}{p(\mathbf{w} \mid \mathbf{t})} = -\frac{\beta}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^T \boldsymbol{\Phi}(\mathbf{x}_n)\}^2 - \frac{\alpha}{2}\mathbf{w}^T\mathbf{w} + \text{constant}$$

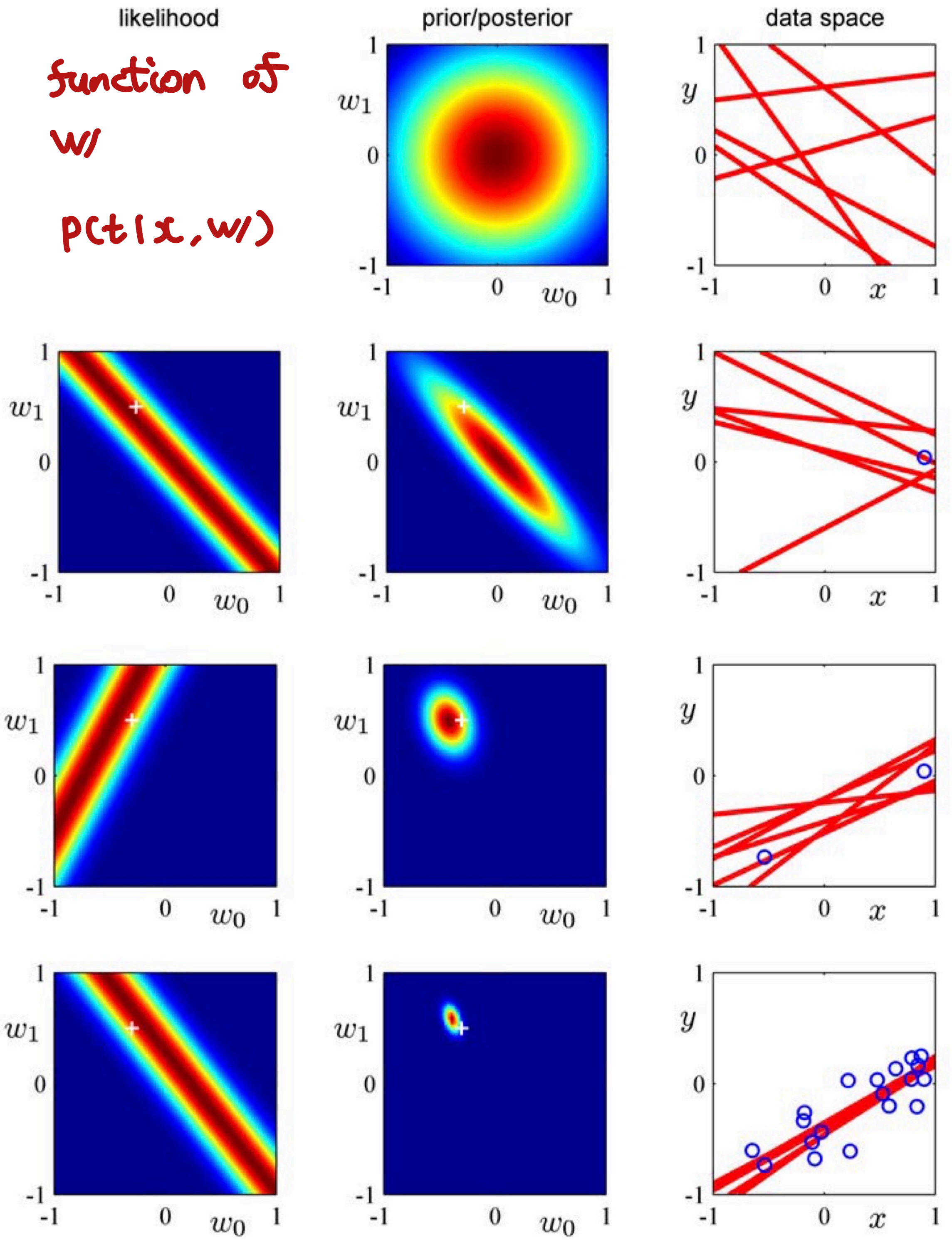
Its MAP solution w.r.t $\mathbf{w}$ is equivalent to minimization of SSE with additional quadratic regularization term $\lambda = \alpha/\beta$.

Linear basis function

- input: x
- target: t
- $y(x, w) := w_0 + w_1 x$
- Observed data ○

generated by $-0.3 + 0.5x$

with std 0.2



function of w

$p(t \mid x, w)$

prior $\alpha^{-1}\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$

(1, 0.1)

Figure 3.7 Illustration of sequential Bayesian learning for a simple linear model of the form $y(x, \mathbf{w}) = w_0 + w_1 x$. A detailed description of this figure is given in the text.

