Kernel-Based Learning & Multivariate Model

MIRI Master

Lluís A. Belanche

belanche@cs.upc.edu

Soft Computing Research Group

Universitat Politècnica de Catalunya

2019-2020

Kernel-Based Learning & Multivariate Model

Syllabus

Sep 10 Introduction to kernel-based learning

Sep 17 The SVM for classification, regression & novelty detection (I)

Oct 01 The SVM for classification, regression & novelty detection (II

Oct 08 Kernel design (I): theoretical issues

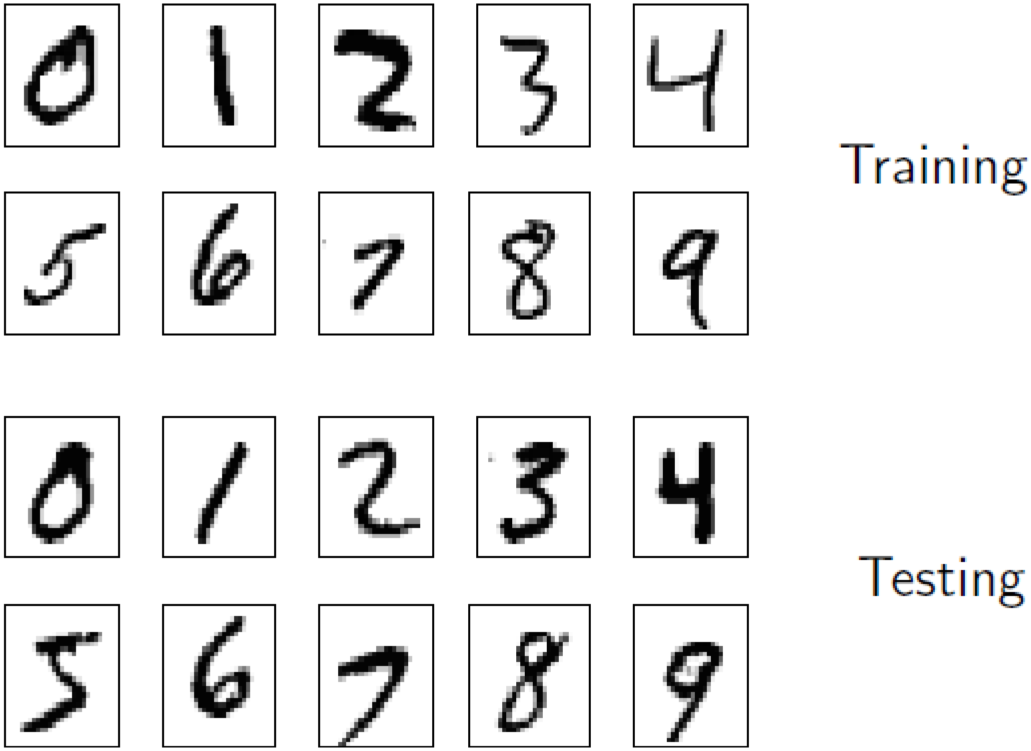
Oct 15 Kernel design (II): practical issues

Oct 22 Kernelizing ML & stats algorithms

Oct 29 Advanced topics

Support Vector Machines

Application (digit recognition)



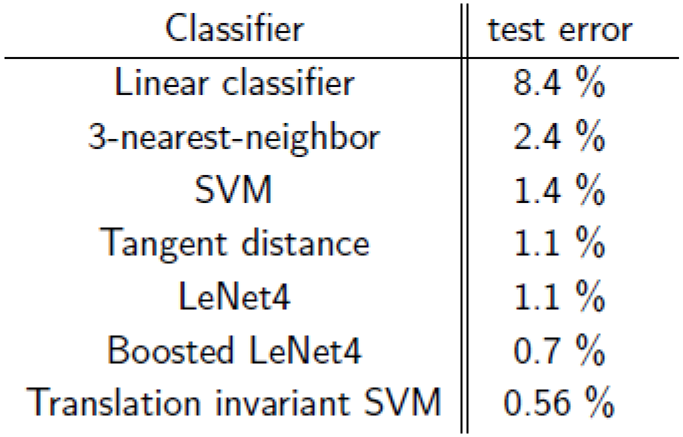
- Handwritten zip code recognition traces back to the 1960's

Support Vector Machines

Application (digit recognition)

