3.1) Linear Basis Function Models and 3.2) The Bias-Variance Decomposition

Elif Yılmaz

Linear Basis Function Models

The simplest linear model for regression (that is; *linear regression*) is

$$y(\mathbf{x}, \mathbf{w}) = w_0 + w_1 x_1 + ... + w_D x_D$$

where $\mathbf{x} = (x_1, ..., x_D)^T$.

Linear Basis Function Models

The simplest linear model for regression (that is; *linear regression*) is

$$y(\mathbf{x}, \mathbf{w}) = w_0 + w_1 x_1 + ... + w_D x_D$$

where $\mathbf{x} = (x_1, ..., x_D)^T$.
*It is a linear function of the variables x_i and of the parameters w_i.

Linear Basis Function Models

The simplest linear model for regression (that is; *linear regression*) is

$$y(\mathbf{x}, \mathbf{w}) = w_0 + w_1 x_1 + ... + w_D x_D$$

where $\mathbf{x} = (x_1, ..., x_D)^T$.
*It is a linear function of the variables x_i and of the parameters w_i.
Because the first one imposes significant limitations on the model, we can extend it as

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

where $\phi_j(x)$'s are known as *basis functions*.
*The total number of parameters in this model is M.

By taking $\phi_0(\mathbf{x}) = 1$, we can write

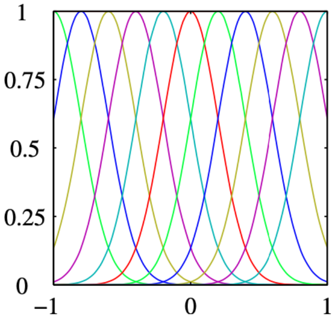
$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{M-1} w_j \phi_j(x) = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$$

where $\mathbf{w} = (w_0, ..., w_{M-1})^T$ and $\boldsymbol{\phi} = (\phi_0, ..., \phi_{M-1})^T$.

We can make many possible choices for basis functions $\phi_j(x)$. As an example,

$$\phi_j(x) = e^{-\frac{(x-\mu_j)^2}{2s^2}}$$

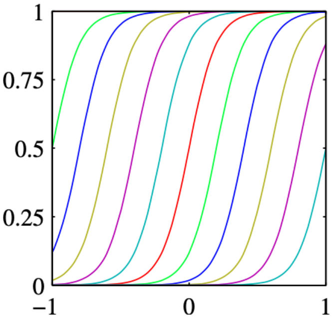
where the μ_j govern the locations of the basis functions in input space, and the parameter s governs their spatial scale.



Another choice of basis functions called sigmoidal basis function is

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

where $\sigma(a) = \frac{1}{1+e^{-a}}$ is the logistic sigmoid function.



Gaussian distribution and its likelihood function

For the next part, recall Gaussian or normal distribution.

$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} exp\Big\{ - \frac{1}{2\sigma^2}(x-\mu)^2 \Big\}$$

Gaussian distribution and its likelihood function

For the next part, recall Gaussian or normal distribution.

$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu)^2 \Big\}$$

Because the data set $\mathbf{x}$ has iid assumption, the likelihood function is $p(\mathbf{x}|\mu,\sigma^2) = \prod_{n=1}^{N} \mathcal{N}(x_n|\mu,\sigma^2)$ where $\mathbf{x} = (x_1, ..., x_N)^T$.

Gaussian distribution and its likelihood function

For the next part, recall Gaussian or normal distribution.

$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu)^2 \Big\}$$

Because the data set $\mathbf{x}$ has iid assumption, the likelihood function is $p(\mathbf{x}|\mu, \sigma^2) = \prod_{n=1}^{N} \mathcal{N}(x_n|\mu, \sigma^2)$ where $\mathbf{x} = (x_1, ..., x_N)^T$.
Therefore, the log likelihood function is:

$$\ln p(\mathbf{x}|\mu, \sigma^2) = -\frac{1}{2\sigma^2} \sum_{n=1}^{N} (x_n - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)$$

Maximum likelihood and least squares

Assume that the target variable t is given by a deterministic function y($\mathbf{x}$,$\mathbf{w}$) with additive Gaussian noise so that

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

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

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

Consider a data set of inputs $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ with corresponding target values $t_1, ..., t_N$ and $\mathbf{t} = (t_1, ..., t_N)^T$.

Consider a data set of inputs $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ with corresponding target values $t_1, ..., t_N$ and $\mathbf{t} = (t_1, ..., t_N)^T$. Then, we obtain

$$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|\mathbf{w}^T \phi(\boldsymbol{x}_n), \beta^{-1})$$

for the likelihood function, which is a function of the parameters $\mathbf{w}$ and β.

