Basics of Machine Learning

TSIA-SD 210 - Lecture 3

3.1 NonLinear Support Vector Machines and Kernels

3.2 ML Methodology

Florence d'Alché-Buc

Contact: florence.dalche@telecom-paristech.fr,
2A Filière SD, Télécom ParisTech,Université of Paris-Saclay, France

Outline

Reminder

Nonlinear SVM and Kernels

Methodology ctd': ROC curves

References

Supervised binary classification

Probabilistic and Statistical Framework 1/2

- Let X be a random vector $\mathcal{X} = \mathbb{R}^p$
- and Y be a discrete random variable$\mathcal{Y} = \{-1, 1\}$
- Let $\mathbb{P}$ be the joint probability law of (X,Y)
- LEt $\mathcal{S}_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, i.i.d. sample from $\mathbb{P}$.

Supervised binary classification

Probabilistic and Statistical Framework 2/2

- Let $f : \mathbb{R}^p \rightarrow \{-1, +1\}$ a binary classification : $f(x) = \text{sign}(h(x))$ with $h : \mathbb{R}^p \rightarrow \mathbb{R} \in \mathcal{H}$
- Let $\ell : \{-1, +1\} \times \mathbb{R} \rightarrow \mathbb{R}$ be a local loss function
- Empirical risk : $R_n(h) = \frac{1}{n} \sum_i \ell(y_i, h(x_i))$, Regulatizing term: $\Omega(h)$ which measures *complexity* de h.
- We search for : $\hat{h} = \text{arg min}_{h \in \mathcal{H}} R_n(h) + \lambda \Omega(h)$

Outline

Reminder

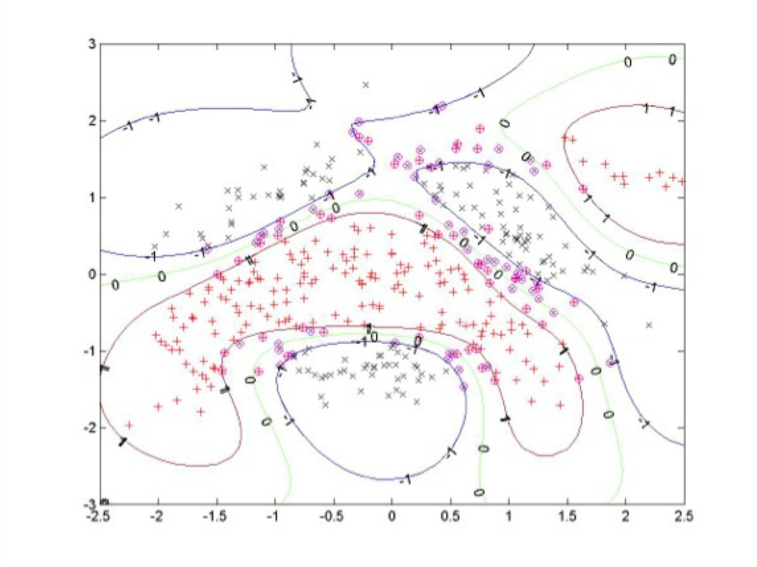
Nonlinear SVM and Kernels

- Kernels
- Back to VC-dimension and generalization bounds
- Support Vector Regression

Methodology ctd': ROC curves

References

Support Vector Machine : nonlinear frontiers in 2D space



Remark 1

Finding the Optimal Margin Hyperplane does involve training data only through inner products.

$$\begin{array}{ll} \max\limits_{\alpha} & \sum\limits_i \alpha_i - \frac{1}{2}\sum\limits_{i,j} \alpha_i\alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \\ \text{sous les contraintes} & 0 \leq \alpha_i \leq C \; i = 1, \ldots, n. \\ & \sum\limits_i \alpha_i y_i \; i = 1, \ldots, n. \end{array}$$

Let us use a feature map

If data are transformed according a nonlinear feature map $\phi : \mathcal{X} \to \mathcal{F}$, and if we know how to compute all the inner products $\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$, then we are able to learn a nonlinear decision frontier.