Generalized the Gaussian prior

$$p(w \mid \alpha) := \left[\frac{q}{2} \left(\frac{\alpha}{2} \right)^{1/q} \frac{1}{\Gamma(1/q)} \right]^{M} \exp\left(-\frac{q}{2} \sum_{j=0}^{M-1} |w_j|^q \right)$$

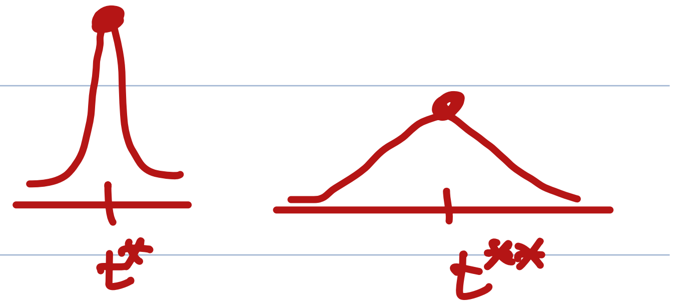
in which $q=2$ corresponds to the Gaussian.

If $q=2$, MAP solution of w is the minimization solution of (3.29) which is SSE + regularization term

If $q \neq 2$, it is not true. (mode of posterior $\neq$ mean)

3.3.2 Predictive distribution

In practice, we are interested in making predictions of t for a new $\mathbf{x}$ (not the value of $\mathbf{w}$)



Predictive distribution of t

$$p(t \mid \underset{\text{new input}}{\mathbf{x}}, \mathbf{t}, \alpha, \beta) = \int p(t \mid \mathbf{x}, \mathbf{w}, \beta)\, \underbrace{p(\mathbf{w} \mid \mathbf{t}, \alpha, \beta)}_{\text{w 에 대한 posterior}}\, d\mathbf{w}$$

where $\mathbf{t}$ is the vector of training target values.

α is from prior assumption, β is Gaussian noise of t

$$p(\mathbf{w} \mid \alpha) = \mathcal{N}(\mathbf{w} \mid 0, \alpha^{-1} I) \qquad p(t \mid \mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}(t \mid y(\mathbf{x}, \mathbf{w}), \beta^{-1})$$

The predictive distribution takes the form

$$p(t|\underset{\text{new input}}{\mathbf{x}}, \overset{\text{training data}}{\mathbf{t}}, \alpha, \beta) = \mathcal{N}(t| \mathbf{m}_N^T \boldsymbol{\Phi}(\mathbf{x}), \overset{\in \mathbb{R}}{\sigma_N^2(\mathbf{x})}))$$

where

$$\sigma_N^2(\mathbf{x}) = \underbrace{\frac{1}{\beta}}_{\text{data noise}} + \boldsymbol{\Phi}(\mathbf{x})^T \underset{\text{uncertainty of } \mathbf{w}}{S_N} \boldsymbol{\Phi}(\mathbf{x}) \qquad (3.59)$$

By [Qazaz et al., 1997]

$$\sigma_{N+1}^2(\mathbf{x}) \leq \sigma_N^2(\mathbf{x})$$

If $N \to \infty$, then the second term in (3.59) $\longrightarrow 1/\beta$

Uncertainty in predictions governed by (3.59)