Because we are looking for modelling the distribution of input variables $\mathbf{x}$ in supervised learning problems and they always appear in the set of conditioning variables, we can write $p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta)$ as $p(\mathbf{t}|\mathbf{w}, \beta)$.

Because we are looking for modelling the distribution of input variables $\mathbf{x}$ in supervised learning problems and they always appear in the set of conditioning variables, we can write $p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta)$ as $p(\mathbf{t}|\mathbf{w}, \beta)$. Then,

$$\begin{aligned}\ln p(\mathbf{t}|\mathbf{w}, \beta) =& \sum_{n=1}^{N} \ln \mathcal{N}(t_n|\mathbf{w}^T \phi(\boldsymbol{x}_n), \beta^{-1}) \\ =& \frac{N}{2} \ln \beta - \frac{N}{2} \ln(2\pi) - \beta E_D(\mathbf{w})\end{aligned}$$

where

$$E_D(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{t_n - \mathbf{w}^T \phi(\boldsymbol{x}_n)\}^2$$

is the sum of squares error function.

Let's look at maximum likelihood to determine $\mathbf{w}$ and β and consider the first maximization with respect to $\mathbf{w}$.

Let's look at maximum likelihood to determine $\mathbf{w}$ and β and consider the first maximization with respect to $\mathbf{w}$. The gradient of the log likelihood function:

$$\nabla \ln p(\mathbf{t}|\mathbf{w}, \beta) = \sum_{n=1}^{N} \{t_n - \mathbf{w}^T \phi(\boldsymbol{x}_n)\}\phi(\boldsymbol{x}_n)^T$$

Let's look at maximum likelihood to determine $\mathbf{w}$ and β and consider the first maximization with respect to $\mathbf{w}$. The gradient of the log likelihood function:

$$\nabla \ln p(\mathbf{t}|\mathbf{w}, \beta) = \sum_{n=1}^{N} \{t_n - \mathbf{w}^T \phi(\boldsymbol{x}_n)\} \phi(\boldsymbol{x}_n)^T$$

By setting the gradient to zero, it gives us

$$\sum_{n=1}^{N} t_n \phi(\boldsymbol{x}_n)^T = \mathbf{w}^T \Big(\sum_{n=1}^{N} \phi(\boldsymbol{x}_n) \phi(\boldsymbol{x}_n)^T \Big)$$

If we solve this equation for **w**, we will obtain

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

where Φ is NxM matrix, called *design matrix*, whose elements are given by $\Phi_{nj} = \phi_j(\mathbf{x}_n)$ such that

$$\Phi = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & ... & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & ... & \phi_{M-1}(\mathbf{x}_2) \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & ... & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

If we solve this equation for **w**, we will obtain

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

where Φ is NxM matrix, called *design matrix*, whose elements are given by $\Phi_{nj} = \phi_j(\mathbf{x}_n)$ such that

$$\Phi = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \dots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \dots & \phi_{M-1}(\mathbf{x}_2) \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \dots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

* The quantity $\Phi^\dagger = (\Phi^T\Phi)^{-1}\Phi^T$ is known as the *Moore-Penrose pseudo-inverse* of the matrix Φ.

If we solve this equation for $\mathbf{w}$, we will obtain

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

where Φ is NxM matrix, called *design matrix*, whose elements are given by $\Phi_{nj} = \phi_j(\mathbf{x}_n)$ such that

$$\Phi = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \dots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \dots & \phi_{M-1}(\mathbf{x}_2) \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \dots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

* The quantity $\Phi^\dagger = (\Phi^T \Phi)^{-1} \Phi^T$ is known as the *Moore-Penrose pseudo-inverse* of the matrix Φ.
*In our case, we see that $\Phi^\dagger = \Phi^{-1}$.

To gain some insight for bias parameter w_0, we can write the error function as

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

To gain some insight for bias parameter w_0, we can write the error function as

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

By setting the derivative wrt w_0 to zero and solving it for w_0, we get

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

where $\bar{t} = \frac{1}{N}\sum_{n=1}^{N} t_n$ and $\bar{\phi}_j = \frac{1}{N}\sum_{n=1}^{N}\phi_j(\mathbf{x}_n)$.

When we maximize the log likelihood function wrt the noise precision parameter β, we obtain

$$\frac{1}{\beta_{ML}} = \frac{1}{N}\sum_{n=1}^{N}\{t_n - \mathbf{w}_{ML}^T\phi(\boldsymbol{x}_n)\}^2$$

Regularized least squares

To control overfitting, we add a regularization term to error function, so that the total error function becomes

$$E_D(\mathbf{w}) + \lambda E_W(\mathbf{w})$$

where λ is the regularization coefficient and the regularization term is $E_W(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w}$.

