Pattern Recognition and Machine Learning

Chapter 6: Kernel Methods

Polynomial regression

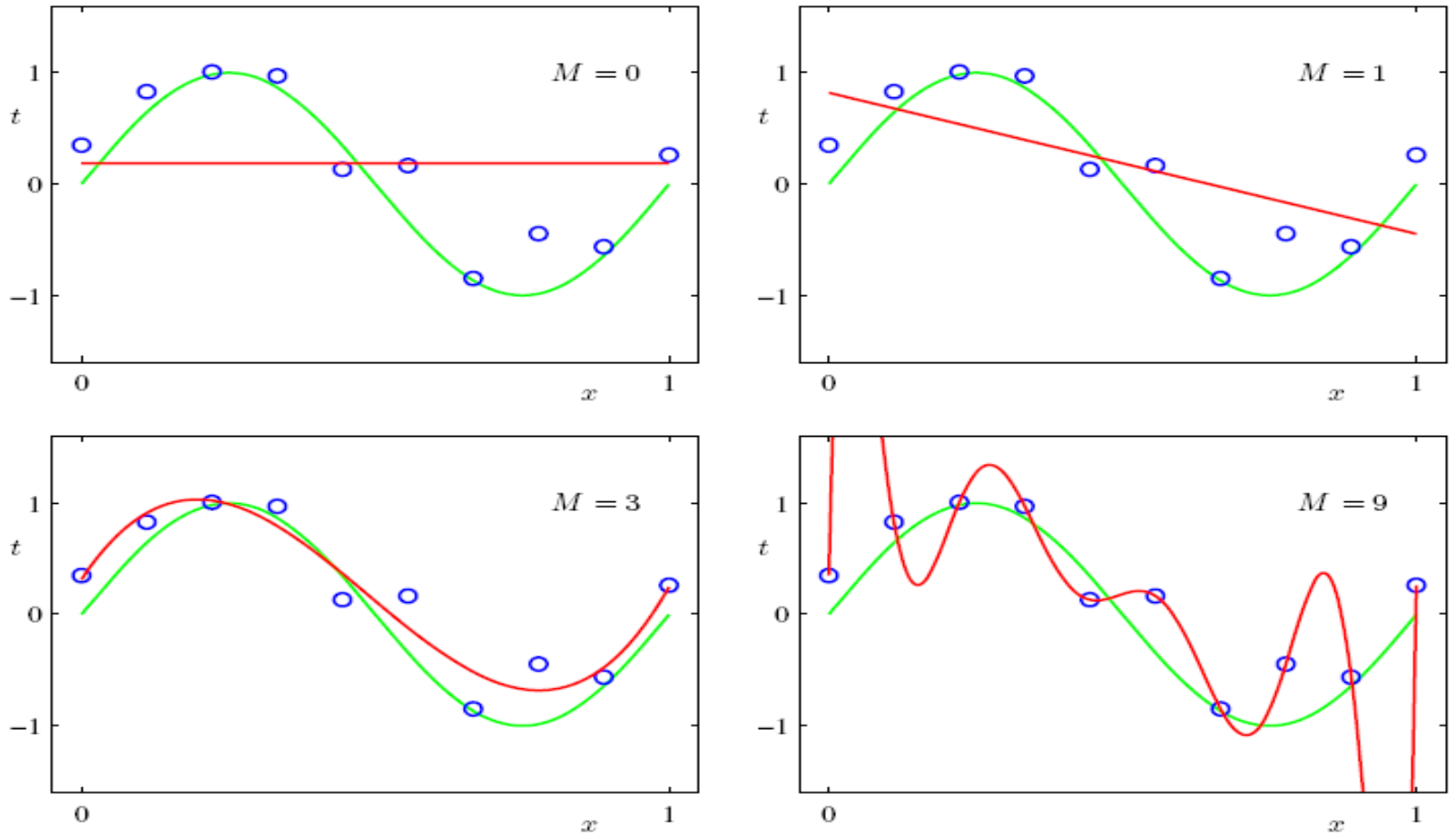
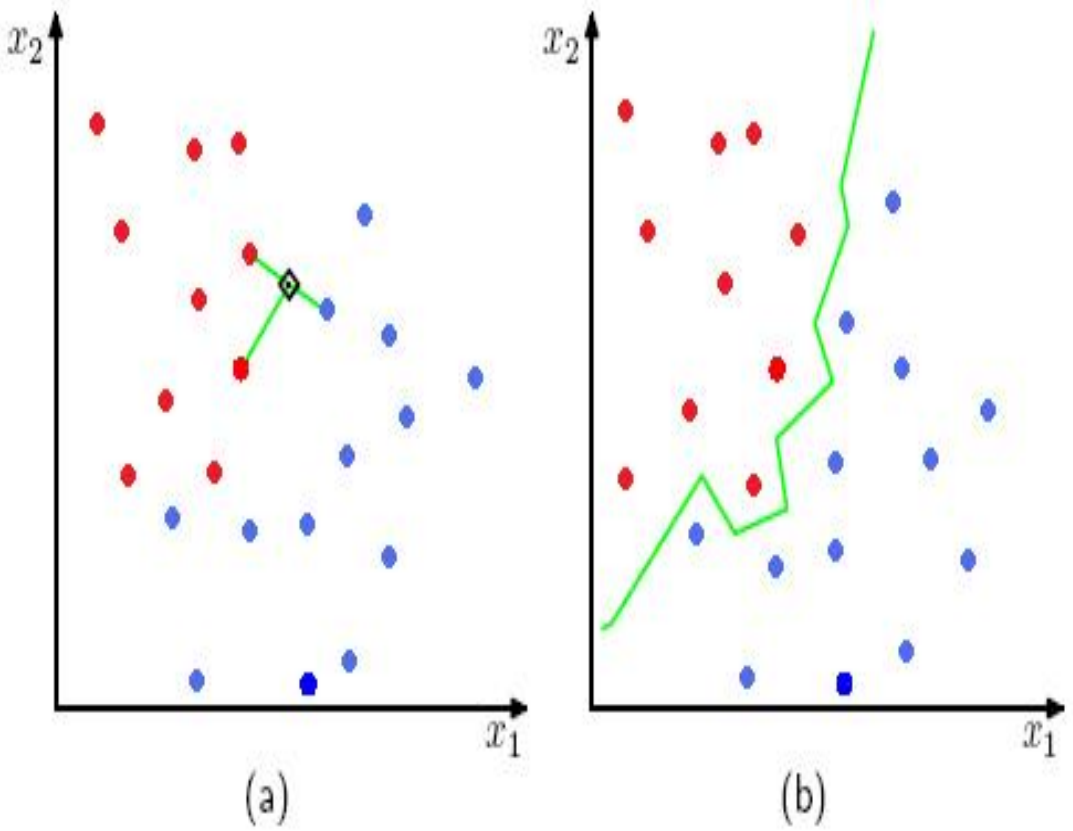