- MNIST handwritten zip code recognition

- 60.000 training, 10.000 test examples (28 × 28 pixels)



Classifier	test error
Linear classifier	8.4 %
3-nearest-neighbor	2.4 %
SVM	1.4 %
Tangent distance	1.1 %
LeNet4	1.1 %
Boosted LeNet4	0.7 %
Translation invariant SVM	0.56 %

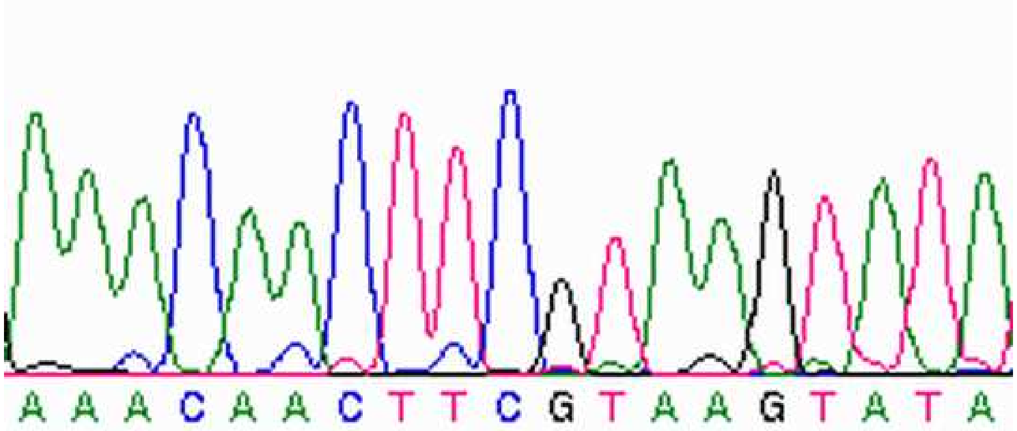
Support Vector Machines

Application: Classification of DNA sequences

- A *promoter* is a region of DNA that initiates or facilitates transcriptio of a particular gene

- The dataset consists of 106 DNA sequences described by 57 cat egorical variables, represented as the nucleotide at each positio [A]denine, [C]ytosine, [G]uanine, [T]hymine

- The response is a binary class: "+" for a promoter gene and "−" f a non-promoter gene



Support Vector Machines

Application: Classification of DNA sequences

The similarity between two multivariate categorical vectors is the fractio of the number of matching values.

Overlap/Dirac kernel]

$$k_0(\boldsymbol{x}, \boldsymbol{x}') = \frac{1}{d} \sum_{i=1}^{d} \mathbb{I}_{\{x_i = x'_i\}}$$

Another kernel that can be used is the RBF kernel:

Gaussian Radial Basis Function kernel:

$$k_{\text{RBF}}(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\gamma ||\boldsymbol{x} - \boldsymbol{x}'||^2\right) , \gamma > 0$$

In order to use this kernel, categorical variables with m modalities a coded using a binary expansion representation.

Support Vector Machines

Application: Classification of DNA sequences

Univariate kernel $k_1^{(U)}$:

$$k_1^{(U)}(x_i, x_i') = \begin{cases} h_\alpha(P_Z(x_i)) & \text{if } x_i = x_i' \\ 0 & \text{if } x_i \neq x_i' \end{cases}$$

