# Regularization in Kernel Methods

based on Chapter 4 in *Learning with Kernels*

by B. Schölkopf and A. J. Smola

Alexander Ilin

Laboratory of Computer and Information Science

# Introduction

- The problem of learning: given a training sequence $(x_1, y_1), \ldots, (x_m, y_m), x_i \in \mathcal{X}, y_i \in \mathbb{R}$, we estimate a function $f : \mathcal{X} \to \mathbb{R}$ such that $y = f(x)$
- The *training error* (or *empirical risk*) is minimized:

$$R_{emp}[f] = c((x_1, y_1, f(x_1)), \ldots, (x_m, y_m, f(x_m)))$$

- Minimizing $R_{emp}[f]$ can lead to numerical instabilities and bad generalization performance
- Regularization: we avoid this problem by restricting the class of admissible solutions $f$

## The Key Idea of Regularization

- The class of possible minimizers $f \in \mathcal{F}$ of $R_{emp}[f]$ is restricted such that $\mathcal{F}$ becomes a compact set
- Directly specifying the compact set $\mathcal{F}$ would lead to a constrained optimization problem, which can be cumbersome in practice
- Instead, we complement the original objective function $R_{emp}[f]$ with a stabilization (regularization) term $\Omega[f]$
- Regularized risk functionals penalize possible solutions with "bad" properties

# Regularized Risk Functional

- We consider the following class of regularized risk functionals

$$R_{reg}[f] = R_{emp}[f] + \lambda\Omega[f]$$

- Small $\Omega[f]$ enforce the smoothness or simplicity of the solution $f$
- $\lambda > 0$ is the regularization (trade-off) parameter
- Example: SV classification

$$R_{reg}[f] = R_{emp}[f] + \frac{\lambda}{2}\|\mathbf{w}\|^2,$$

which means that we maximize the margin of classification

## Regularized Risk in Kernel Methods

- In Kernel methods, we might optimize $R_{reg}[f]$ in the reproducing kernel Hilbert space (RKHS) $f \in \mathcal{H}$ associated to a kernel $k$
- $\mathcal{H}$ contains all linear combinations of kernels centered in *arbitrary* points of $\mathcal{X}$
- We can rewrite $R_{reg}[f]$ in terms of the RKHS representation:

$$R_{reg}[f] = R_{emp}[f] + \frac{\lambda}{2}\|f\|^2_{\mathcal{H}},$$

## How $R_{reg}[f]$ Restricts The Solution

- **The Representer Theorem**: For any strictly monotonic increasing function $\Omega$, an arbitrary loss function $c$, each minimizer $f \in \mathcal{H}$ of the regularized risk

$$R_{reg}[f] = c((x_1, y_1, f(x_1)), \ldots, (x_m, y_m, f(x_m))) + \Omega\left(\|f\|_{\mathcal{H}}\right)$$

admits a representation of the form

$$f(x) = \sum_{i=1}^{m} \alpha_i k(x_i, x).$$

- Even though we might be trying to minimize $R_{reg}[f]$ in an infinite-dimensional space $\mathcal{H}$, the solution lies in the span of $m$ particular kernels – those centered on the training points!

# Regularization Operator

- In order to explain why kernel methods work, the authors introduce the regularization operator $\Upsilon : \mathcal{F} \rightarrow \mathcal{D}$ (a linear map)
- Space $\mathcal{D}$ contains transformed versions of the original functions $f$, where the transformation $\Upsilon$ "extracts" those parts that should be affected by the regularization
- The regularization term is then represented as

$$\Omega[f] = \frac{1}{2} \langle \Upsilon f, \Upsilon f \rangle_{\mathcal{D}}$$

- For common kernels, the associated transformation $\Upsilon$ extracts properties like *derivatives* of functions: these kernels induce a form of regularization that penalizes non-smooth functions

## Why Kernel Methods Work

- Fixing the regularization operator $\Upsilon$ determines the possible set of functions that we might obtain, independently of the class of functions in which we expand the estimate $f$ (Theorem 4.9)
- Operator $\Upsilon$ restricts the RKHS of possible solutions to a subspace of the eigenvector decomposition of $\Upsilon^*\Upsilon$
- Thus, SVMs are simply a very convenient way of specifying the regularization and a matching class of basis functions via one kernel function

## Semiparametric Regularization

- We want to use some additional knowledge we may have about the solution
- For instance, we may know that the major properties of the data are described by a combination of a set of functions $\{\psi_j\}_{j=1}^n$
- These reasons motivate the construction of *semiparametric* models: $f(x) = g(x) + h(x)$, $g \in \mathcal{H}$, $h \in \text{span}\{\psi_j\}_{j=1}^n$
- These types of models are both easy to understand (due to the parametric part) and perform well (often thanks to the nonparametric term)

## Backfitting

- A common approach is to fit the data with the parametric model and train the nonparametric add-on using the errors of the parametric part
- In most cases, however, this does not lead to the minimum of the regularized risk functional
- Example: SV regression machine with linear kernel $k(x, x') = \langle x, x' \rangle$ in one dimension, and a constant term as parametric part ($f(x) = wx + \beta$)

# Global Solution

- It is better instead to solve a convex optimization problem, as in standard SVMs, but with a different set of admissible functions:

$$f(x) = g(x) + h(x)$$

  where $g \in \mathcal{H}$, $h \in$ span of $n$ functions $\psi_1, \ldots, \psi_n$