Figure 1.4 Plots of polynomials having various orders M, shown as red curves, fitted to the data set shown in Figure 1.2.

K-Nearest Neighbour

Figure 2.27 (a) In the K-nearest-neighbour classifier, a new point, shown by the black diamond, is classified according to the majority class membership of the K closest training data points, in this case $K = 3$. (b) In the nearest-neighbour ($K = 1$) approach to classification, the resulting decision boundary is composed of hyperplanes that form perpendicular bisectors of pairs of points from different classes.



Kernel Methods

- Are used to find and study general types of relations (eg. Clustering, correlation, classification) in general types of data (vectors, images, sets).

- For nonlinear feature space mapping $\phi(\mathbf{x})$, the kernel function is given as

$$k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^{\mathrm{T}} \phi(\mathbf{x}').$$

Kernel Methods

- Kernel Trick (Kernel Substitution) :- If the input vector enters in the form of scalar product, then the scalar product can be replaced by some other kernel functions .

- Linear Kernels: $k(\mathrm{x}, \mathrm{x}') = \mathrm{x}^{\mathrm{T}}\mathrm{x}'$

- Stationary Kernels: $k(\mathrm{x}, \mathrm{x}') = k(\mathrm{x} - \mathrm{x}')$

- Homogeneous Kernels : $k(\mathrm{x}, \mathrm{x}') = k(\|\mathrm{x} - \mathrm{x}'\|)$.

Dual Representations

- Consider linear regression model whose regularized sum-of-squares error function is given by

$$J(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left\{\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - t_n\right\}^2 + \frac{\lambda}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} \qquad \text{where } \lambda \geqslant 0.$$

- Minimizing with respect to *w,*

$$\mathbf{w} = -\frac{1}{\lambda}\sum_{n=1}^{N}\left\{\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - t_n\right\}\boldsymbol{\phi}(\mathbf{x}_n) = \sum_{n=1}^{N} a_n\boldsymbol{\phi}(\mathbf{x}_n) = \boldsymbol{\Phi}^{\mathrm{T}}\mathbf{a}$$

where Φ is the design matrix , whose nth row is given by $\boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}}$.

$$a_n = -\frac{1}{\lambda}\left\{\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - t_n\right\}.$$

$$\mathbf{a} = (a_1, \ldots, a_N)^{\mathrm{T}}$$

Dual Representations

- Substituting $\mathrm{w} = \Phi^{\mathrm{T}}\mathrm{a}$ into $J(\mathbf{w})$

$$J(\mathrm{a}) = \frac{1}{2}\mathrm{a}^{\mathrm{T}}\Phi\Phi^{\mathrm{T}}\Phi\Phi^{\mathrm{T}}\mathrm{a} - \mathrm{a}^{\mathrm{T}}\Phi\Phi^{\mathrm{T}}\mathbf{t} + \frac{1}{2}\mathbf{t}^{\mathrm{T}}\mathbf{t} + \frac{\lambda}{2}\mathrm{a}^{\mathrm{T}}\Phi\Phi^{\mathrm{T}}\mathrm{a}$$

where $\mathbf{t} = (t_1, \ldots, t_N)^{\mathrm{T}}$.