where

$$h_\alpha(z) = (1 - z^\alpha)^{1/\alpha},\ \alpha > 0$$

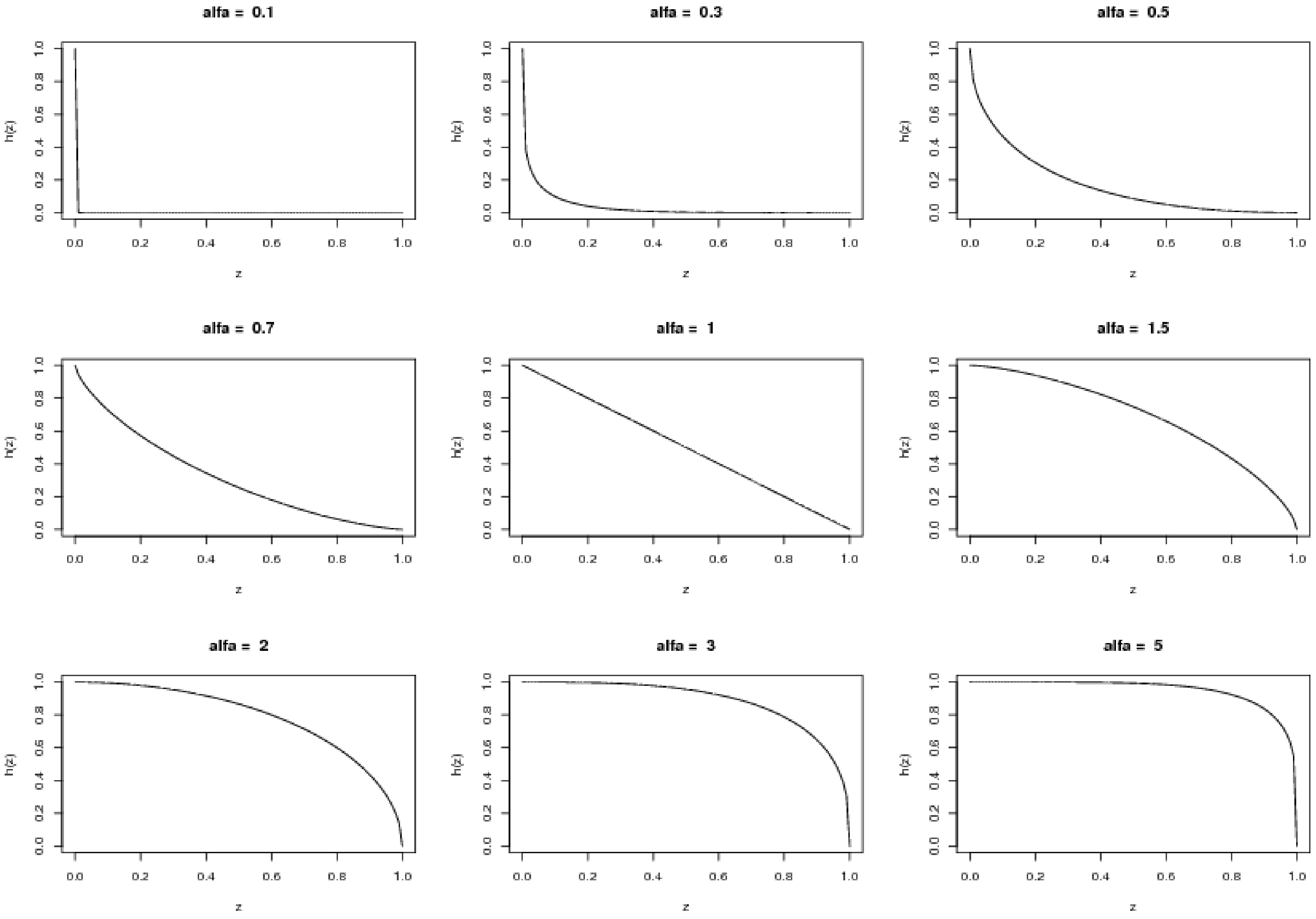
Multivariate kernel k_1:

$$k_1(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(\frac{\gamma}{d} \sum_{i=1}^{d} k_1^{(U)}(x_i, x_i')\right),\ \gamma > 0$$

The kernel matrices generated by k_1 are p.s.d.