Regularized least squares

To control overfitting, we add a regularization term to error function, so that the total error function becomes

$$E_D(\mathbf{w}) + \lambda E_W(\mathbf{w})$$

where λ is the regularization coefficient and the regularization term is $E_W(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w}$. Therefore, the total error function:

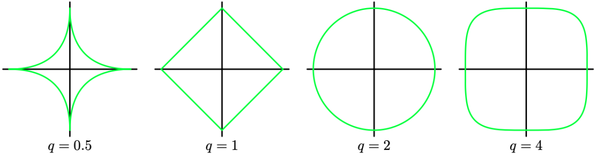
$$E_D(\mathbf{w}) + \lambda E_W(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^T\phi(\boldsymbol{x}_n)\}^2 + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}$$

By setting gradient wrt $\mathbf{w}$ of the total error function to zero and solving it for $\mathbf{w}$, we get

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

With more general regularizer, the total error function takes the form

$$\frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^T\phi(\boldsymbol{x}_n)\}^2 + \frac{\lambda}{2}\sum_{j=1}^{M}|w_j|^q$$



Contours of the regularization function for different values of q

*The case of q = 1 is know as the *lasso regression* and The case of q = 2 is know as the *ridge regression*.

The Bias-Variance Decomposition

For this part, let's go back to section 1.5.5 and remember loss functions for regression.

The Bias-Variance Decomposition

For this part, let's go back to section 1.5.5 and remember loss functions for regression.
Choose a specific estimate $y(\mathbf{x})$ of the value t for each input $\mathbf{x}$.

For this part, let's go back to section 1.5.5 and remember loss functions for regression.
Choose a specific estimate y(**x**) of the value t for each input **x**.
Suppose that a loss L(t,y(**x**)) consists of. The average or expected loss is

$$\mathbb{E}[L] = \int\int L(t, y(\mathbf{x}))p(\mathbf{x}, t)d\mathbf{x}dt$$

A common choice of loss function is given by $L(t, y(\mathbf{x})) = \{y(\mathbf{x}) - t\}^2$.

A common choice of loss function is given by $L(t, y(\mathbf{x})) = \{y(\mathbf{x}) - t\}^2$. Then, the expected loss can be written as

$$\mathbb{E}[L] = \int \int \{y(\mathbf{x}) - t\}^2 p(\mathbf{x}, t) d\mathbf{x} dt$$

A common choice of loss function is given by $L(t, y(\mathbf{x})) = \{y(\mathbf{x}) - t\}^2$. Then, the expected loss can be written as

$$\mathbb{E}[L] = \int \int \{y(\mathbf{x}) - t\}^2 p(\mathbf{x}, t) d\mathbf{x} dt$$

Goal: to choose y($\mathbf{x}$) which minimizes $\mathbb{E}[L]$.
Set its derivative wrt y($\mathbf{x}$) to zero as

$$\frac{\partial \mathbb{E}[L]}{\partial y(\mathbf{x})} = 2 \int \{y(\mathbf{x}) - t\} p(\mathbf{x}, t) dt = 0.$$

When we solve it for y($\mathbf{x}$), we get

$$y(\mathbf{x}) = \frac{\int tp(\mathbf{x}, t)dt}{p(\mathbf{x})} = \int tp(t|\mathbf{x})dt = \mathbb{E}_t[t|\mathbf{x}]$$

When we solve it for y($\mathbf{x}$), we get

$$y(\mathbf{x}) = \frac{\int tp(\mathbf{x}, t)dt}{p(\mathbf{x})} = \int tp(t|\mathbf{x})dt = \mathbb{E}_t[t|\mathbf{x}]$$

Expand $\{y(\mathbf{x}) - t\}^2$ as

$$\begin{aligned}\{y(\mathbf{x}) - \mathbb{E}_t[t|\mathbf{x}] + \mathbb{E}_t[t|\mathbf{x}] - t\}^2 =& \{y(\mathbf{x}) - \mathbb{E}_t[t|\mathbf{x}]\}^2 \\ &+ 2\{y(\mathbf{x}) - \mathbb{E}_t[t|\mathbf{x}]\}\{\mathbb{E}_t[t|\mathbf{x}] - t\} \\ &+ \{\mathbb{E}_t[t|\mathbf{x}] - t\}^2.\end{aligned}$$

Therefore, we obtain loss function in the form

$$\mathbb{E}[L] = \int \{y(\mathbf{x}) - \mathbb{E}_t[t|\mathbf{x}]\}^2 p(\mathbf{x}) d\mathbf{x} + \int \{\mathbb{E}_t[t|\mathbf{x}] - t\}^2 p(\mathbf{x}) d\mathbf{x}$$

by substituting the loss function and performing the integral over t.