- Define *Gram Matrix* $\mathrm{K} = \Phi\Phi^{\mathrm{T}}$, which is an N x N symmetric matrix with elements

$$K_{nm} = \phi(\mathrm{x}_n)^{\mathrm{T}}\phi(\mathrm{x}_m) = k(\mathrm{x}_n, \mathrm{x}_m)$$

- Re-writing the sum-of-squares error using the *Gram Matrix*

$$J(\mathrm{a}) = \frac{1}{2}\mathrm{a}^{\mathrm{T}}\mathrm{K}\mathrm{K}\mathrm{a} - \mathrm{a}^{\mathrm{T}}\mathrm{K}\mathbf{t} + \frac{1}{2}\mathbf{t}^{\mathrm{T}}\mathbf{t} + \frac{\lambda}{2}\mathrm{a}^{\mathrm{T}}\mathrm{K}\mathrm{a}.$$

Dual Representations

- Minimizing $J(\mathbf{a})$ with respect to **a** , and solving

$$\mathrm{a} = (\mathbf{K} + \lambda \mathbf{I}_N)^{-1} \mathbf{t}.$$

- Substituting back to the original linear regression model

$$y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}} \phi(\mathbf{x}) = \mathbf{a}^{\mathrm{T}} \mathbf{\Phi} \phi(\mathbf{x}) = \mathrm{k}(\mathbf{x})^{\mathrm{T}} (\mathbf{K} + \lambda \mathbf{I}_N)^{-1} \mathbf{t}$$

- Note that vector **a** is computed by inverting N x N matrix.

- *Compare:* $\mathbf{w}_{\mathrm{ML}} = \left(\mathbf{\Phi}^{\mathrm{T}} \mathbf{\Phi}\right)^{-1} \mathbf{\Phi}^{\mathrm{T}} \mathbf{t}$

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

Dual Representations

- Even though the dual formulation requires higher dimensions , it comes with some advantages.
  - We can see that $y(\mathbf{x})$ is expressed entirely in terms of the kernel functions.
  - This allows us to work with feature spaces of higher dimension, even with infinite dimensionality.

Constructing Kernels

- *First Approach:* Select the feature space mapping $\phi(\mathrm{x})$ and use it to construct the corresponding kernel.

$$k(x, x') = \phi(x)^{\mathrm{T}}\phi(x') = \sum_{i=1}^{M} \phi_i(x)\phi_i(x')$$

- Examples of basis function:
  - Polynomial: $\phi_j(x) = x^j$
  - Gaussian: $\phi_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\}$
  - Sigmoid: $\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right)$