Support Vector Machines

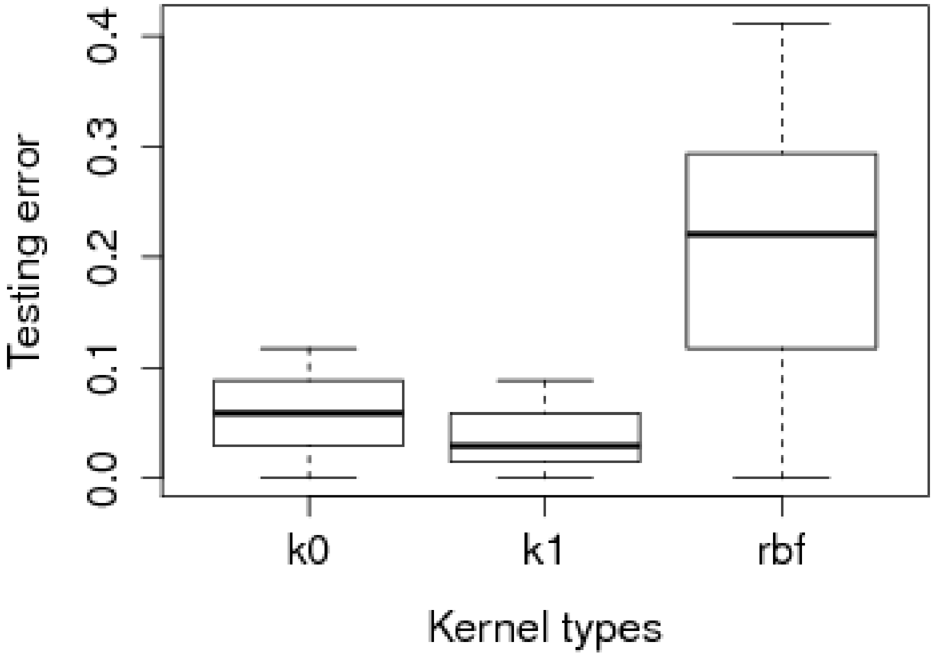
Application: Classification of DNA sequences



The family of inverting functions $h_\alpha(z)$ for different values of α

Support Vector Machines

Application: Classification of DNA sequences



Test error distributions on the PromoterGene problem

(*joint work with M. Villegas*)

Support Vector Machines

The role of the C parameter

Increasing the value of C ...

- penalizes margin errors more ⇒ narrower margin ⇒ larger VC-dimensi

- allows the $\alpha_i \leq C$ to be larger (so more opportunities for outliers)