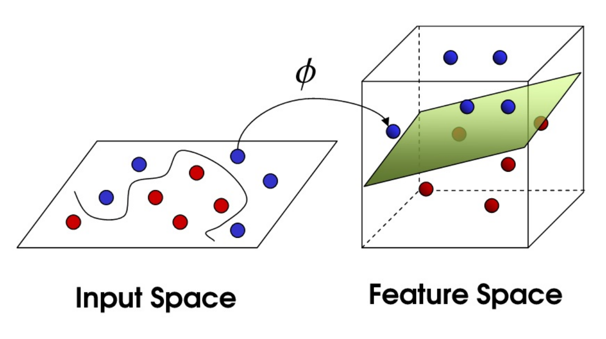
$$\begin{aligned} \max_{\alpha} \quad & \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \\ \text{sous les contraintes} \quad & 0 \leq \alpha_i \leq C \; i = 1, \ldots, n. \\ & \sum_i \alpha_i y_i \; i = 1, \ldots, n. \end{aligned}$$

To classify a new datapoint $\mathbf{x}$, we only need to be able to calculate $\phi(\mathbf{x})^T \phi(\mathbf{x}_i)$.

Kernel trick

If we substitute $\mathbf{x}_i^T\mathbf{x}_j$ by the image of a function k : $k(\mathbf{x}_i, \mathbf{x}_j)$ such that there exists a feature space $\mathcal{F}$ and feature map $\phi : \mathcal{X} \rightarrow \mathcal{F}$ et $\forall(\mathbf{x}, \mathbf{x}') \in \mathcal{X}, k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T\phi(\mathbf{x}')$, then We are able to apply the same learning algorithm (Optimal Margin Hyperplane) and we get
$f(\mathbf{x}) = \text{signe}(\sum_{i=1}^{n} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b)$
Such functions do exist and they are called PDS kernels: positive definite symmetric kernels.

Kernel trick and feature space 1/2



Kernel trick and feature space 2/2

$$\begin{tikzcd}
\mathscr{F}_{\mathrm{x}} \arrow[rd, "g"] & \\
\mathscr{X} \arrow[u, "\phi_{\mathrm{x}}"] \arrow[r, "h"'] & \mathbb{R}
\end{tikzcd}$$

Kernel trick and feature space 2/2

$$\begin{tikzcd}
\mathscr{F}_{\mathrm{x}} \arrow[dr, "\mathbf{g}"] & \\
\mathscr{X} \arrow[u, "\phi_{\mathrm{x}}"] \arrow[r, "\mathrm{h}"'] & \mathbb{R}
\end{tikzcd}$$

$$h(\mathbf{x}) = \sum_{i=1}^{n} \beta_i \phi(x)^T \phi(x_i) = \sum_{i=1}^{n} \beta_i k(x, x_i),$$

with $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ a positive definite symmetric kernel.

Outline

Reminder

Nonlinear SVM and Kernels

Kernels

Back to VC-dimension and generalization bounds

Support Vector Regression

Methodology ctd': ROC curves

References

Kernels

Définition
Let $\mathcal{X}$ be a non empty set. Let k:$\mathcal{X} \times \mathcal{X} \to \mathbb{R}$, be a symmetric function. Function k is called a Positive Definite Symmetric kernel if and only if for any finite set of size m, $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subset \mathcal{X}$, and any column vector $\mathbf{c} \in \mathbb{R}^m$,
$\mathbf{c}^T K \mathbf{c} = \sum_{i,j=1}^{m} c_i c_j k(x_i, x_j) \geq 0$

NB: any finite Gram matrix built from k and a finite number of elements of $\mathcal{X}$ is semi-definite positive

Kernel properties

Moore-Aronzajn Theorem
Let K be PDS kernel. Then, there exists a Hilbert Space called *Feature Space* and a function called a *feature map* $\phi : \mathcal{X} \to \mathcal{F}$, such that $\forall (x, x') \in \mathcal{X}^2, \langle \phi(x), \phi(x') \rangle_{\mathcal{F}} = k(x, x')$.
Moreover, there exists a unique feature space $\phi(x) = k(\cdot, x) \in \mathcal{F}$ that satisfies the reproducing property, i.e.:

$$\forall f \in \mathcal{F}, \forall x \in \mathcal{X}, \langle k(\cdot, x), f \rangle_{\mathcal{F}} = f(x)$$