Constructing Kernels

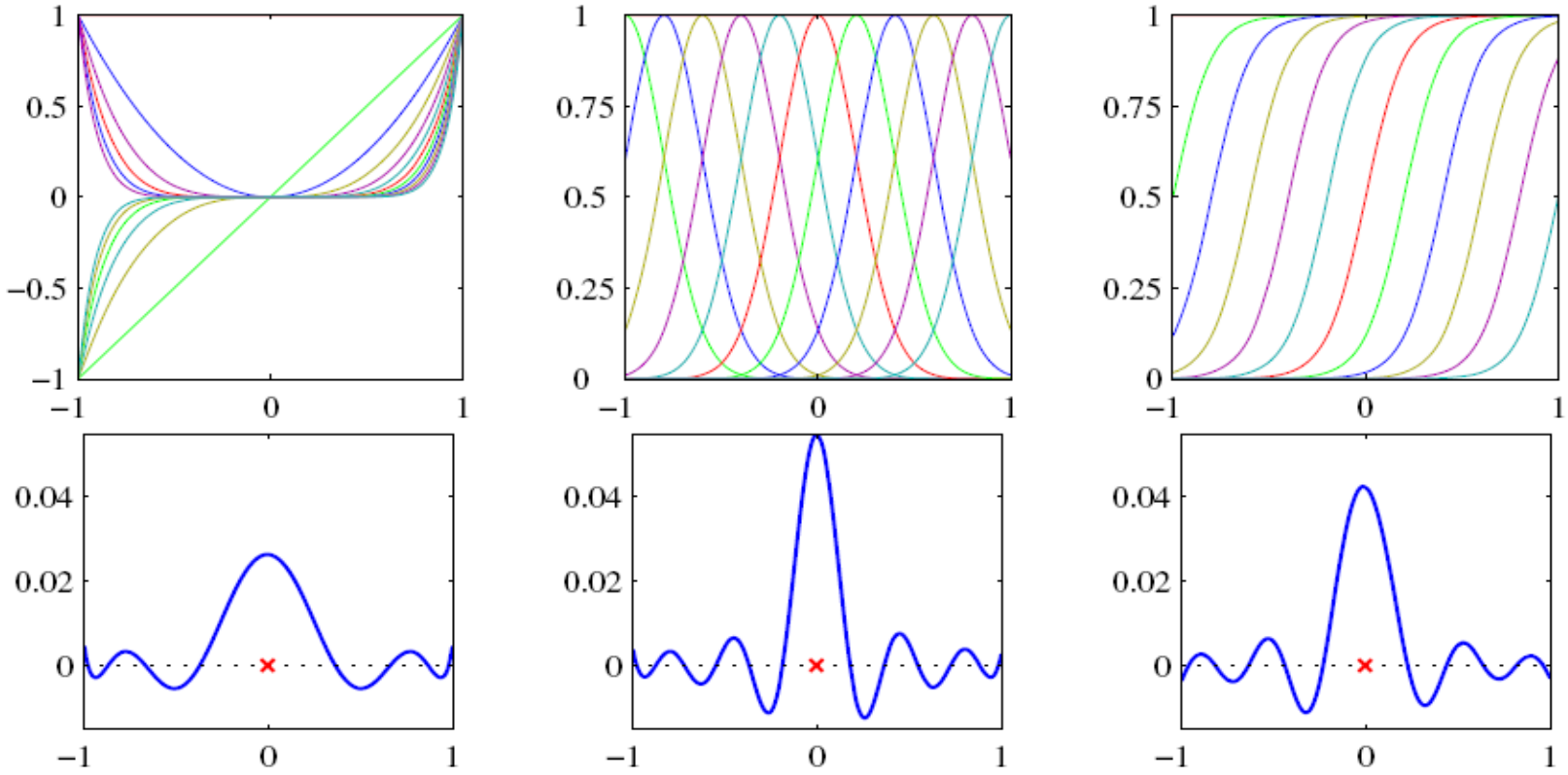


Figure 6.1 Illustration of the construction of kernel functions starting from a corresponding set of basis functions. In each column the lower plot shows the kernel function $k(x, x')$ defined by (6.10) plotted as a function of x for $x' = 0$, while the upper plot shows the corresponding basis functions given by polynomials (left column), 'Gaussians' (centre column), and logistic sigmoids (right column).

Constructing Kernels

- *Second Approach:* Construct kernel functions directly. We need to make sure that we are selecting a valid kernel.

- Valid Kernels: kernels who has a Gram Matrix whose components are positive semi-definite for all possible choices of input data.

Constructing Kernels

- Consider a simple example: $k(\mathrm{x}, \mathrm{z}) = \left(\mathrm{x}^{\mathrm{T}}\mathrm{z}\right)^2$.

- Considering a 2D input space

$$\begin{aligned} k(\mathbf{x}, \mathbf{z}) &= \left(\mathbf{x}^{\mathrm{T}}\mathbf{z}\right)^2 = (x_1 z_1 + x_2 z_2)^2 \\ &= x_1^2 z_1^2 + 2 x_1 z_1 x_2 z_2 + x_2^2 z_2^2 \\ &= (x_1^2, \sqrt{2} x_1 x_2, x_2^2)(z_1^2, \sqrt{2} z_1 z_2, z_2^2)^{\mathrm{T}} \\ &= \boldsymbol{\phi}(\mathbf{x})^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{z}). \end{aligned}$$

- New complex Kernels can also be reconstructed by using simpler kernels as building blocks.

Techniques for Constructing New Kernels.

Given valid kernels $k_1(\mathbf{x}, \mathbf{x}')$ and $k_2(\mathbf{x}, \mathbf{x}')$, the following new kernels will also be valid:

$$
\begin{aligned}
k(\mathbf{x}, \mathbf{x}') &= c k_1(\mathbf{x}, \mathbf{x}') && (6.13)\\
k(\mathbf{x}, \mathbf{x}') &= f(\mathbf{x}) k_1(\mathbf{x}, \mathbf{x}') f(\mathbf{x}') && (6.14)\\
k(\mathbf{x}, \mathbf{x}') &= q\left(k_1(\mathbf{x}, \mathbf{x}')\right) && (6.15)\\
k(\mathbf{x}, \mathbf{x}') &= \exp\left(k_1(\mathbf{x}, \mathbf{x}')\right) && (6.16)\\
k(\mathbf{x}, \mathbf{x}') &= k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}') && (6.17)\\
k(\mathbf{x}, \mathbf{x}') &= k_1(\mathbf{x}, \mathbf{x}') k_2(\mathbf{x}, \mathbf{x}') && (6.18)\\
k(\mathbf{x}, \mathbf{x}') &= k_3\left(\boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{x}')\right) && (6.19)\\
k(\mathbf{x}, \mathbf{x}') &= \mathbf{x}^{\mathrm{T}} \mathbf{A} \mathbf{x}' && (6.20)\\
k(\mathbf{x}, \mathbf{x}') &= k_a(\mathbf{x}_a, \mathbf{x}'_a) + k_b(\mathbf{x}_b, \mathbf{x}'_b) && (6.21)\\
k(\mathbf{x}, \mathbf{x}') &= k_a(\mathbf{x}_a, \mathbf{x}'_a) k_b(\mathbf{x}_b, \mathbf{x}'_b) && (6.22)
\end{aligned}
$$