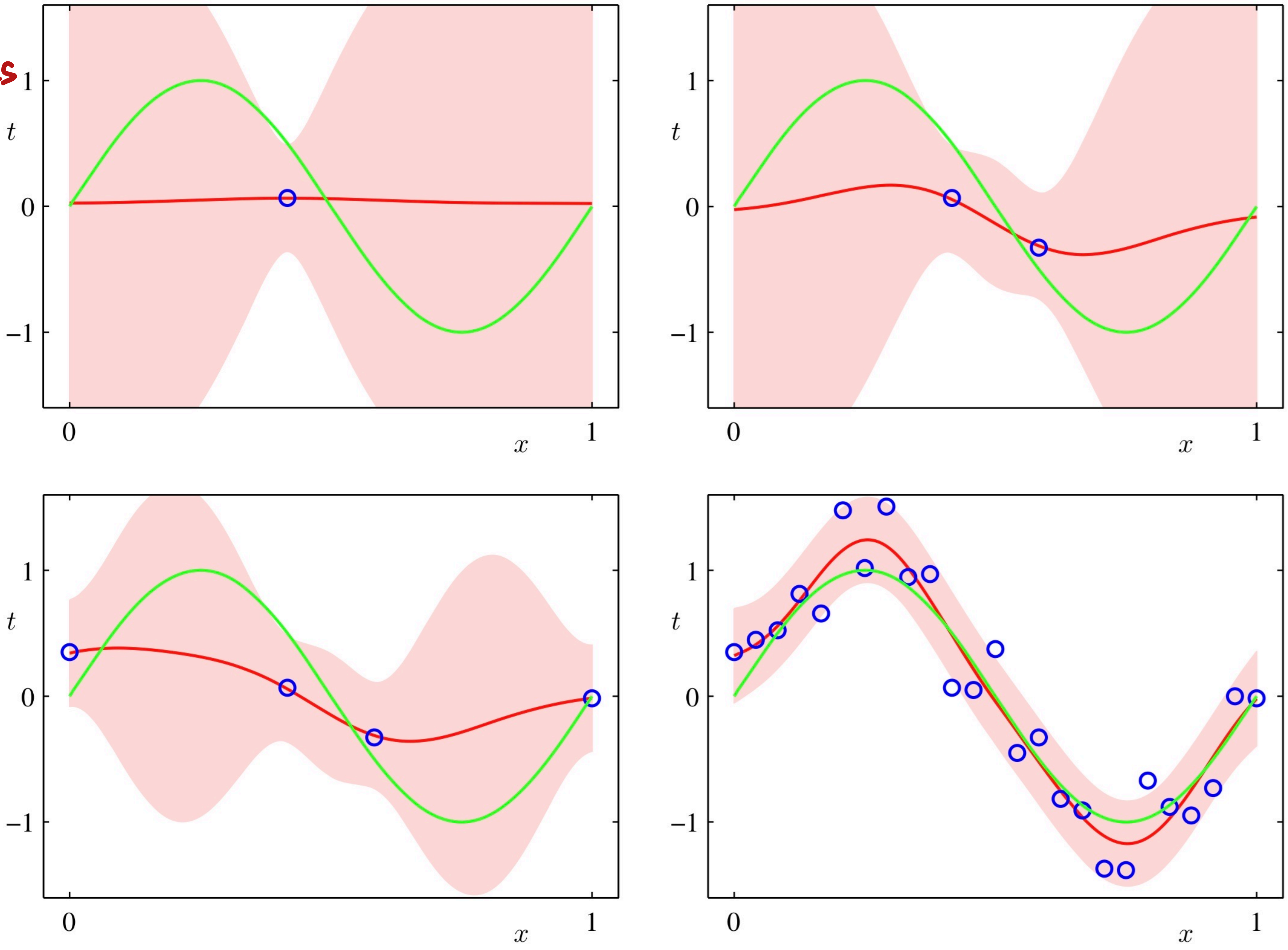


Figure 3.8 Examples of the predictive distribution (3.58) for a model consisting of 9 Gaussian basis functions of the form (3.4) using the synthetic sinusoidal data set of Section 1.1. See the text for a detailed discussion.

Sampling from posterior and plotting corresponding model

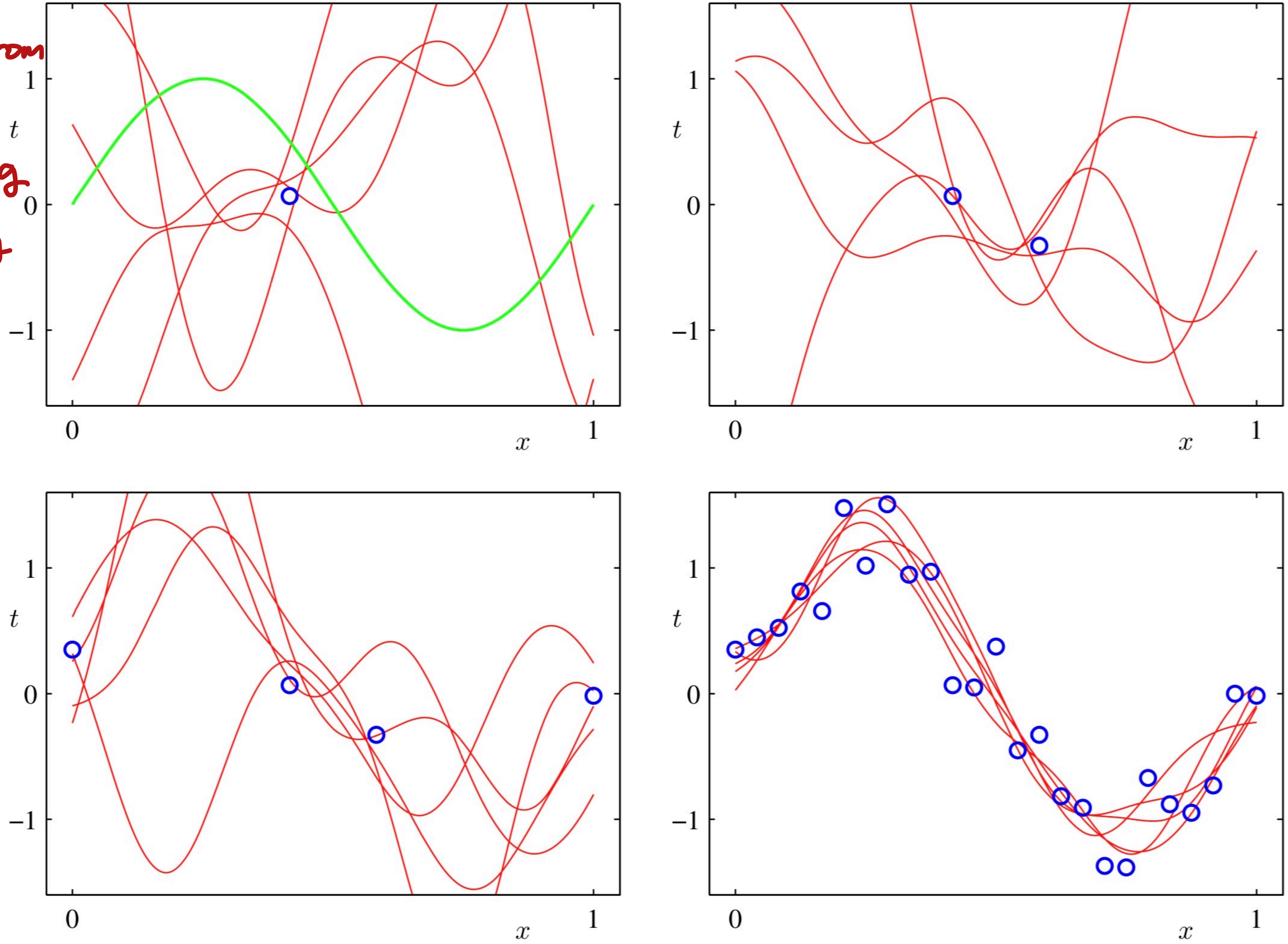


Figure 3.9 Plots of the function $y(x, \mathbf{w})$ using samples from the posterior distributions over $\mathbf{w}$ corresponding to the plots in Figure 3.8.