- The regularized risk ($\psi_1, \ldots, \psi_n$ are not regularized):

$$R_{reg}[f] = c((x_1, y_1, f(x_1)), \ldots, (x_m, y_m, f(x_m))) + \Omega\left(\|g\|_{\mathcal{H}}\right)$$

- Theorem states: The solution admits the form:

$$f(x) = \sum_{i=1}^{m} \alpha_i k(x_i, x) + \sum_{j=1}^{n} \beta_j \psi_j(x)$$

## Example: Semiparametric SV Regression (1)

- For Support Vector regression with the $\varepsilon$-intensive loss we have

$$
\begin{aligned}
c((x_i, y_i, f(x_i))_{i \in [m]}) &= \frac{1}{m} \sum_{i=1}^{m} |y_i - f(x_i)|_\varepsilon \\
\Omega[\|f\|] &= \frac{\lambda}{2} \|g\|^2
\end{aligned}
$$

- $\lambda > 0$ and $\varepsilon \geq 0$ are fixed parameters which determine the trade-off between regularization and fit to the training set
- In addition, a single ($M = 1$) constant function $\psi_1(x) = 1$ is used as an offset, and is not regularized by the algorithm

## Example: Semiparametric SV Regression (2)

- Using $n$ functions $\psi_j$ in the parametric part, we eventually arrive at the optimization problem:

$$
\begin{array}{lrl}
\text{maximize} & - & \displaystyle \frac{1}{2}\sum_{i,j=1}^{m}(\alpha_i-\alpha_i^*)(\alpha_i-\alpha_i^*)k(x_i,x_j) \\
 & - & \displaystyle \varepsilon\sum_{i=1}^{m}(\alpha_i+\alpha_i^*)+\sum_{i=1}^{m}y_i(\alpha_i-\alpha_i^*) \\
\text{subject to} & & \displaystyle \sum_{i=1}^{m}(\alpha_i-\alpha_i^*)\psi_j(x_i)=0, \quad 1\le j\le n \\
 & & \alpha_i,\alpha_i^*\in[0,1/\lambda]
\end{array}
$$

## Coefficient-Based Regularization

- Instead of regularization in RKHSs, we now use a regularization operator that acts *directly* on the space of coefficients $\alpha_i$ of the expansion

$$f(x) = \sum_{i=1}^{n} \alpha_i k(x_i', x)$$

- It is possible that $x_i'$ and the training patterns $x_i$ do not coincide and $m \neq n$

- $\Omega[f]$ is written as a function of coefficients $\alpha_i$: $\Omega[f] = \Omega[\alpha]$

- The motivation is for instance the heuristic that small coefficients generally translate into simple functions

## Ridge Regression

- A popular choice to regularize linear combinations of basis functions is to use a weight decay term, which penalizes large weights:

$$\Omega[f] = \frac{1}{2}\sum_{i=1}^{n}\alpha_i^2 = \frac{1}{2}\|\alpha\|^2$$

- It is possible to find an equivalent regularization operator:

$$\Omega[f] = \frac{1}{2}\langle \Upsilon f, \Upsilon f\rangle = \frac{1}{2}\|\alpha\|^2$$

- We can get the same regularization under the assumption that the penalities on $\alpha_i$ are given by a prior probability

## Linear Programming Regularization

- The disadvantage of the squared penalty: some kernel functions $k(x_i, x)$ may not contribute much to the solution but still appear in the function expansion
- A regularizer whose derivative does not vanish in the neighbourhood of 0 will not exhibit such problems:

$$\Omega[f] = \sum_i |\alpha_i|$$

- We arrive at an optimization problem with linear constraints (linear programming problem)

## Example: Linear Programming SV Regression

- The regularized risk minimization problem:

$$\text{minimize } R_{reg}[f] = \lambda \sum_{i=1}^{m} |\alpha_i| + \sum_{i=1}^{m} (\xi_i + \xi_i^*)$$

$$\text{subject to } \begin{cases} y_i - \sum_{j=1}^{m} \alpha_j k(x_j, x_i) - \sum_{j=1}^{n} \phi_j(x_i) - b \leq \varepsilon + \xi_i \\ \sum_{j=1}^{m} \alpha_j k(x_j, x_i) + \sum_{j=1}^{n} \phi_j(x_i) + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases}$$

- We can replace $\alpha_i$ with $\alpha_i - \alpha_i^*$, $|\alpha_i|$ with $\alpha_i + \alpha_i^*$, and require $\alpha_i, \alpha_i^* \geq 0$.

## Mixed Semiparametric Regularizers

- We construct a mixture of linear and quadratic regularizers:

$$\Omega[f] = \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{n} |\beta_i|$$

- This is the SV estimation model with an additional linear regularization term added for the parametric part

- The common idea is to have two different sets of basis functions which are regularized differently, or a subset that is not regularized at all

## Conclusions

- Regularized risk functional was introduced
- Regularized risk restricts the solution
- The notion of the regularization operator was presented to explain how the solution is regularized
- Different types of regularization were motivated and discussed
- The nice property of SVMs was discovered: For common choices of kernels, the mapping into feature space is not arbitrary, but corresponds to useful regularization operators

## Problems

- Problem 4.2, p. 123
- Problem 4.3, p. 123
- You can substitute any of these problems by Problems 4.5, 4.6 on p.123 if you find them nicer