where $c > 0$ is a constant, $f(\cdot)$ is any function, $q(\cdot)$ is a polynomial with nonnegative coefficients, $\boldsymbol{\phi}(\mathbf{x})$ is a function from $\mathbf{x}$ to $\mathbb{R}^M$, $k_3(\cdot, \cdot)$ is a valid kernel in $\mathbb{R}^M$, $\mathbf{A}$ is a symmetric positive semidefinite matrix, $\mathbf{x}_a$ and $\mathbf{x}_b$ are variables (not necessarily disjoint) with $\mathbf{x} = (\mathbf{x}_a, \mathbf{x}_b)$, and k_a and k_b are valid kernel functions over their respective spaces.

Constructing Kernels

- Gaussian Kernels

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2\sigma^2\right)$$

Taking the inner part:

$$\|\mathbf{x} - \mathbf{x}'\|^2 = \mathbf{x}^{\mathrm{T}}\mathbf{x} + (\mathbf{x}')^{\mathrm{T}}\mathbf{x}' - 2\mathbf{x}^{\mathrm{T}}\mathbf{x}'$$

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\mathbf{x}^{\mathrm{T}}\mathbf{x}/2\sigma^2\right) \exp\left(\mathbf{x}^{\mathrm{T}}\mathbf{x}'/\sigma^2\right) \exp\left(-(\mathbf{x}')^{\mathrm{T}}\mathbf{x}'/2\sigma^2\right)$$

Substituting kernel

$$k(\mathbf{x}, \mathbf{x}') = \exp\left\{-\frac{1}{2\sigma^2}\left(\kappa(\mathbf{x}, \mathbf{x}) + \kappa(\mathbf{x}', \mathbf{x}') - 2\kappa(\mathbf{x}, \mathbf{x}')\right)\right\}.$$

Linear Regression Revisited

- Consider

$$y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x})$$

- Taking prior distribution over **w**

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

- Note that this induces a probability distribution over function y(x).

- In terms of vector representation

$$\mathbf{y} = \boldsymbol{\Phi}\mathbf{w}$$

Linear Regression Revisited

- Note that since **y** is a linear combination of Gaussian distributed values **w,** and hence it is also a Gaussian distributed.

$$\mathbb{E}[\mathbf{y}] = \mathbf{\Phi}\mathbb{E}[\mathbf{w}] = \mathbf{0}$$

$$\mathrm{cov}[\mathbf{y}] = \mathbb{E}\left[\mathbf{y}\mathbf{y}^{\mathrm{T}}\right] = \mathbf{\Phi}\mathbb{E}\left[\mathbf{w}\mathbf{w}^{\mathrm{T}}\right]\mathbf{\Phi}^{\mathrm{T}} = \frac{1}{\alpha}\mathbf{\Phi}\mathbf{\Phi}^{\mathrm{T}} = \mathbf{K}$$

- where **K** is the Gram matrix with elements

$$K_{nm} = k(\mathbf{x}_n, \mathbf{x}_m) = \frac{1}{\alpha}\phi(\mathbf{x}_n)^{\mathrm{T}}\phi(\mathbf{x}_m)$$

- Definition: A Gaussian process is defined as a probability distribution over functions $y(\mathbf{x})$ evaluated at an arbitrary set of points $\mathbf{x}_1, \ldots, \mathbf{x}_N$ jointly have a Gaussian distribution.

Linear Regression Revisited

- Note that the joint distribution of Gaussian process can be completely specified by the mean and covariance.

- Note also that the covariance can be evaluated from the kernel function.