Remark

- We have used Gaussian basis function (localized)
- If $\mathbf{x}$ is away from the basis function centers, then the contribution from the second term in (3.59) goes to 0 i.e. left the noise β^{-1}.

3.3.3 Equivalent kernel (kernel method)

Substitute (3.53) into (3.3) (expected prediction)

$$y(\mathbf{x}, \mathbf{m}_N) = \overset{M\times 1}{\mathbf{m}_N^T} \boldsymbol{\Phi}(\mathbf{x}) = \beta\, \boldsymbol{\Phi}(\mathbf{x})^T S_N \overset{M\times N}{\boldsymbol{\Phi}^T} \mathbf{t} = \underline{\sum_{n=1}^{N} \beta\, \boldsymbol{\Phi}(\mathbf{x})^T S_N\, \boldsymbol{\Phi}(\mathbf{x}_n)\, t_n}$$

where $\boldsymbol{\Phi}(\mathbf{x}) := (\phi_0(\mathbf{x}), .. \phi_{M-1}(\mathbf{x}))^T$, $S_N^{-1} = S_0^{-1} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}$ and

$$\boldsymbol{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & & \phi_{M-1}(\mathbf{x}_1) \\ \vdots & \cdots & \\ \phi_0(\mathbf{x}_N) & & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix} \quad \text{design matrix}$$

Thus, $y(\mathbf{x}, \mathbf{m}_N)$ is the linear combination of the training set target variables t_n.

$$\Rightarrow \quad y(\mathbf{x}, \mathbf{m}_N) = \sum_{n=1}^{N} k(\mathbf{x}, \mathbf{x}_n)\, t_n$$

where the function

$$k(\mathbf{x}, \mathbf{x}') := \beta\, \Phi(\mathbf{x})^T S_N\, \Phi(\mathbf{x}')$$

is known as smoother matrix or equivalent kernel

Linear smoother : regression function makes predictions by taking linear combinations of training target values

This kernel depends on $\mathbf{x}_n$ because of S_N

Consider the covariance between $y(\mathbf{x})$ and $y(\mathbf{x}')$

$$\mathrm{cov}[y(\mathbf{x}), y(\mathbf{x}')] = \mathrm{cov}_{\mathbf{w}}[\mathbf{w}^T\Phi(\mathbf{x}), \mathbf{w}^T\Phi(\mathbf{x}')]$$

$$= \Phi(\mathbf{x})^T S_N \Phi(\mathbf{x}') = \beta^{-1} k(\mathbf{x}, \mathbf{x}')$$

$y(\mathbf{x})$ = mean of $\mathcal{N}(t \mid \mathbf{m}_N^T \Phi(\mathbf{x}), \beta^{-1} + \Phi(\mathbf{x})^T S_N \Phi(\mathbf{x}))$

$y(\mathbf{x}) = \mathbf{w}^T \Phi(\mathbf{x})$ where $\mathbf{w} = \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, S_N)$

$$\mathrm{cov}[\mathbf{w}^T\Phi(\mathbf{x}), \mathbf{w}^T\Phi(\mathbf{x}')] = \underbrace{\mathbb{E}[\Phi(\mathbf{x})^T \mathbf{w}\mathbf{w}^T \Phi(\mathbf{x}')]}_{} - \Phi(\mathbf{x})^T \mathbf{m}_N \mathbf{m}_N^T \Phi(\mathbf{x}')$$

$$= \Phi(\mathbf{x})^T \underbrace{\mathbb{E}[\mathbf{w}\mathbf{w}^T]}_{} \Phi(\mathbf{x}')$$

$$= \mathrm{cov}[\mathbf{w}] + \mathbb{E}[\mathbf{w}]\mathbb{E}[\mathbf{w}]^T$$

$$= S_N + \mathbf{m}_N \mathbf{m}_N^T$$

For regression, we introduced a set of basis functions so equivalent kernel was implicitly determined. But we can define a localized kernel directly and use this to make predictions

The equivalent kernel (3.62) can be expressed in the form an inner product w.r.t $\boldsymbol{\psi}(\mathbf{x})$ of nonlinear functions

$$k(\mathbf{x}, \mathbf{z}) := \boldsymbol{\psi}(\mathbf{x})^T \boldsymbol{\psi}(\mathbf{z})$$

where $\boldsymbol{\psi}(\mathbf{x}) := \beta^{1/2} S_N^{1/2} \boldsymbol{\phi}(\mathbf{x})$.

3.4 Bayesian Model Comparison

The problem of model selection from a Bayesian perspective.
The over-fitting associated with Maximum likelihood can be avoided by marginalizing over model parameters.
The Bayesian view of model comparison involves the use of probabilities to represent uncertainty in the choice of model.