We refer to this kernel as the canonical one.
Please also notice that:

$$\forall (x, x') \in \mathcal{X}^2, \langle k(\cdot, x), k(\cdot, x') \rangle_{\mathcal{F}} = k(x, x')$$

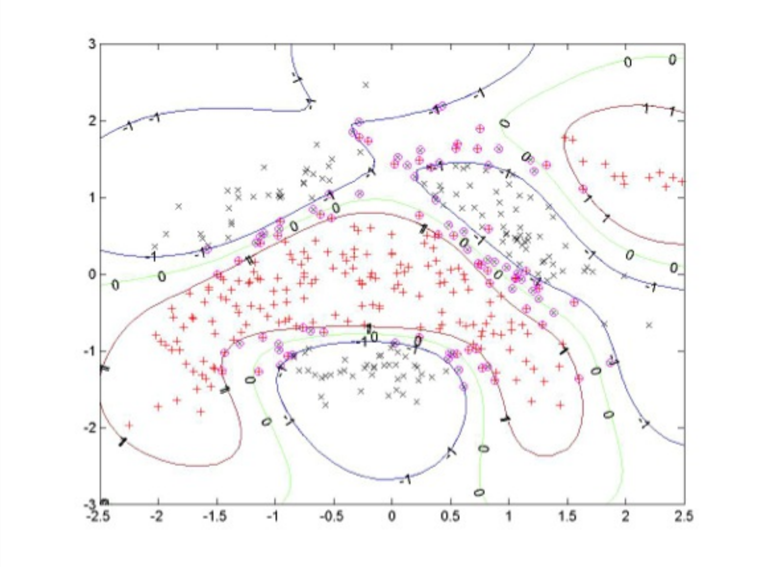
Kernels

Kernel between vectors

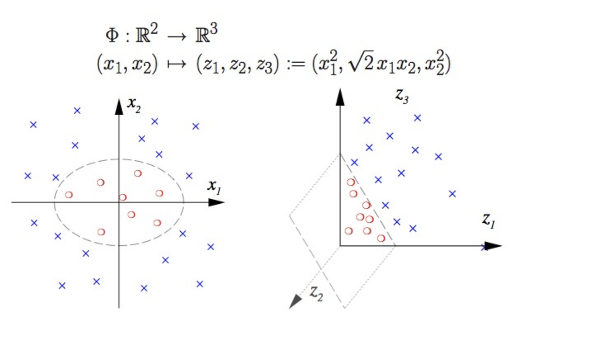
$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^p$

- Trivial linear kernel : $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{x}'$
- Polynomial kernel : $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}' + c)^d$
- Gaussian kernel : $k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma ||\mathbf{x} - \mathbf{x}'||^2)$

Support Vector Machine



Example: polynomial kernel



$$\Phi : \mathbb{R}^2 \to \mathbb{R}^3$$
$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}\,x_1 x_2, x_2^2)$$

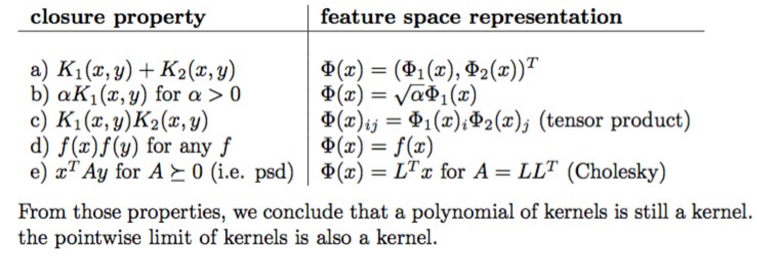
Exemple : noyau polynomial

Kernel trick

We notice that $\phi(\mathbf{x}_1)^T\phi(\mathbf{x}')$ can be computed without working in $\mathbb{R}^3$

We can define directly $k(\mathbf{x},\mathbf{x}') = \phi(\mathbf{x})^T\phi(\mathbf{x}') = (\mathbf{x}^T\mathbf{x}')^2$