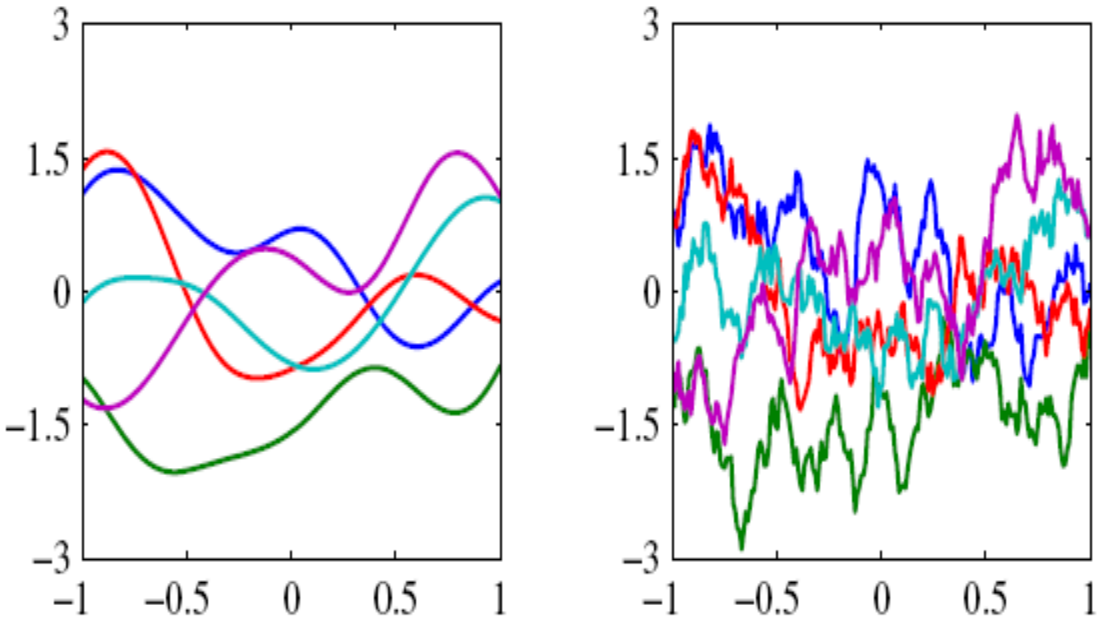
$$\mathbb{E}\left[y(\mathbf{x}_n)y(\mathbf{x}_m)\right] = k(\mathbf{x}_n, \mathbf{x}_m).$$

- Taking the Gaussian and exponential kernel functions, for example, samples are drawn.

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\|\mathbf{x} - \mathbf{x}'\|^2/2\sigma^2\right) \qquad k(x, x') = \exp\left(-\theta\left|x - x'\right|\right)$$

Linear Regression Revisited

Figure 6.4 Samples from Gaussian processes for a 'Gaussian' kernel (left) and an exponential kernel (right).



Gaussian processes for regression

- Considering noise on the observed target

$$t_n = y_n + \epsilon_n$$

- Considering Gaussian noise

$$p(t_n|y_n) = \mathcal{N}(t_n|y_n, \beta^{-1})$$

- For independent noise, the joint distribution is given by

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

- Using the Gaussian process

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\mathbf{0}, \mathbf{K}).$$

- Marginalizing the probability

$$p(\mathbf{t}) = \int p(\mathbf{t}|\mathbf{y})p(\mathbf{y})\,\mathrm{d}\mathbf{y} = \mathcal{N}(\mathbf{t}|\mathbf{0}, \mathbf{C})$$

Gaussian processes for regression

- Where **C** is the covariance matrix given as

$$C(\mathbf{x}_n, \mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m) + \beta^{-1}\delta_{nm}.$$

- Note the summation of the covariance.

- Widely used Kernel function

$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left\{-\frac{\theta_1}{2}\|\mathbf{x}_n - \mathbf{x}_m\|^2\right\} + \theta_2 + \theta_3 \mathbf{x}_n^{\mathrm{T}} \mathbf{x}_m.$$