Compare a set of L models $\{M_i\}_{i=1,\dots L}$

Model refers to a probability distribution over the observed data D

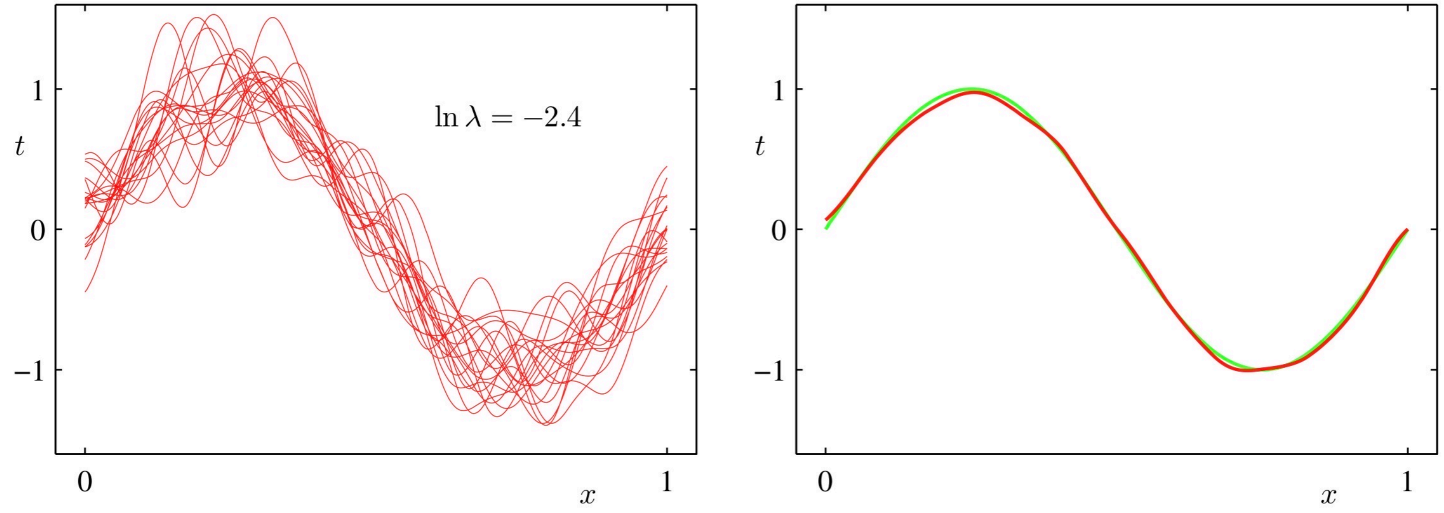


Figure 3.5 Illustration of the dependence of bias and variance on model complexity, governed by a regularization parameter λ, using the sinusoidal data set from Chapter 1. There are $L = 100$ data sets, each having $N = 25$ data points, and there are 24 Gaussian basis functions in the model so that the total number of parameters is $M = 25$ including the bias parameter. The left column shows the result of fitting the model to the data sets for various values of $\ln\lambda$ (for clarity, only 20 of the 100 fits are shown). The right column shows the corresponding average of the 100 fits (red) along with the sinusoidal function from which the data sets were generated (green).

Suppose that the data is generated from one of models but we are uncertain which one

This uncertainty is expressed through $p(M_i)$

Given a training data set D, we want to evaluate

$$p(M_i | D) \propto \underset{\text{prior}}{p(M_i)}\ \underset{\text{model evidence}}{p(D | M_i)}$$

θ

$p(D|\theta)$

The prior can express a preference for different models. But for simplicity assume that all models have the same prior.

$p(D|M_i)$ model evidence (marginal likelihood) expresses the preference shown by the data for different models.

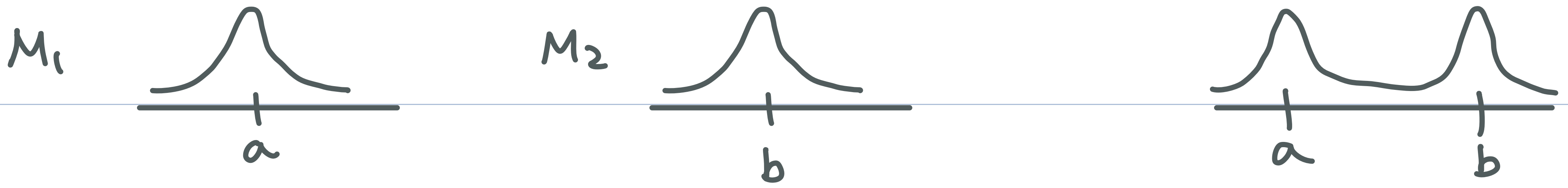
(likelihood function over the model space in which the parameters have been marginalized out)

The predictive distribution (mixture distribution)

$$p(t \mid \mathbf{x}, D) = \sum_{i=1}^{L} \boxed{p(\boxed{t} \mid \mathbf{x}, \boxed{M_i}, D)}\, \overset{\text{posterior over models}}{p(\boxed{M_i} \mid \underline{D})}$$

Average of the predictive distributions $p(t \mid \mathbf{x}, M_i, D)$ of individual models weighted by the posterior probabilities $p(M_i \mid D)$

For example, two models



Model selection: use the single most probable model alone.