- increases training times

Support Vector Machines

The role of the C parameter

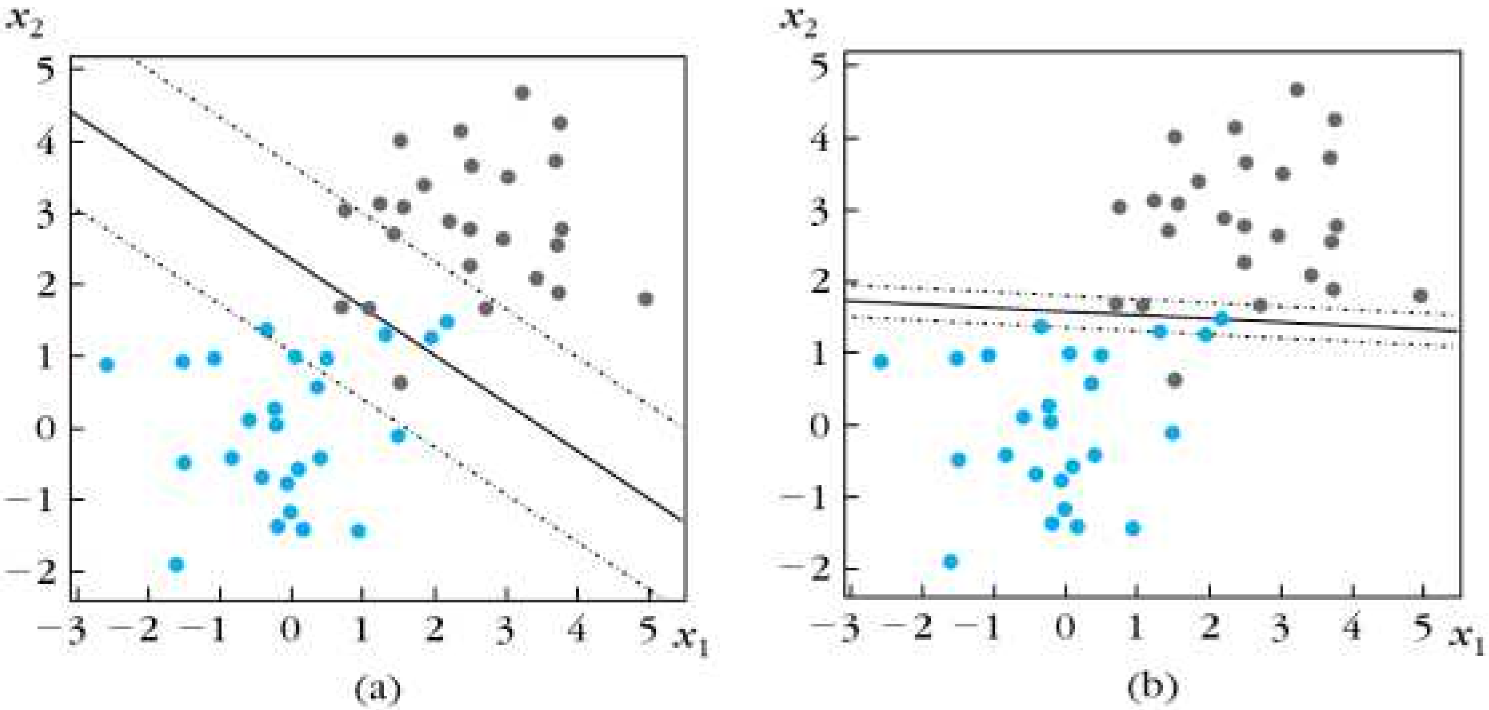


FIGURE 3.13

An example of two nonseparable classes and the resulting SVM linear classifier (full line) with the associated margin (dotted lines) for the values (a) $C = 0.2$ and (b) $C = 1000$. In the latter case, the location and direction of the classifier as well as the width of the margin have changed in order to include a smaller number of points inside the margin.

–from *Pattern Recognition (Fourth Edition)*, S. Theodoridis and K. Koutroumbas

Support Vector Machines

ν-SVMs

There are two commonly used versions of the SVM for classification:

'C-SVC': original SVM formulation, uses a parameter $C \in (0, \infty)$ to apply a penalty to the optimization for those data points not entire separated by the OSH (violating the margins)

'nu-SVC': C is replaced by $\nu \in (0, 1)$:

- upper bound on the fraction of examples which are training erro (missclassified)
- lower bound on the fraction of points which are SVs.

Support Vector Machines

SVMs for regression

“The Support Vector method can also be applied to the case of regressio maintaining all the main features that characterise the maximal marg algorithm: a non-linear function is learned by a linear learning machine in kernel-induced feature space while the capacity of the system is controlle by a parameter that does not depend on the dimensionality of the space

–from N. Cristianini and J. Shawe-Taylor, *An introduction to Support Vector Machin* (2000)