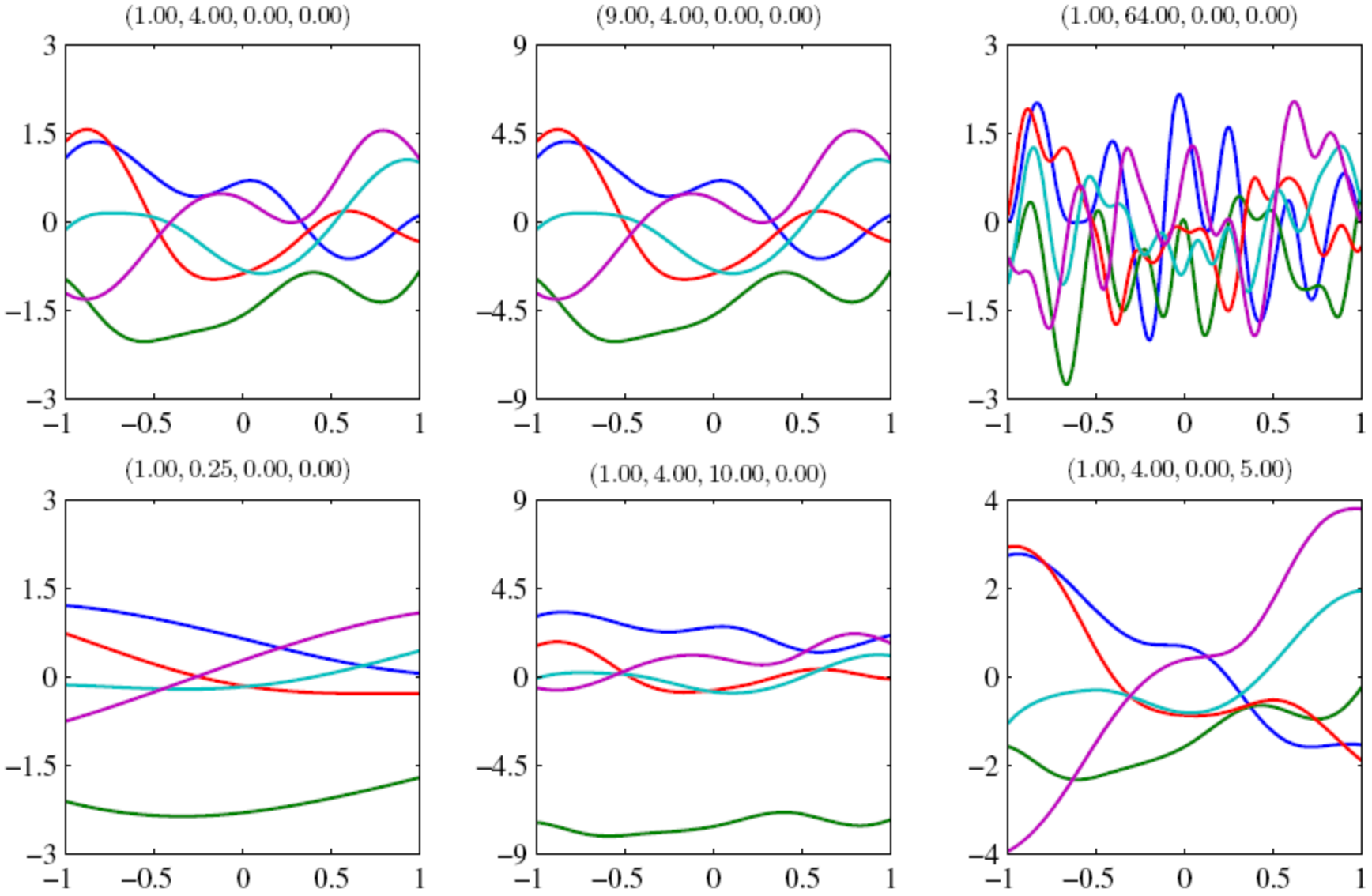
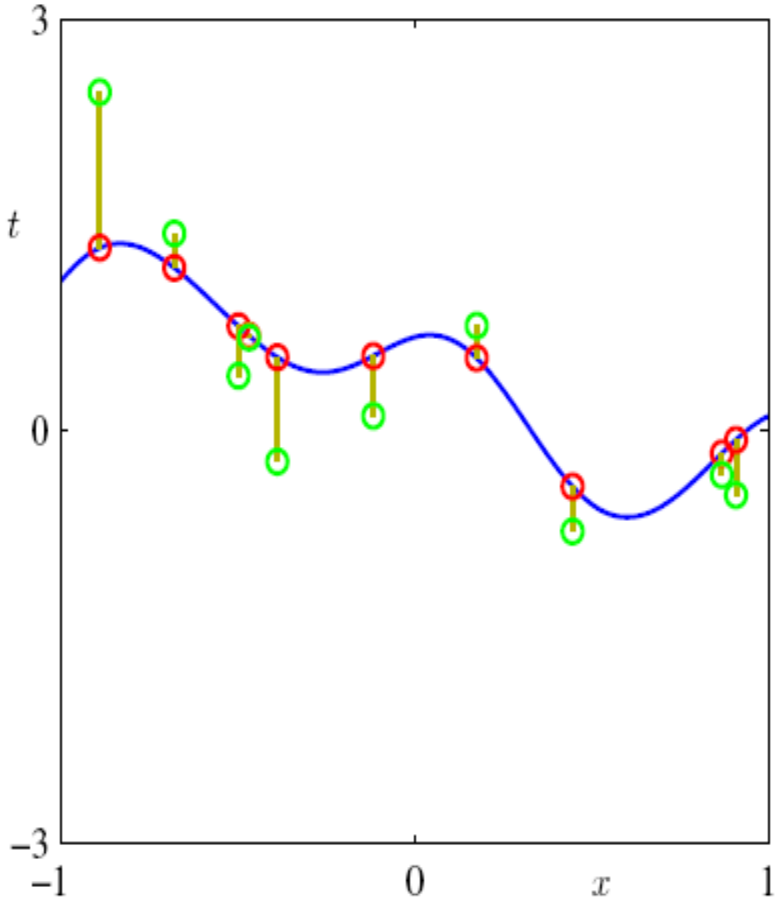


Figure 6.5 Samples from a Gaussian process prior defined by the covariance function (6.63). The title above each plot denotes $(\theta_0, \theta_1, \theta_2, \theta_3)$.