Consider model M_i governed by the parameter w. The model evidence is given by

$$p(D \mid M_i) = \int p(D \mid w, M_i)\, p(w \mid M_i)\, dw$$

The model evidence (marginal likelihood) $p(D \mid M_i)$ can be viewed as the probability of generating the data set D from a model whose parameters are sampled at random from the prior

Note that

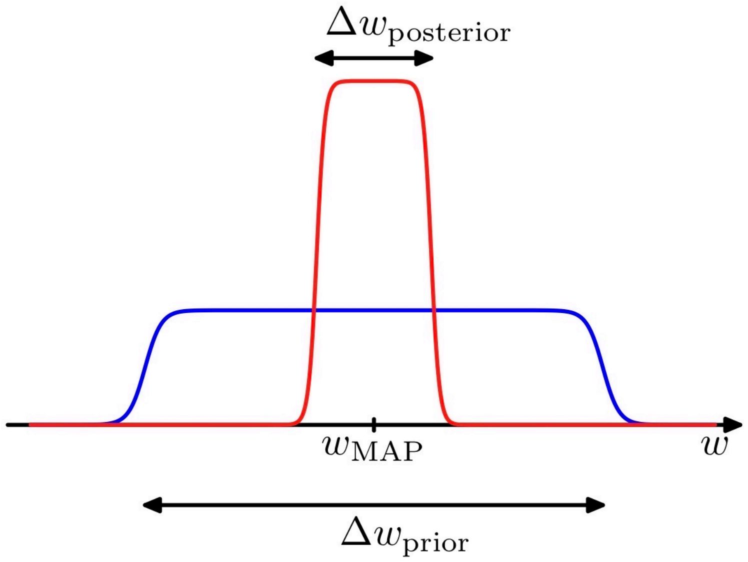
$$p(w \mid D, M_i) = \frac{p(D \mid w, M_i)\, p(w \mid M_i)}{\boxed{p(D \mid M_i)}}$$

The model evidence is the normalization term appearing in the denominator in Bayes Theorem when evaluating the posterior w

Consider a single parameter w. The posterior distribution over w is proportional to $p(D|w)p(w)$.

For simplicity, assume the posterior distribution is sharply peaked around the most probable value w_{MAP} with width $\Delta w_{posterior}$

Figure 3.12 We can obtain a rough approximation to the model evidence if we assume that the posterior distribution over parameters is sharply peaked around its mode w_{MAP}.



and the prior is flat with width Δw_{prior} so that

$p(w) = 1/\Delta w_{prior}$.

Thus we have a simple approximation to the integral over w

$$p(D) = \int p(D|w)\, p(w)\, dw \simeq p(D|w_{MAP}) \frac{\Delta w_{posterior}}{\Delta w_{prior}}$$

So

$$\ln p(D) \simeq \underbrace{\ln p(D|w_{MAP})}_{\text{fit to data given by the most probable parameter}} + \underbrace{\ln\left(\frac{\Delta w_{posterior}}{\Delta w_{prior}}\right)}_{\text{penalty of the model complexity}}$$

$\Delta w_{posterior} < \Delta w_{prior}$, then the second term is negative.

So it increases in magnitude as the ratio $\Delta w_{post} / \Delta w_{prior}$ gets smaller. If the parameters are finely tuned to the data in posterior, then the penalty term is large.

For a model with M parameters, assume all parameters have the same ratio of $\Delta w_{posterior} / \Delta w_{prior}$, then we obtain a similar approximation as follows

$$\ln p(D) \simeq \ln p(D \mid w_{MAP}) + M \ln \left(\frac{\Delta w_{posterior}}{\Delta w_{prior}} \right)$$

Thus, the size of the complexity penalty increases linearly

3.5 The evidence Approximation

Fully Bayesian treatment of linear basis function model

- Introduce prior distributions over hyperparameters α and β
- Make predictions by marginalizing w.r.t these hyperparameters and parameters w
- But the complete marginalization over all of these variables α, β and w is analytically intractable.

Discuss an approximation in which we set the hyperparameters to specific values determined by maximizing the 'marginal likelihood function' obtained by first integrating over $\mathbf{w}$.

If we introduce hyperprior over α and β, the predictive distribution is given by

$$p(t|\mathbf{t}) = \iiint \underbrace{p(t|\mathbf{w},\beta)}_{\substack{\text{model assumption}\\(3.8)}} \overbrace{p(\mathbf{w}|\mathbf{t},\alpha,\beta)}^{\substack{(3.49)\\\text{posterior over } \mathbf{w}}} \underbrace{p(\alpha,\beta|\mathbf{t})}_{\text{hyper posterior}} \, d\mathbf{w} \, d\alpha \, d\beta$$

Here we omitted the dependence on input $\mathbf{x}$.