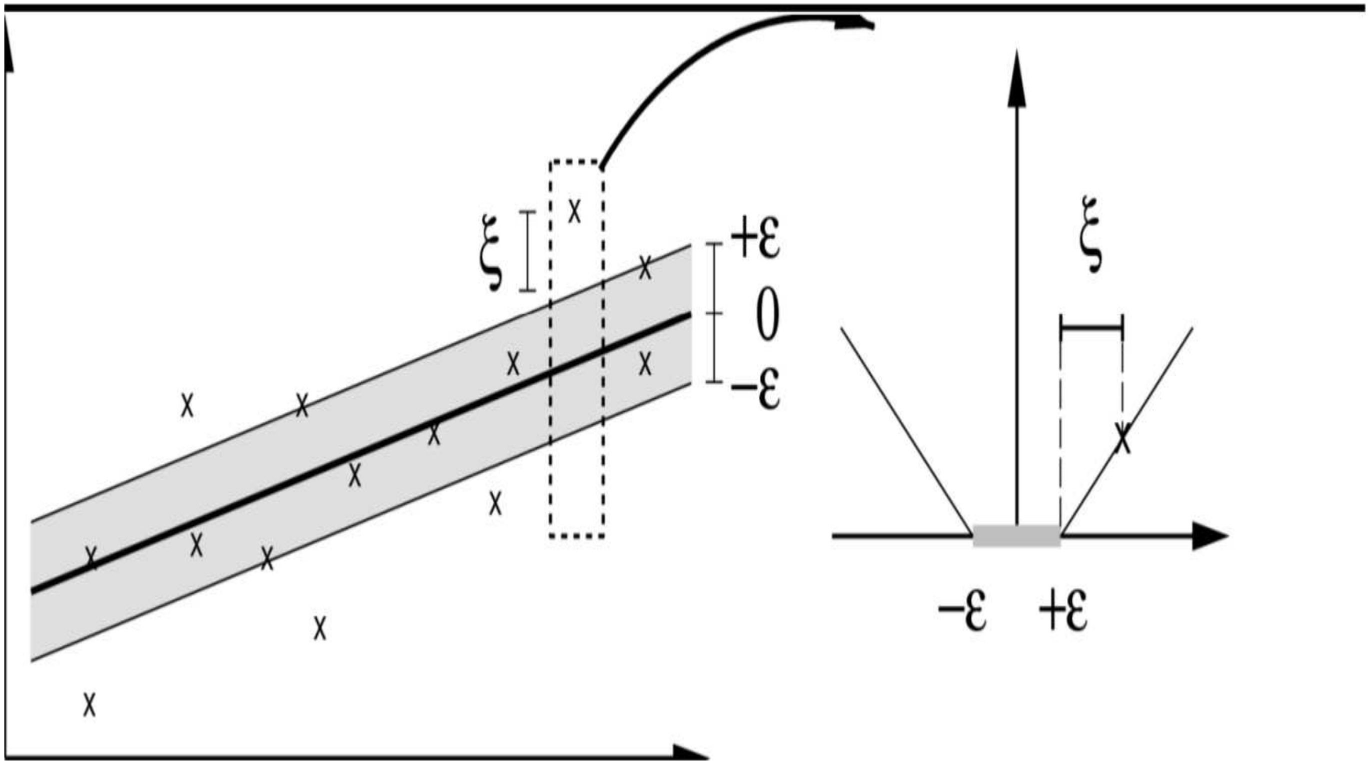
Support Vector Machines

SVMs for regression

Here we choose the ε-**insensitive** loss:

$$L(t_i, \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle) = |t_i - g(\boldsymbol{x}_i)|_\varepsilon = \max(|t_i - g(\boldsymbol{x}_i)| - \varepsilon, 0)$$

where $g(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b$



Support Vector Machines

SVMs for regression

minimize $$\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*)$$

subject to $$\begin{aligned} &\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b - t_i \leq \varepsilon + \xi_i, \\ &t_i - \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b \leq \varepsilon + \xi_i^*, \\ &\xi_i, \xi_i^* \geq 0 \end{aligned}$$

where the ξ_i, ξ_i^* are again slack variables controlling the “violations”

Support Vector Machines

SVMs for regression

Then feature maps $\phi(\cdot)$ are introduced, the primal optimization proble is transformed into the dual, and kernelised, to give:

$$y_{\mathsf{SVM}}(\boldsymbol{x}) = \sum_{i=1}^{n} \beta_i k(\boldsymbol{x}, \boldsymbol{x}_i)$$

with $0 \leq \alpha_i, \alpha_i^* \leq C$

For convenience, we have defined $\beta_i := \alpha_i - \alpha_i^*$.

Support Vector Machines

SVMs for regression

A closer look at the structure of the solution:

- Data points that end up **within** the ε-tube have inactive slacks (*i.e* $\xi_i = \xi_i^* = 0$) and therefore $\beta_i = 0$ (**not SVs**)

- Data points that end up **not within** the ε-tube have exactly one activ slack (*i.e.*, either $\xi_i > 0$ and $\xi_i^* = 0$, or vice versa) and therefore $\beta_i \neq$ (**non-bound SVs**)

- Data points that end up **outside** the ε-tube have exactly one bour slack (*i.e.*, $\xi_i = C$ and $\xi_i^* = 0$, or vice versa) and therefore $\beta_i \neq$ (**bound SVs**)

Support Vector Machines

SVMs for regression

In comparison to ridge regression, the only difference is in the choice o the loss (since both are **regularized machines** and both are amenable t **kernelisation**) and its consequences:

- Deviations lower than ε are ignored

- The loss grows linearly (and not quadratically) in the residual, makin it more robust against outliers

- The solution is **sparse** (the number of **basis functions** $\phi(\boldsymbol{x}_i)$ is au tomatically adapted)

Support Vector Machines

C versus ε

- C determines the trade off between model complexity (flatness) ar tolerance to deviations larger than ε

- ε controls the width of the ε-insensitive tube

Larger ε or C implies less SVs (while smaller ε or C implies more SVs); bu larger ε gives flatter models while larger C implies more complex mode

