Basic Philosophy

* In some cases, formulating an original (primal) problem completely in terms of other variables (dual) is possible. There is no guarantee that given a primal problem, it should be possible to formulate a dual one in terms of another variable

* Even if one is able to formulate a dual problem, there is no guarantee that the dual problem may have a 'better' solution: better in term of the computational complexity, attractiveness in terms of a kernel trick.

Recap:

$$J(\underline{w}) = \frac{1}{2}\sum_{i=1}^{N}\{\underline{w}^T\underline{\phi}(\underline{x}) - t_i\}^2 + \frac{\lambda}{2}\underline{w}^T\underline{w}$$

$$\frac{\partial J(\underline{w})}{\partial \underline{w}} = 0 \xrightarrow{\text{gave}} \underline{w} = \Phi^T \underline{a}, \quad a_i \triangleq \frac{1}{\lambda}\left[\underline{w}^T\underline{\phi}(\underline{x}_i) - t_i\right]$$

Put in the original expression

$$J(\underline{a}) = \text{1st term} + \text{2nd term} + \text{3rd term} + \text{4th term}$$

$$-\underline{a}^T\Phi\Phi^T\underline{t} \qquad \frac{\lambda}{2}\underline{a}^T\Phi\Phi^T\underline{a} \qquad \frac{1}{2}\underline{t}^T\underline{t}$$

$$\frac{1}{2}\sum_{i=1}^{N}\{\underline{a}^T\Phi\,\underline{\phi}(\underline{x}_i)\}^2$$

s^Ts or s^2 will not give a 'nice' expression

trick: the square of a scalar can be written as ss^T

$$= \frac{1}{2} \sum_{i=1}^{N} \{ \underline{a}^T \Phi \, \underline{\phi}(\underline{x}_i) \} \{ \underline{a}^T \Phi \, \underline{\phi}(\underline{x}_i) \}^T$$

$$= \frac{1}{2} \sum_{i=1}^{N} \underline{a}^T \Phi \, \underbrace{\underline{\phi}(\underline{x}_i) \, \underline{\phi}^T(\underline{x}_i)} \, \Phi^T \underline{a}$$

Take the parts not involved in the summation, outside

$$= \frac{1}{2} \underline{a}^T \Phi \, \underbrace{\{ \sum_{i=1}^{N} \underline{\phi}(\underline{x}_i) \underline{\phi}^T(\underline{x}_i) \}}_{\text{consider this summation alone}} \Phi^T \underline{a}$$

write as an inner product in one of the two possible ways

$$\underbrace{\underbrace{\left[\underline{\phi}(\underline{x}_1) \; \underline{\phi}(\underline{x}_2) \cdots \underline{\phi}(\underline{x}_N) \right]}_{\Phi^T} \underbrace{\begin{bmatrix} \underline{\phi}^T(\underline{x}_1) \\ \underline{\phi}^T(\underline{x}_2) \\ \vdots \\ \underline{\phi}^T(\underline{x}_N) \end{bmatrix}}_{\Phi}}_{\Phi^T \Phi}$$

$\Rightarrow$ the first term $= \frac{1}{2} \underline{a}^T (\Phi \Phi^T)(\Phi \Phi^T) \underline{a}$

The complete expression at the optimal value becomes

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

We define the gram Matrix $K \triangleq \Phi \Phi^T$

What is this?

$$\underbrace{\underset{N\times M}{\Phi}\ \underset{M\times N}{\Phi^T}}_{N\times N} = \underset{N\times M}{\begin{bmatrix} \underline{\phi}^T(\underline{x}_1) \\ \underline{\phi}^T(\underline{x}_2) \\ \vdots \\ \underline{\phi}^T(\underline{x}_N) \end{bmatrix}} \underset{M\times N}{\left[\underline{\phi}(\underline{x}_1)\ \underline{\phi}(\underline{x}_2) \ldots \underline{\phi}(\underline{x}_N) \right]}$$

Now, what is $K(i,j)$? i'th row $\times$ j'th column

$$K(i,j) = \underbrace{\underset{1\times M}{\underline{\phi}^T(\underline{x}_i)}\ \underset{M\times 1}{\underline{\phi}(\underline{x}_j)}}_{\text{this is symmetric, scalar!}}$$

$= k(\underline{x}_i, \underline{x}_j)$ the kernel function

$$K(i,j) = \underline{\phi}^T(\underline{x}_i)\,\underline{\phi}(\underline{x}_j) = k(\underline{x}_i, \underline{x}_j)$$

$$J(\underline{a}) = \underbrace{\tfrac{1}{2}\underline{a}^T K K \underline{a}} - \underbrace{\underline{a}^T K \underline{t}} + \tfrac{1}{2}\underline{t}^T\underline{t} + \underbrace{\tfrac{\lambda}{2}\underline{a}^T K \underline{a}}$$

[dual]

optimisation theory $\rightarrow$ $\dfrac{\partial J(\underline{a})}{\partial \underline{a}} = 0$

Use result: for a quadratic form

$$\frac{\partial}{\partial \underline{a}}\left(\underline{a}^T K \underline{a}\right) = 2K\underline{a}$$

$$\frac{\partial J(\underline{a})}{\partial \underline{a}} = \frac{1}{2}\cdot 2KK\underline{a} - K\underline{t} + \frac{\lambda}{2}\cdot 2K\underline{a} = 0$$

$$\Rightarrow K\underline{t} = K(K + \lambda I_N)\underline{a}$$

assume K to be invertible $\quad \underline{a} = (K + \lambda I_N)^{-1}\underline{t}$

[please go to p 4]

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.

$$y(\underline{x}) = \underline{w}^T \underline{\phi}(\underline{x}) = (\underline{\Phi}\,\underline{a})^T \underline{\phi}(\underline{x}) = \underbrace{\underline{a}^T}_{1 \times N} \boxed{\underbrace{\underline{\Phi}}_{N \times M} \underbrace{\underline{\phi}(\underline{x})}_{M \times 1}}$$

$N \times 1$

scalar 1×1

consider

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

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

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

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

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

2-D 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 weighing between them.

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

(2-D) (3-D)