1.2.6 Bayesian curve fitting

Note: the Gaussian curve problem demonstrated, can be/is solved analytically

In a fully Bayesian approach, we should consistently apply the sum and product rules of probability, which requires, as we shall see shortly, that we integrate over all val- ues of w. Such marginalizations lie at the heart of Bayesian methods for pattern recognition.

given the training data **x** and **t**, along with a new test point x, and our goal is to predict the value of t.

we assume that the parameters α and β are fixed and known in advance (in later chapters we shall discuss how such parameters can be inferred from data in a Bayesian setting).

is the posterior distribution

predictive

distribution

$$p(t|x, \mathbf{x}, \mathbf{t}) = \int p(t|x, \mathbf{w})p(\mathbf{w}|\mathbf{x}, \mathbf{t})\, d\mathbf{w}. \qquad (1.68)$$

can be found by normalizing the right-hand side of (1.66).

$$p(\mathbf{w}|\mathbf{x}, \mathbf{t}, \alpha, \beta) \propto p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta)p(\mathbf{w}|\alpha). \qquad (1.66)$$

x = test data
x = training data

posterior ∝ likelihood × prior

omitted the dependence on α and β to simplify the notation

likelihood

target values $\mathbf{t} = (t_1, \ldots, t_N)^{\mathrm{T}}$. *(sometimes t is labels)*

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j \qquad (1.1)$$

$$p(t|x, \mathbf{w}, \beta) = \mathcal{N}\left(t|y(x, \mathbf{w}), \beta^{-1}\right) \qquad (1.60)$$

β corresponding to the inverse variance of the distribution.

1.2.6 Bayesian curve fitting (cont.)

solving 1.60:

☐ solve for $\mathbf{w}_{\mathrm{ML}}$ by minimizing the negative log-likelihood:

$$\ln p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta) = -\frac{\beta}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{N}{2} \ln \beta - \frac{N}{2} \ln(2\pi). \quad (1.62)$$

$$\frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2 = 0$$

where $y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$

One can use regularization to avoid overfitting by a polynomial order that is too high:

$$\widetilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

where $\|\mathbf{w}\|^2 \equiv \mathbf{w}^{\mathrm{T}}\mathbf{w} = w_0^2 + w_1^2 + \ldots + w_M^2$, and the coefficient λ governs the relative importance of the regularization term compared with the sum-of-squares error term. Note that often the coefficient w_0 is omitted from the regularizer because its

☐ solve for β_{ML}

$$\frac{1}{\beta_{\mathrm{ML}}} = \frac{1}{N} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}_{\mathrm{ML}}) - t_n\}^2. \quad (1.63)$$

1.2.6 Bayesian curve fitting (cont.)

then 1.60 becomes 1.64:

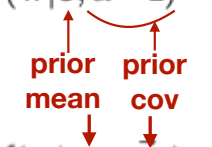$$p(t|x, \mathbf{w}, \beta) = \mathcal{N}\left(t|y(x,\mathbf{w}), \beta^{-1}\right) \tag{1.60}$$

$$p(t|x, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}}) = \mathcal{N}\left(t|y(x,\mathbf{w}_{\mathrm{ML}}), \beta_{\mathrm{ML}}^{-1}\right). \tag{1.64}$$

Can be expressed using the parameters w and basis functions ϕ: $Norm(t_n \mid \vec{w}^T \vec{\phi}(\vec{x}_n), \beta^{-1})$

for the prior of 1.66 consider a Gaussian over the polynomial coefficients of w:

$$p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I}) = \left(\frac{\alpha}{2\pi}\right)^{(M+1)/2} \exp\left\{-\frac{\alpha}{2}\mathbf{w}^T\mathbf{w}\right\} \tag{1.65}$$

prior mean   prior cov

$$= \mathcal{N}(\mathbf{w}|\vec{m}_0, \vec{S}_0)$$

where α is the precision of the distribution, and $M+1$ is the total number of elements in the vector $\mathbf{w}$ for an Mth order polynomial. Variables such as α, which control

solve for $\mathbf{W}$ by minimizing the negative log-likelihood of 1.66:

$$\frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\alpha}{2}\mathbf{w}^T\mathbf{w}. \tag{1.67}$$

Thus we see that maximizing the posterior distribution is equivalent to minimizing the regularized sum-of-squares error function encountered earlier in the form (1.4), with a regularization parameter given by $\lambda = \alpha/\beta$.

