**MLCC: Machine Learning Crash Course** **Spring 2014**

# Lecture 4- Regularization Networks II: Kernels

*Lecturer: F.Odone–L. Rosasco*

In this class we introduce the concepts of feature map and kernel that allows to generalize Regularization Networks, and not only, well beyond linear models. Our starting point will be again Tikhonov regularization,

$$\min_{w\in\mathbb{R}^D} \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f_w(x_i)) + \lambda\|w\|^2. \tag{4.1}$$

## 4.1 Feature Maps

A feature map is a map

$$\Phi : X \to F$$

from the input space into a new space called feature space where there is a scalar product $\Phi(x)^T\Phi(x')$. The feature space can be infinite dimensional and the following notation is used for the scalar product $\langle \Phi(x), \Phi(x')\rangle_F$.

### 4.1.1 Beyond Linear Models.

The simplest case is when $F = \mathbb{R}^p$, and we can view the entries $\Phi(x)^j$, $j = 1, \ldots, p$ as novel measurements on the input points. For illustrative purposes consider $X = \mathbb{R}^2$. An example of feature map could be $x = (x_1, x_2) \mapsto \Phi(x) = (x_1^2, x_2^2, \sqrt{2}x_1x_2)$. With this choice if we now consider

$$f_w(x) = w^T\Phi(x) = \sum_{j=1}^{p} w^j\Phi(x)^j$$

we effectively have that the function is no longer linear but it is a polynomial of degree 2. Clearly the same reasoning holds for much more general choice of measurements (features), in fact *any* finite set of measurements. Although seemingly simple, the above observation allows to consider very general models. Figure 4.1 gives a geometric interpretation of the potential effect of considering a feature map. Points which are not easily classified by a linear model, can be easily classified by a *linear model in the feature space*. Indeed, the model is no longer linear in the original input space.

### 4.1.2 Computations.

While feature maps allow to consider nonlinear models, the computations are essentially the same as in the linear case. Indeed, it is easy to see that essentially the computations considered for linear models, under different loss functions, remain unchanged, as long as we change $x \in \mathbb{R}^D$ into $\Phi(x) \in \mathbb{R}^p$. For example, for least squares we simply need to replace the $n$ by $D$ matrix $X_n$ with a new $n$ by $p$ matrix $\Phi_n$, where each rows is the image of an input point in the feature space as defined by the feature map.

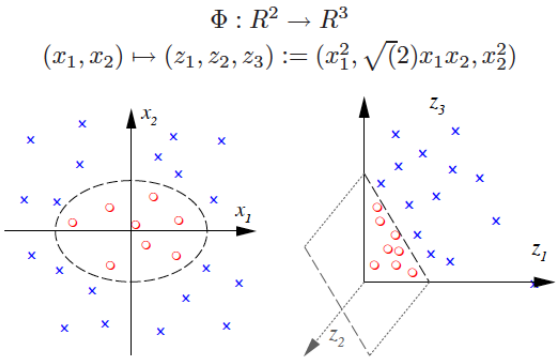


**Figure 4.1.** A pictorial representation of the potential effect of considering a feature map in a simple two dimensional example.

## 4.2 Representer Theorem

In this section we discuss how the above reasoning can be further generalized. The initial observation is that the solution of regularization problems of the form (4.1), can always be written as

$$\hat{w}^T = \sum_{i=1}^{n} x_i^T c_i, \tag{4.2}$$

where $x_1, \ldots, x_n$ are the inputs in the training set and $c = (c_1, \ldots, c_n)$ a set of coefficients. The above result is an instance of the so called representer theorem. While its proof is extremely simple, its implications are remarkable. Before discussing them we briefly sketch the main steps in the proof.