Closure properties of kernels



closure property	feature space representation
a) $K_1(x,y) + K_2(x,y)$	$\Phi(x) = (\Phi_1(x), \Phi_2(x))^T$
b) $\alpha K_1(x,y)$ for $\alpha > 0$	$\Phi(x) = \sqrt{\alpha}\Phi_1(x)$
c) $K_1(x,y)K_2(x,y)$	$\Phi(x)_{ij} = \Phi_1(x)_i \Phi_2(x)_j$ (tensor product)
d) $f(x)f(y)$ for any f	$\Phi(x) = f(x)$
e) $x^T Ay$ for $A \succeq 0$ (i.e. psd)	$\Phi(x) = L^T x$ for $A = LL^T$ (Cholesky)

From those properties, we conclude that a polynomial of kernels is still a kernel. the pointwise limit of kernels is also a kernel.

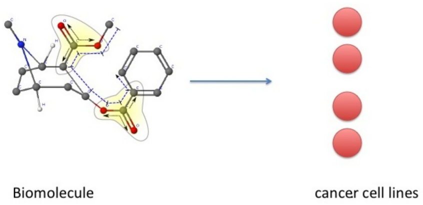
Much more interesting: kernels for complex objects

Kernels for

- **Complex (unstructured) objects**: texts, images, documents, signal, biological objects (gene, mRNA,protein, ...), functions, histograms
- **Structured objects**: sequences, trees, graphs, any composite objects

This made the success of kernels in computational biology, information retrieval (categorization for instance), but also in unexpected areas such as software metrics ....

Example: predict the property of a molecule



- **Inputs** : molecule (drug candidate)
- **Output** : activity on a cancer line (or several cancer lines)

A regression problem from structured data.

Kernel for labeled graphs

For a given length L, let us first enumerate all the paths of length $\ell \leq L$ in the training dataset (data are molecule = labeled graphs). Let m be the size of this (huge) set. For a graph, define $\phi(G) = (\phi_1(G), \ldots, \phi_m(G), \ldots, \phi_L(G))^T$ where (T) is 1 if the m^{th} path appears in the labeled graph G, and 0 otherwise.

Kernel for labeled graphs

Definition 1:

$$k_L(G, G') = < \phi(G), \phi(G') >$$

Tanimoto kernel

$$k_L^t(G, G') = \frac{k_L(G, G')}{k_L(G, G) + k_L(G', G') - k_L(G, G')}$$

idea: k_m^t calculates the ratio between the number of elements of the intersection of the two sets of paths (G and G' are seen as bags of paths) and the number of elements of the union of the two sets.
Reference: Ralaivola et al. 2005, Su et al. 2011

Convolution kernels

Definition:
Suppose that $x \in \mathcal{X}$ is a **composite structure** and $x_1, \ldots, x_D$ are its "parts" according a relation R such that $(R(x, x_1, x_2, \ldots, x_D)$ is true, with $x_d \in \mathcal{X}_d$ for each $1 \leq d \leq D$, D being a positive integer. k_d be a PDS kernel on a set $\mathcal{X} \times \mathcal{X}$, for all (x,x'), we define:

$$k_{conv}(x, x') = \sum_{(x_1,\ldots,x_d) \in R^{-1}(x), (x'_1,\ldots,x'_d) \in R^{-1}(x')} \prod_{d=1}^{D} k_d(x_d, x'_d)$$

$R^{-1}(x) =$ all decompositions $(x_1, \ldots, x_D)$ such that $(R(x, x_1, x_2, \ldots, x_D)$.
k_{conv} is a PDS kernel as well. Intuitive kernel, used as a building principle for a lot of other kernels. Next, we will see two examples.

Fisher kernel

Combine the advantages of graphical models and discriminative methods
Let $x \in \mathbb{R}^p$ be the input vector of a classifier.