---

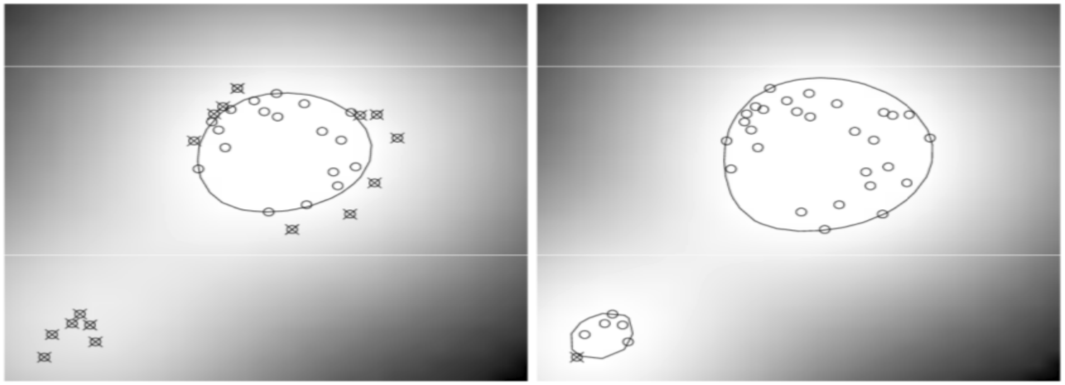
Hence, both parameters affect model complexity and number of SVs (bu in a different way).

Support Vector Machines

SVMs for novelty detection (I)

- You are given a dataset drawn from a pdf $p(\boldsymbol{x})$; the $\boldsymbol{x}$ can be hand written digits (recognizable/strange), process status (normal/faulty credit card transactions (normal/fraudulent), ...

- The goal is to estimate a "simple" subset S of input space s.t. th *probability* that a test point drawn from p lies *outside* S equals som a priori specified $\rho \in (0, 1)$:



–from Alex Smola: Hilbert Space Methods: Basics, Applications, Open Problems

`http://alex.smola.org/talks/rsisesvm.pdf`

Support Vector Machines

SVMs for novelty detection (II)

USPS dataset of handwritten digits: $9,298$ digit images of size 16×16

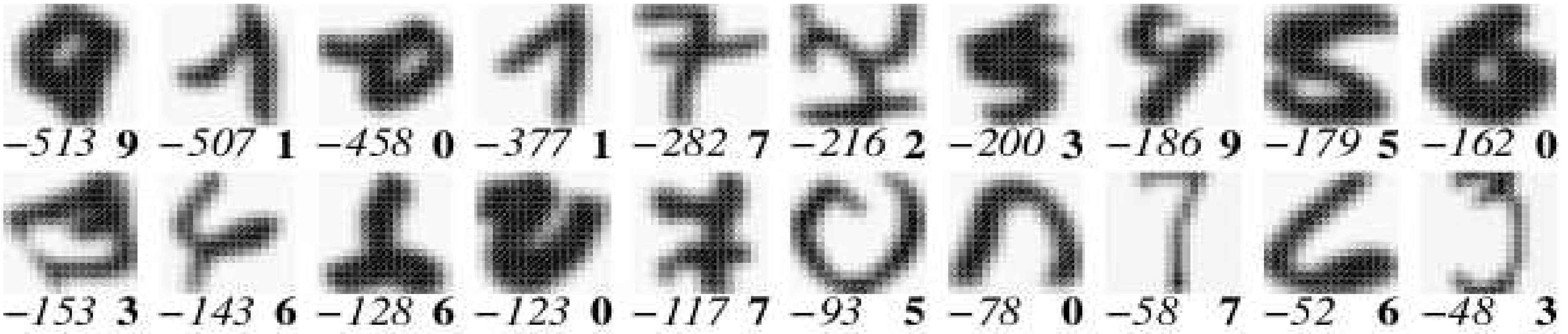


Figure 2: Outliers identified by the proposed algorithm, ranked by the negative output of the SVM (the argument of the sgn in the decision function). The outputs (for convenience in units of 10^{-5}) are written underneath each image in italics, the (alleged) class labels are given in bold face. Note that most of the examples are "difficult" in that they are either atypical or even mislabelled.

The 20 worst outliers for the USPS test set (here $\rho = 0.05$)

–from Schölkopf et al, *Support Vector Method for Novelty Detection*, NIPS'2000

Support Vector Machines

Tricks of the trade for the kernel

1. **Standardizing** the variables is in general good (assumed numerical

2. Kernel matrices close to the identity or close the the "all-ones" matr are also an indication of bad kernel parameter: avoid these situatior

3. Kernel matrix values should not be very large or very small (or both if so, **normalize** the kernel matrix