Therefore, we obtain loss function in the form

$$\mathbb{E}[L] = \int \{y(\mathbf{x}) - \mathbb{E}_t[t|\mathbf{x}]\}^2 p(\mathbf{x}) d\mathbf{x} + \int \{\mathbb{E}_t[t|\mathbf{x}] - t\}^2 p(\mathbf{x}) d\mathbf{x}$$

by substituting the loss function and performing the integral over t. If we denote the conditional expectation $\mathbb{E}_t[t|\mathbf{x}]$ as h($\mathbf{x}$), the expected loss can be written in the form:

$$\mathbb{E}[L] = \int \{y(\mathbf{x}) - h(\mathbf{x})\}^2 p(\mathbf{x}) d\mathbf{x} + \int \{h(\mathbf{x}) - t\}^2 p(\mathbf{x}, t) d\mathbf{x} dt$$

For any given data set $\mathcal{D}$, we can obtain a prediction function $y(\mathbf{x}; \mathcal{D})$. When we apply similar cases mentioned earlier for $\{y(\mathbf{x}; \mathcal{D}) - h(\mathbf{x})\}^2$, it gives

$$\begin{aligned}
\mathbb{E}_{\mathcal{D}}[\{y(\mathbf{x}; \mathcal{D}) - h(\mathbf{x})\}^2] = & \{\mathbb{E}_{\mathcal{D}}[y(\mathbf{x}; \mathcal{D})] - h(\mathbf{x})\}^2 \quad \Big\} \; (bias)^2 \\
& + \mathbb{E}_{\mathcal{D}}[\{y(\mathbf{x}; \mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x}; \mathcal{D})]\}^2] \quad \Big\} \; variance
\end{aligned}$$

For any given data set $\mathcal{D}$, we can obtain a prediction function $y(\mathbf{x};\mathcal{D})$. When we apply similar cases mentioned earlier for $\{y(\mathbf{x};\mathcal{D}) - h(\mathbf{x})\}^2$, it gives

$$\mathbb{E}_{\mathcal{D}}[\{y(\mathbf{x};\mathcal{D}) - h(\mathbf{x})\}^2] = \left.\{\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\}^2 \right\} \textit{(bias)}^2$$
$$\left. + \mathbb{E}_{\mathcal{D}}[\{y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\}^2] \right\} \textit{variance}$$

Therefore, in short we can say

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

where $\textit{noise} = \int \{h(\mathbf{x}) - t\}^2 p(\mathbf{x}, t) d\mathbf{x} dt$.
Goal: to minimize the expected loss.

Example

Generate 100 data sets, each containing N $= 25$ data points, independently from the sinusoidal curve $h(x) = sin(2\pi x)$. The data sets are indexed by $l = 1, ..., L$, where L $= 100$, and for each data set $D^{(l)}$, fit a model with 24 Gaussian basis functions by minimizing the regularized error function

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

to give a prediction function $y^{(l)}(x)$

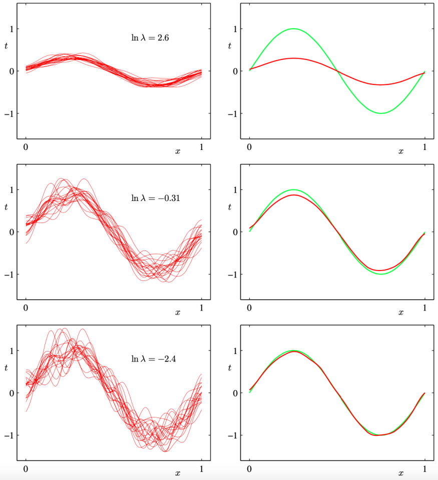
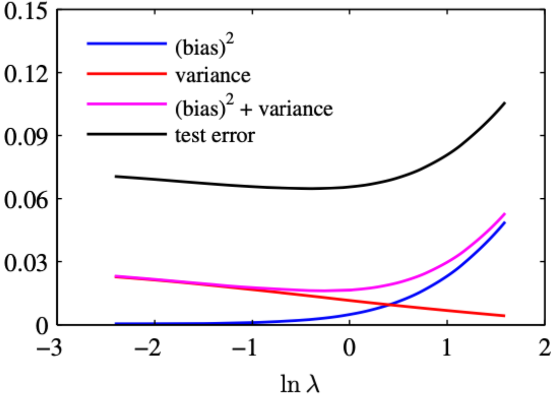


Figure: The **left column** shows the result of fitting the model to the data sets for various values of $\ln\lambda$. The **right column** shows the corresponding average of the 100 fits (red) along with the sinusoidal function from which the data sets were generated (green).



The minimum value of $(\mathit{bias})^2 + \mathit{variance}$ occurs around $\ln \lambda = -0.31$, which is close to the value that gives the minimum error on the test data.