- Learn a generative model $p_\theta(x)$ from unlabeled data $x_1, \ldots, x_n$
- Define the Fisher vector as : $\mathbf{u}_\theta(x) = \nabla_\theta \log p_\theta(x)$
- Estimate the Fisher Information matrix of p_θ:
  $F_\theta = \mathbb{E}_{x \sim p_\theta}[\mathbf{u}_\theta(x)\mathbf{u}_\theta(x)^T]$
- **Definition**: $k_{Fisher}(x, x') = \mathbf{u}_\theta(x)^T F_\theta^{-1} \mathbf{u}_\theta(x)$

Applications
Classification of secondary structure of proteins, topic modeling in documents, image classification and object recognition, audio signal classification . . . Ref: Haussler, 1998. Perronnin et al. 2013.

Kernel Design

- Use closure properties to build new kernels from existing ones
- Kernels can be defined for various objects:
    - **Structured objects**: (sets), graphs, trees, sequences, . . .
    - Unstructured data with underlying structure: texts, images, documents, signal, biological objects
- **Kernel learning**:
    - Hyperparameter learning: see Chapelle et al. 2002
    - Multiple Kernel Learning: given $k_1, \dots, k_m$, learn a convex combination $\sum_i \beta_i k_i$ of kernels (see SimpleMKL Rakotomamonjy et al. 2008, unifying view in Kloft et al. 2010)

Outline

Reminder

Nonlinear SVM and Kernels

- Kernels
- Back to VC-dimension and generalization bounds
- Support Vector Regression

Methodology ctd': ROC curves

References

VC-dimension of canonical hyperplanes

Theorem: **VC-dimension**
Let $\mathcal{S} \subseteq \{\mathbf{x} : \|\mathbf{x}\| \leq r\}$. Then, the VC-dimension d of the set of canonical hyperplanes $\{x \to \mathrm{sgn}(\mathbf{w}^T\mathbf{x}) : min_{\mathbf{x}\in\mathcal{S}}|\mathbf{w}^T\mathbf{x}| = 1 \wedge \|\mathbf{w}\| \leq M\}$ verifies:

$$d \leq r^2 M^2.$$

N.B.: hard margin case.

Proof

Assuming that d is the VC-dimension then there exists $\{\mathbf{x}_1, \ldots, \mathbf{x}_d\}$ a set fully shattered by the canonical hyperplanes. Then, for all $\mathbf{y} = (y_1, \ldots, y_d) \in \{-1, 1\}^d$, there exists a $\mathbf{w}$ such that: $\forall i \in [1, d], 1 \leq y_i(w^T x_i)$ Summing up:

$$d \leq \mathbf{w}^T \sum_{i=1}^{d} y_i \mathbf{x}_i \leq \|w\| \left\| \sum_{i=1}^{d} y_i \mathbf{x}_i \right\| \leq M \left\| \sum_{i=1}^{d} y_i \mathbf{x}_i \right\| .$$

Because this is true for all $y_1 \ldots, y_d$, it also works for the expectation taken over $\mathbf{y} = (y_1, \ldots, y_d)$, the $y_i's$ i.i.d. from a uniform distribution

$$d \leq M \mathbb{E}_{\mathbf{y}}[\left\| \sum_{i=1}^{d} y_i \mathbf{x}_i \right\|]$$

Using Jensen's inequality: $d \leq M \mathbb{E}_{\mathbf{y}}[\left\| \sum_{i=1}^{d} y_i \mathbf{x}_i \right\|^2]]^{\frac{1}{2}}$

Proof ctd'

By linearity of expectation:

$$\mathbb{E}_{\mathbf{y}}[\left\|\sum_{i=1}^{d} y_i \mathbf{x}_i\right\|^2]^{\frac{1}{2}} = (\mathbb{E}_{\mathbf{y}}[(\sum_{i,j=1}^{d} y_i \mathbf{x}_i)^T(\sum_{j=1}^{d} y_j \mathbf{x}_j)])^{\frac{1}{2}} = [\sum_{i,j=1}^{d} \mathbb{E}_{\mathbf{y}}[y_i y_j](\mathbf{x}_i^T \mathbf{x}_j)]^{\frac{1}{2}}$$

