What is the regression? $\underline{w}^T \underline{\phi}(\underline{x}) \rightarrow$ our model $y(\underline{x})$

$y(\underline{x}) = \underline{w}^T \underline{\phi}(\underline{x})$. For the training data, we are given target values t_i.

scalar

input

$y(\underline{x}_i) = \underline{w}^T \underline{\phi}(\underline{x}_i)$ is our modelled output for which Mother Nature (physical process) has given a value 't_i'

i.e., for a 'good' model, $y(\underline{x}_i) = \underline{w}^T \underline{\phi}(\underline{x}_i)$ should be close to t_i.

$$y(\underline{x}) = \underline{w}^T \underline{\phi}(\underline{x}) = (\underline{\Phi}\, \underline{a})^T \underline{\phi}(\underline{x}) = \underline{a}^T \boxed{\underline{\Phi}\, \underline{\phi}(\underline{x})}$$

$1 \times N$ $\quad N \times M$ $\quad M \times 1$

$N \times 1$

scalar 1×1

consider

$$\underset{N\times M}{\underline{\Phi}}\ \underset{M\times 1}{\underline{\phi}(\underline{x})} = \begin{bmatrix} \underline{\phi}^T(\underline{x}_1) \\ \underline{\phi}^T(\underline{x}_2) \\ \vdots \\ \underline{\phi}^T(\underline{x}_N) \end{bmatrix} \underline{\phi}(\underline{x}) = \begin{bmatrix} \underline{\phi}^T(\underline{x}_1)\, \underline{\phi}(\underline{x}) \\ \underline{\phi}^T(\underline{x}_2)\, \underline{\phi}(\underline{x}) \\ \vdots \\ \underline{\phi}^T(\underline{x}_N)\, \underline{\phi}(\underline{x}) \end{bmatrix}$$

$N \times 1$

$$= \begin{bmatrix} k(\underline{x}_1, \underline{x}) \\ k(\underline{x}_2, \underline{x}) \\ \vdots \\ k(\underline{x}_N, \underline{x}) \end{bmatrix} \rightsquigarrow \underline{k}(\underline{x})$$

$$y(\underline{x}) = \underline{a}^T \underline{k}(\underline{x}) = \underline{k}^T(\underline{x})\, \underline{a}$$

$$\Rightarrow y(\underline{x}) = \underline{k}^T(\underline{x}) (K + \lambda I_N)^{-1} \underline{t}$$

[please go

Physical significance :

(*) The dual formulation allows us to express the solution entirely in terms of the kernel function

(*) We recover the original formulation for $\underline{w}$: the solution for $\underline{a}$ can be expressed as a linear combination of the elements of $\underline{\phi}(\underline{x})$

(*) The prediction at $\underline{x}$ is a linear combination of the target values from the training set.

(*) complexity of the primal and dual formulations

primal : $\underbrace{\underline{w}}_{M\times 1} = \underbrace{\underbrace{\Phi^T}_{M\times N} \underbrace{\underline{a}}_{N\times 1}}_{M\times 1}$

solving for $\underline{w}$ will typically involve inverting an $M\times M$ matrix, and

typically, $N >> M$

Dual : $\underline{a} = (K + \lambda I_N)^{-1} \underline{t}$: inverting an $N\times N$ matrix

→ complexity-wise, not wise!

However, the dual formulation is entirely expressible in terms of the kernel function $k(\cdot,\cdot)$

If we use the kernel trick (if it is possible), we can work directly with kernels, and avoid the explicit introduction of the feature transformation $\phi(\underline{x})$. This allows us to use features of high (even infinite) dimensionality.

CONSTRUCTING KERNEL FUNCTIONS DIRECTLY

Example: $k(\underline{x}, \underline{z}) \triangleq (\underline{x}^T \underline{z})^2$

2-D

Recap: $\underbrace{k(\underline{x}, \underline{x}')}_{\text{kernel (scalar)}} = \underline{\phi}^T(\underline{x})\,\underline{\phi}(\underline{x}')$

original space

$\underline{\phi}$: mapping function

$$\underline{x}^T \underline{z} = [x_2 \;\; x_1] \begin{bmatrix} z_2 \\ z_1 \end{bmatrix} = x_2 z_2 + x_1 z_1$$

$$\Rightarrow (\underline{x}^T \underline{z})^2 = (x_2 z_2 + x_1 z_1)^2$$
$$= x_2^2 z_2^2 + 2\, x_1 z_1 x_2 z_2 + x_1^2 z_1^2$$

try to separate into $\boxed{\quad\quad\quad}$ $\boxed{\;}$

$\underline{\phi}^T(\underline{x})$ $\quad$ $\underline{\phi}(\underline{z})$

2-D $\quad$ 2-D

$$\begin{bmatrix} x_2^2 & \sqrt{2}\, x_2 x_1 & x_1^2 \end{bmatrix} \begin{bmatrix} z_2^2 \\ \sqrt{2}\, z_2 z_1 \\ z_1^2 \end{bmatrix}$$

$$\underline{\phi}(\underline{x}) = \begin{bmatrix} x_2^2 \\ \sqrt{2}\, x_2 x_1 \\ x_1^2 \end{bmatrix}$$

mapping, comprises all second order terms with a specific weighting between them.