### 4.2.1 Representer Theorem's Proof Sketch

- The vectors of the form (4.2) form a linear subspace $\widehat{W}$ of $\mathbb{R}^D$. Hence for every $v \in \mathbb{R}^D$ we have the decomposition $w = \hat{w} + \hat{w}^\perp$, where $\hat{w} \in \widehat{W}$ and $\hat{w}^\perp$ belongs to the space $\widehat{W}^\perp$ of vectors orthogonal to those in $\widehat{W}$ , i.e.

$$\hat{w}^T \hat{w}^\perp = 0. \tag{4.3}$$

- The following is the key observation: for all $i = 1, \ldots, n$ $x_i \in \widehat{W}$, so that

$$f_w(x_i) = x_i^T w = x_i^T (\hat{w} + \hat{w}^\perp) = x_i^T \hat{w}.$$

  It follows that the empirical error depends only on $\hat{w}$!

- For the regularizer we have

$$\|w\|^2 = \|\hat{w} + \hat{w}^\perp\|^2 = \|\hat{w}\|^2 + \|\hat{w}^\perp\|^2,$$

  because of (4.3). Clearly the above expression is minimized if we take $\hat{w}^\perp = 0$.

The theorem is hence proved, the first term in (4.1) depends only on vector of the form (4.2) and the same form is the best to minimize the second term

### 4.2.2 Representer Theorem Implications

Using Equation (4.2) it possible to show how the vector $c$ of coefficients can computed considering different loss functions.

**Least Squares**. The vector of coefficients satisfies the following linear system

$$(K_n + \lambda n I)c = Y_n.$$

where $K_n$ is the $n$ by $n$ matrix with entries $(K_n)_{i,j} = x_i^T x_j$. The matrix $K_n$ is called the *kernel matrix* and is symmetric and positive semi-definite.

**Logistic Regression**, The vector of coefficients can be computed by the following iteration

$$c_t = c_{t-1} - \frac{\gamma}{n} B(c_{t-1}), \quad t = 1, \dots, T$$

for $c_0 = 0$, and where $B(c_{t-1}) \in \mathbb{R}^n$ with

$$B(c_{t-1})^i = \sum_{j=1}^{n} \frac{-y_j}{1 + e^{y_j \sum_{k=1}^{n} x_k^T x_j c_{t-1}^j}}.$$

**SVM** The coefficients are simply $c^i = y^i \alpha^i$ for $i = 1, \dots, n$, with $\alpha$ given by the solution of the dual problem,

$$\min_{\alpha \in \mathbb{R}^n} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j, \quad \text{subject to,} \quad \alpha_i \geq 0, \quad i = 1, \dots, n, \tag{4.4}$$

see previous lecture.

## 4.3 Kernels

One of the main advantages of using the representer theorem is that the solution of the problem depends on the input points only through inner products $x^T x'$. Kernel methods can be seen as replacing the inner product with a more general function $K(x, x')$. In this case, the representer theorem (4.2), that is $f_w(x) = w^T = \sum_{i=1}^{n} x_i^T c_i$, becomes

$$\hat{f}(x) = \sum_{i=1}^{n} K(x_i, x) c_i. \tag{4.5}$$

and we can promptly derive kernel versions of the Regularization Networks induced by different loss functions.

The function $K$ is often called a kernel and to be admissible it should *behave like* an inner product. More precisely it should be: 1) symmetric, and 2) positive definite, that is the kernel matrix $K_n$ should be positive semi-definite for any set of $n$ input points. While the symmetry property is typically easy to check, positive semi definiteness is trickier. Popular examples of positive definite kernels include:

- the linear kernel $K(x, x') = x^T x'$,
- the polynomial kernel $K(x, x') = (x^T x' + 1)^d$,
- the Gaussian kernel $K(x, x') = e^{-\frac{\|x-x'\|^2}{2\sigma^2}}$,

where the last two kernels have a tuning parameter, the degree and Gaussian width, respectively.

A positive definite kernel is often called a *reproducing kernel* and is a key concept in the theory of reproducing kernel Hilbert spaces.