Gaussian processes for regression

Figure 6.6 Illustration of the sampling of data points $\{t_n\}$ from a Gaussian process. The blue curve shows a sample function from the Gaussian process prior over functions, and the red points show the values of y_n obtained by evaluating the function at a set of input values $\{x_n\}$. The corresponding values of $\{t_n\}$, shown in green, are obtained by adding independent Gaussian noise to each of the $\{y_n\}$.



Gaussian processes for regression

- Goal of regression: predict $p(t_{N+1}|\mathbf{t}_N)$
- Using the joint distribution

$$p(\mathbf{t}_{N+1}) = \mathcal{N}(\mathbf{t}_{N+1}|\mathbf{0}, \mathbf{C}_{N+1})$$

$p(t_{N+1}|\mathbf{t})$

- Partitioning the covariance

$$\mathbf{C}_{N+1} = \begin{pmatrix} \mathbf{C}_N & \mathbf{k} \\ \mathbf{k}^{\mathrm{T}} & c \end{pmatrix}$$

- **K** has elements $k(\mathbf{x}_n, \mathbf{x}_{N+1})$ and c is given as

$$c = k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) + \beta^{-1}$$

- $p(t_{N+1}|\mathbf{t})$ is Gaussian with mean and covariance given by

$$\begin{aligned} m(\mathbf{x}_{N+1}) &= \mathbf{k}^{\mathrm{T}}\mathbf{C}_N^{-1}\mathbf{t} \\ \sigma^2(\mathbf{x}_{N+1}) &= c - \mathbf{k}^{\mathrm{T}}\mathbf{C}_N^{-1}\mathbf{k}. \end{aligned}$$

Gaussian processes for regression

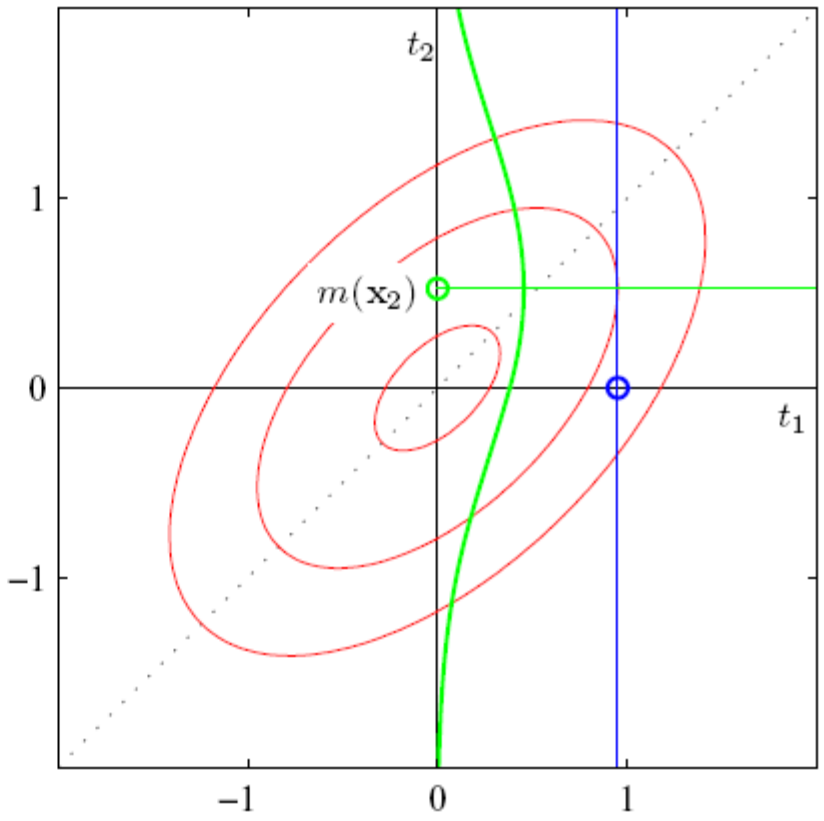
- Note that the mean and covariance are dependent on the term **k** which is dependent on the input $\mathbf{x}_{N+1}$

- Note also that the additional kernel matrix should be a valid kernel.

- The Gaussian process viewpoint is advantageous in that we can consider covariance functions that can be expressed in terms of an infinite number of basis functions.

Gaussian processes for regression

Figure 6.7 Illustration of the mechanism of Gaussian process regression for the case of one training point and one test point, in which the red ellipses show contours of the joint distribution $p(t_1, t_2)$. Here t_1 is the training data point, and conditioning on the value of t_1, corresponding to the vertical blue line, we obtain $p(t_2|t_1)$ shown as a function of t_2 by the green curve.



Gaussian processes for regression

Figure 6.8 Illustration of Gaussian process regression applied to the sinusoidal data set in Figure A.6 in which the three right-most data points have been omitted. The green curve shows the sinusoidal function from which the data points, shown in blue, are obtained by sampling and addition of Gaussian noise. The red line shows the mean of the Gaussian process predictive distribution, and the shaded region corresponds to plus and minus two standard deviations. Notice how the uncertainty increases in the region to the right of the data points.