$$\underline{x} \rightarrow \begin{bmatrix} x_2 \\ x_1 \end{bmatrix}$$

(2-D) (3-D)

(Mercer's Condition) Necessary & Sufficient condition for a function $k(\underline{x}, \underline{x}')$ to be a valid kernel :

The Gram Matrix K should be PSD $\forall\ \underline{x}_i$ $K(i,j) = k(\underline{x}_i, \underline{x}_j)$

i.e., $\underline{x}_i^T K \underline{x}_i \geq 0$

Properties: given valid kernels $k_1(\underline{x}, \underline{x}')$ and $k_2(\underline{x}, \underline{x}')$, the following new kernels will also be valid kernels:-

1) $k(\underline{x}, \underline{x}') = c\, k_1(\underline{x}, \underline{x}')$, $c > 0$

2) $k(\underline{x}, \underline{x}') = f(\underline{x})\, k_1(\underline{x}, \underline{x}')\, f(\underline{x}')$ → $f(\cdot)$ is a function

3) $k(\underline{x}, \underline{x}') = q\left(k_1(\underline{x}, \underline{x}')\right)$, $q(\cdot)$: polynomial with non-negative coefficients

4) $k(\underline{x}, \underline{x}') = \exp\left(k_1(\underline{x}, \underline{x}')\right)$

5) $k(\underline{x}, \underline{x}') = k_1(\underline{x}, \underline{x}') + k_2(\underline{x}, \underline{x}')$

6) $k(\underline{x}, \underline{x}') = k_1(\underline{x}, \underline{x}')\, k_2(\underline{x}, \underline{x}')$

7) $k(\underline{x}, \underline{x}') = k_3\left(\underline{\phi}(\underline{x}), \underline{\phi}(\underline{x}')\right)$, $\underline{\phi}(\underline{x}): \underline{x} \to \mathbb{R}^M$

8) $k(\underline{x}, \underline{x}') = k_a(\underline{x}_a, \underline{x}_a') + k_b(\underline{x}_b, \underline{x}_b')$

$\underline{x} = (\underline{x}_a, \underline{x}_b)$: $\underline{x}_a$ & $\underline{x}_b$ are not necessarily disjoint, $k_a(\cdot)$ and $k_b(\cdot)$ are valid kernel functions.

9) $k(\underline{x}, \underline{x}') = k_a(\underline{x}_a, \underline{x}_a')\, k_b(\underline{x}_b, \underline{x}_b')$

"Gaussian kernel"

Interesting Example: $k(\underline{x}, \underline{x}') = \exp\left(-\frac{\|\underline{x} - \underline{x}'\|^2}{2\sigma^2}\right)$

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

$$k(\underline{x}, \underline{x}') = \underbrace{\exp\left(\frac{-\underline{x}^T\underline{x}}{2\sigma^2}\right)}\cdot\underbrace{\exp\left(-\frac{\underline{x}'^T\underline{x}'}{2\sigma^2}\right)}\cdot\underbrace{\exp\left(\frac{\underline{x}^T\underline{x}'}{\sigma^2}\right)}$$

$$\exp\left(\frac{-\underline{x}^T\underline{x}}{2\sigma^2}\right)\underbrace{\exp\left(\frac{\underline{x}^T\underline{x}'}{\sigma^2}\right)}\exp\left(\frac{-\underline{x}'^T\underline{x}'}{2\sigma^2}\right)$$

$\underline{x}^T\underline{x}'$ is a kernel (linear kernel)

$\Rightarrow \left(\frac{1}{\sigma}\underline{x}^T\right)\left(\frac{1}{\sigma}\underline{x}'\right)$ is also a kernel

$\Rightarrow \frac{\underline{x}^T\underline{x}'}{\sigma^2}$ is a kernel

$\Rightarrow \underbrace{\exp\left(\frac{\underline{x}^T\underline{x}'}{\sigma^2}\right)}_{k_1(\underline{x}, \underline{x}')}$ is also a kernel

(∵ $\exp(k_1(\underline{x}, \underline{x}'))$ is a kernel)

$\Rightarrow f(\underline{x})\, k_1(\underline{x}, \underline{x}')\cdot f(\underline{x}')$ is also a kernel

$f(\underline{x}) = \exp\left(\frac{-\underline{x}^T\underline{x}}{2\sigma^2}\right)$, $\quad f(\underline{x}') = \exp\left(\frac{-\underline{x}'^T\underline{x}'}{2\sigma^2}\right)$

QED.