If posterior $p(\alpha, \beta \mid \mathbf{t})$ is sharply peaked around $\hat{\alpha}$ and $\hat{\beta}$, then

$$p(t \mid \mathbf{t}) \simeq p(t \mid \mathbf{t}, \hat{\alpha}, \hat{\beta}) = \int p(t \mid \mathbf{w}, \hat{\beta}) \, p(\mathbf{w} \mid \mathbf{t}, \hat{\alpha}, \hat{\beta}) \, d\mathbf{w}$$

From Bayes Theorem, the posterior distribution for α, β

$$p(\alpha, \beta \mid \mathbf{t}) \propto p(\mathbf{t} \mid \alpha, \beta) \, p(\alpha, \beta)$$

So if prior is relatively flat, the values $\hat{\alpha}$ and $\hat{\beta}$ are obtained by maximizing the marginal likelihood function $p(\mathbf{t} \mid \alpha, \beta)$

Here, we evaluate the marginal likelihood for the linear basis model and then finding its maxima.

So this will allow us to determine values for hyperparameters from the training data alone. ($\alpha/\beta \simeq$ regularization parameter)

Two approaches of maximization of the log evidence

- Evaluate the evidence function analytically and then set its derivative equal to 0 to obtain re-estimation for α, β
- Use the technique called expectation maximization algorithm in Section 9.3.4.

3.5.1 Evaluation of the evidence function

The marginal likelihood function $p(\mathbf{t} | \alpha, \beta)$ is obtained by integrating over $\mathbf{w}$

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

By the result (2.115) for the conditional distribution in a linear - Gaussian model, we can evaluate this integral.

From, (3.11), (3.12) and (3.52), we can write the evidence function in the form (Excercise 3.17)

$$p(t \mid \alpha, \beta) = \left(\frac{\beta}{2\pi}\right)^{N/2} \left(\frac{\alpha}{2\pi}\right)^{M/2} \int \exp\{-E(w)\}\, dw \qquad (3.78)$$

where M is the dimensionality of w and

$$\begin{aligned} E(w) &= \beta E_D(w) + \alpha E_W(w) \\ &= \frac{\beta}{2} \| \underset{N}{t} - \underset{(N\times M)(M\times 1)}{\Phi w} \|^2 + \frac{\alpha}{2} w^T w \end{aligned} \qquad (3.79)$$

$$w \sim N(m_N, S_N^{-1})$$

Furthermore,

$$E(\mathbf{w}) = E(\mathbf{m}_N) + \frac{1}{2}(\mathbf{w} - \mathbf{m}_N)^T A (\mathbf{w} - \mathbf{m}_N)$$

where we have introduced

$$A = \alpha I + \beta \Phi^T \Phi$$

together with

$$E(\mathbf{m}_N) = \frac{\beta}{2} \| \mathbf{t} - \Phi \mathbf{m}_N \|^2 + \frac{\alpha}{2} \mathbf{m}_N^T \mathbf{m}_N$$

A is the matrix of second derivatives of error function and a.k.a Hessian matrix

$$A = \nabla \nabla E(\mathbf{w}).$$

Here we have also defined $\mathbf{m}_N$ given by

$$\mathbf{m}_N = \beta A^{-1} \Phi^T \mathbf{t}. \qquad (3.84)$$

$S_N^{-1} = \alpha I + \beta \Phi^T \Phi$

Using (3.54), we see $A = S_N^{-1}$, hence (3.84) = (3.53)

Back to the integration (3.78)

$$\int \exp(-E(\mathbf{w}))\,d\mathbf{w}$$

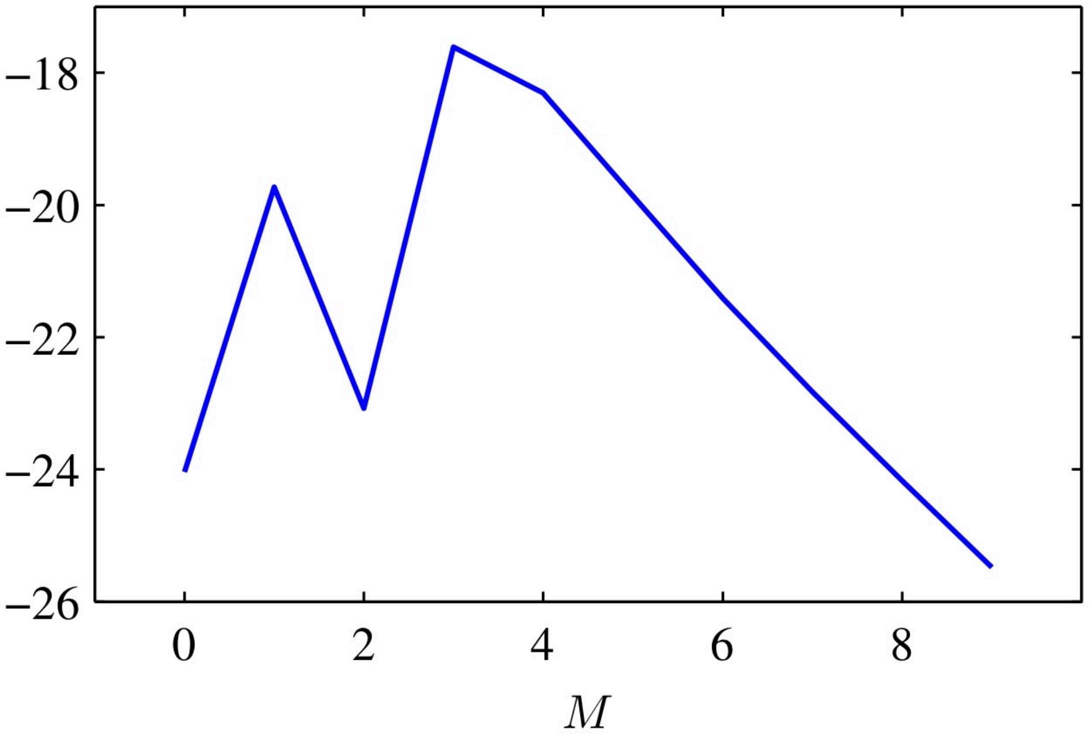
$$= \exp\{-E(\mathbf{m}_N)\}\int \exp\left\{-\frac{1}{2}(\mathbf{w}-\mathbf{m}_N)^T A(\mathbf{w}-\mathbf{m}_N)\right\}d\mathbf{w}$$