Integration of 1.68 can be solved analytically:

predictive distribution

$$p(t|x, \mathbf{x}, \mathbf{t}) = \int p(t|x, \mathbf{w}) p(\mathbf{w}|\mathbf{x}, \mathbf{t}) \, d\mathbf{w}. \qquad (1.68)$$

$$= \underbrace{\mathcal{N}(t_n \mid \vec{w}^T \vec{\phi}(\vec{x}_n), \beta^{-1})}_{\text{likelihood}} \; \underbrace{\mathcal{N}(\mathbf{w} \mid \vec{m}_0, \vec{S}_0)}_{\substack{\text{prior} \quad \text{prior} \\ \text{mean} \quad \text{cov}}}$$

$$= Norm(\vec{w} \mid \vec{m}_N, \vec{S}_N)$$

f_0: m0, s0
f_1: w^T * phi, beta^(-1)

** Note: Gram Matrix is in section 6.1 of Bishop's PRML.

sn^-1 = s0^-1 + beta * phiXT * phiX
    = s0^-1 + (s1^-1) * m1
is the weighted sums in quadrature of the variances which is derived from using partial differential of where the probability distribution is max
(and least sum squares is smallest):
    (d/dx) of summation_((x_i - μ)^2/(2*s_i^2)) = 0

    sigma^2 = (df0/dx)^2 * sigma0^2 + (df1/dx)^2 * sigma1^2
        where sigma^2 = s

mn:
    sn * mn = s0 * m0 + s1 * m1

Integration of 1.68 can be solved analytically:

predictive distribution

$$p(t|x, \mathbf{x}, \mathbf{t}) = \int p(t|x, \mathbf{w}) \, p(\mathbf{w}|\mathbf{x}, \mathbf{t}) \, d\mathbf{w}. \qquad (1.68)$$

$$= \underbrace{\mathcal{N}(t_n \mid \vec{w}^T \vec{\phi}(\vec{x}_n), \beta^{-1})}_{\text{likelihood}} \; \underbrace{\mathcal{N}(\mathbf{w} \mid \vec{m}_0, \vec{S}_0)}_{\substack{\text{prior} \quad \text{prior} \\ \text{mean} \quad \text{cov}}}$$

https://github.com/zjost/bayesian-linear-regression/ blob/master/src/bayes-regression.ipynb

$$= Norm(\vec{w} \mid \vec{m}_N, \vec{S}_N)$$

corrections to Bishop's 1.70-1.72 are present in Bishop's chap 3 and in ctgk PRML GitHub

Note that if data comes in sequentially, the posterior of the previous step becomes the prior of the current step and we only need to calculate the updates to m_N and S_N^-1

$$\mathbf{w}_k = \left(\mathbf{\Phi}^T \mathbf{\Phi} \right)^{-1} \mathbf{\Phi}^T \mathbf{t}_k = \mathbf{\Phi}^\dagger \mathbf{t}_k \qquad (3.35)$$

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_0, \mathbf{S}_0) \qquad (3.48)$$

eral result (2.116), which allows us to write down the posterior distribution directly in the form

$$p(\mathbf{w}|\mathbf{t}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N) \qquad (3.49)$$

where

$$\mathbf{m}_N = \mathbf{S}_N \left(\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta\mathbf{\Phi}^T\mathbf{t} \right) \qquad (3.50)$$
$$\mathbf{S}_N^{-1} = \mathbf{S}_0^{-1} + \beta\mathbf{\Phi}^T\mathbf{\Phi}. \qquad (3.51)$$

Note that because the posterior distribution is Gaussian, its mode coincides with its mean. Thus the maximum posterior weight vector is simply given by $\mathbf{w}_{MAP} = \mathbf{m}_N$. If we consider an infinitely broad prior $\mathbf{S}_0 = \alpha^{-1}\mathbf{I}$ with $\alpha \to 0$, the mean $\mathbf{m}_N$ of the posterior distribution reduces to the maximum likelihood value $\mathbf{w}_{ML}$ given by (3.15). Similarly, if $N = 0$, then the posterior distribution reverts to the prior.

Furthermore, if data points arrive sequentially, then the posterior distribution at any stage acts as the prior distribution for the subsequent data point, such that the new posterior distribution is again given by (3.49).