$$= \exp\{-E(\mathbf{m}_N)\}(2\pi)^{M/2}|A|^{-1/2}$$

Using (3.78), we can write the log of the marginal likelihood in the form

$$\ln p(\mathbf{t}|\alpha,\beta) = \frac{M}{2}\ln\alpha + \frac{N}{2}\ln\beta - E(\mathbf{m}_N) - \frac{1}{2}\ln|A| - \frac{N}{2}\ln(2\pi)$$

Figure 3.14 Plot of the log model evidence versus the order M, for the polynomial regression model, showing that the evidence favours the model with $M = 3$.



$\ln p(\mathbf{t}|\alpha,\beta)$

$\alpha = 5\cdot 10^{-3}$

Remark

- The underlying sinusoidal function is an odd function
- In $M=3$ case, we obtain a significant improvement in data fit.

3.5.2 Maximizing the evidence function

Consider the maximization of $p(\mathbf{t} | \alpha, \beta)$ w.r.t α.

This can be done by first defining the following eigenvector equation

$$(\beta \Phi^T \Phi) u_i = \lambda_i u_i$$

Since $A = \alpha I + \beta \Phi^T \Phi$, A has eigenvalues $\alpha + \lambda_i$

Now consider the partial derivative of $\ln |A|$ w.r.t α.

We have

$$\frac{\partial}{\partial\alpha}\ln|A| = \frac{\partial}{\partial\alpha}\ln\prod_i(\lambda_i+\alpha) = \frac{\partial}{\partial\alpha}\sum_i\ln(\lambda_i+\alpha) = \sum_i\frac{1}{\lambda_i+\alpha}$$

Thus,

$$0 = \frac{M}{2\alpha} - \frac{1}{2}\mathbf{m}_N^T\mathbf{m}_N - \frac{1}{2}\sum_i\frac{1}{\lambda_i+\alpha}$$

Multiplying by 2α and rearranging, we obtain

$$\alpha\,\mathbf{m}_N^T\mathbf{m}_N = M - \alpha\sum_i\frac{1}{\lambda_i+\alpha} =: \gamma$$

Since there are M terms in the sum over i,

(3.91) $$\gamma = \sum_i \frac{\lambda_i}{\alpha + \lambda_i}$$ (depends on α)

So the following α maximizes the marginal likelihood

$$\alpha = \frac{\gamma}{\mathbf{m}_N^T \mathbf{m}_N} \quad (3.92)$$

Note that γ depends on α and the mode $\mathbf{m}_N$ of the posterior distribution depends on the choice of α.

Thus, this solution is implicit and is adopted an iterative procedure

Make an initial choice of α and use this to find m_N (3.53) and evaluate γ (3.91)

Using (3.92), re-estimate α and the process repeat until convergence.

Note that because the matrix $\Phi^T \Phi$ is fixed, we can compute its eigenvalues once at the start

The value of α has been determined purely by training data.

Similarly, maximize the log marginal likelihood (3.86) w.r.t β.

Note that the eigenvalues λ_i are proportional to β

and hence $d\lambda_i/d\beta = \lambda_i/\beta$ giving

$$\frac{\partial}{\partial\beta}\ln|A| = \frac{\partial}{\partial\beta}\sum_i \ln(\lambda_i+\alpha) = \frac{1}{\beta}\sum_i \frac{\lambda_i}{\lambda_i+\alpha} = \frac{\gamma}{\beta}$$

So the stationary point of the marginal likelihood satisfies

$$0 = \frac{N}{2\beta} - \frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{m}_N^T\Phi(\mathbf{x}_n)\}^2 - \frac{\gamma}{2\beta}$$

and rearranging we obtain

$$\frac{1}{\beta} = \frac{1}{N-\gamma} \sum_{n=1}^{N} \{ t_n - \mathbf{m}_N^T \boldsymbol{\Phi}(\mathbf{x}_n) \}^2$$

Again, this is an implicit solution for β. So choose an initial value for β and calculate $\mathbf{m}_N$ and γ and then re-estimate β using (3.95), repeating until convergence.

3.6 Limitations of Fixed Basis functions

Models comprising a linear combination of fixed nonlinear basis functions.

The assumption of linearity in the $\overset{w}{\text{parameters}}$ led to a range of useful properties including closed-form solutions to the least squares problem. We can model arbitrary nonlinearities in the mapping from inputs to targets.

But there are some significant shortcomings

The basis functions $\phi_j(\mathbf{x})$ are fixed before the training data is observed.

The number of basis functions needs to grow rapidly with the dimensionality D.