For the remainder of this chapter, we shall consider a particular form of Gaussian prior in order to simplify the treatment. Specifically, we consider a zero-mean isotropic Gaussian governed by a single precision parameter α so that

$$p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I}) \qquad (3.52)$$

and the corresponding posterior distribution over $\mathbf{w}$ is then given by (3.49) with

$$\mathbf{m}_N = \beta\mathbf{S}_N\mathbf{\Phi}^T\mathbf{t} \qquad (3.53)$$
$$\mathbf{S}_N^{-1} = \alpha\mathbf{I} + \beta\mathbf{\Phi}^T\mathbf{\Phi}. \qquad (3.54)$$

where $\mathbf{I}$ is the unit matrix, and we have defined the vector $\phi(x)$ with elements $\phi_i(x) = x^i$ for $i = 0, \ldots, M$.

$\phi\_i(x) = x^i$. let $x=x\_0=0$. then $\phi\_i(x\_0) = 0^i$. $\phi(x\_0)=(0^0,0^1,0^2,\ldots0^9)$

x = data points drawn randomly from 0 to 1.
y = the values generated from sin(2πx)
or use those from Figure 1.17:
x = (0., 6./60, 13./60, 21./60, 27./60, 34./60, 40./60, 47./60, 54./60, 60./60)
y = (3.5/9, 7.5/9, 9./9, 8.5/9, -1./9, -1.5/9, -8./9, -4./9, -5./9, 2.5/9)
alpha = 5E-3
beta = 11.1
M = order of fit = 9 for this example
from Section 3.1.2 of PRML more details of the basis function:

---

x = (0., 6./60, 13./60, 21./60, 27./60, 34./60, 40./60, 47./60, 54./60, 60./60)
1 to N are data element numbers
0 to M are the order for this basis.
N=10, M=9.
Φ(x) is [N x M]

$\Phi(x) = \begin{vmatrix} 0.^0, & \ldots & 0.^9 \\ 6./60^0 & \ldots & 6./60^9 \\ 13./60^0 & \ldots & 13./60^9 \\ 21./60^0 & \ldots & 21./60^9 \\ 27./60^0 & \ldots & 27./60^9 \\ 34./60^0 & \ldots & 34./60^9 \\ 40./60^0 & \ldots & 40./60^9 \\ 47./60^0 & \ldots & 47./60^9 \\ 54./60^0 & \ldots & 54./60^9 \\ 60./60^0 & \ldots & 60./60^9 \end{vmatrix}$

$\phi(\mathbf{x}_n) = \phi(\mathbf{x}_2)$

$\varphi_j = \varphi_0$

Vectors are denoted by lower case bold Roman letters such as $\mathbf{x}$, and all vectors are assumed to be column vectors. A superscript T denotes the transpose of a matrix or vector, so that $\mathbf{x}^{\mathrm{T}}$ will be a row vector. Uppercase bold roman letters, such as $\mathbf{M}$, denote matrices. The notation $(w_1, \ldots, w_M)$ denotes a row vector with M elements, while the corresponding column vector is written as $\mathbf{w} = (w_1, \ldots, w_M)^{\mathrm{T}}$.

If we have N values $\mathbf{x}_1, \ldots, \mathbf{x}_N$ of a D-dimensional vector $\mathbf{x} = (x_1, \ldots, x_D)^{\mathrm{T}}$, we can combine the observations into a data matrix $\mathbf{X}$ in which the nth row of $\mathbf{X}$ corresponds to the row vector $\mathbf{x}_n^{\mathrm{T}}$. Thus the n, i element of $\mathbf{X}$ corresponds to the ith element of the nth observation $\mathbf{x}_n$. For the case of one-dimensional variables we shall denote such a matrix by $\mathbf{x}$, which is a column vector whose n^{th} element is x_n. Note that $\mathbf{x}$ (which has dimensionality N) uses a different typeface to distinguish it from $\mathbf{x}$ (which has dimensionality D).

here, x is a row vector

training data comprising N input values $\mathbf{x} = (x_1, \ldots, x_N)^{\mathrm{T}}$

original data set consists of N data points $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$.

Figure 1.17

The predictive distribution resulting from a Bayesian treatment of polynomial curve fitting using an $M = 9$ polynomial, with the fixed parameters $\alpha = 5 \times 10^{-3}$ and $\beta = 11.1$ (corresponding to the known noise variance), in which the red curve denotes the mean of the predictive distribution and the red region corresponds to ± 1 standard deviation around the mean.