By independence property of $y_1, \ldots, y_d$ and because the distribution is uniform, we have:

$$\begin{aligned}\mathbb{E}_{\mathbf{y}}[\left\|\sum_{i=1}^{d} y_i x_i\right\|^2]]^{\frac{1}{2}} &= [\sum_{i\neq j}^{d} \mathbb{E}_{\mathbf{y}}[y_i]\mathbb{E}_{\mathbf{y}}[y_j](\mathbf{x}_i^T \mathbf{x}_j) + \sum_{i=1}^{d} E[y_i^2](\mathbf{x}_i^T \mathbf{x}_i)]^{\frac{1}{2}} \\ &= \sum_{i=1}^{d}(\mathbf{x}_i^T \mathbf{x}_i)]^{\frac{1}{2}} \leq [dr^2]^{\frac{1}{2}} = r\sqrt{d}\end{aligned}$$

Eventually, we have: $d \leq Mr\sqrt{d}$ Therefore $\sqrt{d} \leq rM$.

Generalization Bounds and Optimal Margin Hyperplane

Theorem:
For any $\delta > 0$, with probability at least $1 - \delta$, over a random sampling $\mathcal{S} \sim P^n$, and for canonical hyperplane h defined using $\mathcal{S}$, the following holds:

$$R_P(h) \leq R_{\mathcal{S}}(h) + \sqrt{\frac{2r^2M^2\log(\frac{en}{r^2M^2})}{n}} + \sqrt{\frac{\log(\frac{1}{\delta})}{2n}}$$

(with d replaced by the previous bound).

Generalization bounds and Optimal Margin Hyperplane

From the previous result on VC-dimension of canonical hyperplanes, we have:
controlling the norm **w** allows to control the VC-dimension of canonical hyperplanes and therefore reduces the second term of the bound. OMH (SVM) has been invented to implement SRM principle.

Outline

Reminder

Nonlinear SVM and Kernels

- Kernels
- Back to VC-dimension and generalization bounds
- Support Vector Regression

Methodology ctd': ROC curves

References

Regression from ML point of view

Probabilistic and Statistical Framework 1/2

- Let X be a random vector $\mathcal{X} = \mathbb{R}^p$
- and Y be a continuous random variable$\mathcal{Y} = \mathbb{R}$
- Let $\mathbb{P}$ be the joint probability law of (X,Y)
- Let $\mathcal{S}_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, i.i.d. sample from $\mathbb{P}$.

Regression from ML point of view

Probabilistic and Statistical Framework 2/2

- Let $h : \mathbb{R}^p \rightarrow \mathbb{R} \in \mathcal{H}$, $\mathcal{H}$: some family of functions
- Let $\ell : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ be a local loss function
- Empirical risk : $R_n(h) = \frac{1}{n} \sum_i \ell(y_i, h(x_i))$, Regularizing term: $\Omega(h)$ which measures *complexity* de h.
- We search for : $\hat{h} = \text{arg min}_{h \in \mathcal{H}} R_n(h) + \lambda\Omega(h)$

Regression function

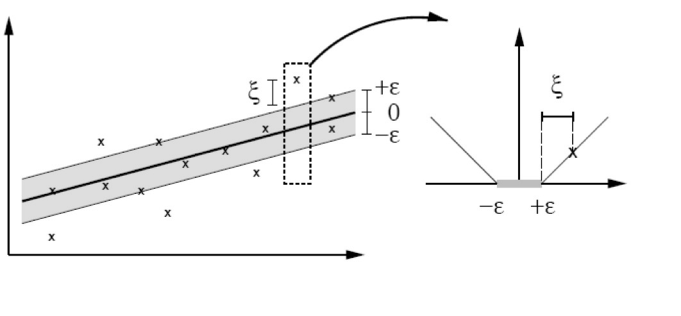
Theorem (Minimal Risk under Squared Error Loss (MSE)
When ℓ is the squared loss: $\ell(y, h(x)) = (y - h(x))^2$, the best solution for the regression problem is the so-called regression function $h^*(x) = \mathbb{E}[Y|x]$. h^* is the function that provides the minimal risk.

Proof:

Let h a predictive model.
Show that $R(h) = \mathbb{E}[(\mathbb{E}[Y|X] - h(X))^2 + R(h^*)$. Then, $R(h) \geq R(h^*)$ for any predictive model h, and therefore, $\min R(h) = R(h^*)$.

Support Vector Regression

- Extend the idea of maximal soft margin to regression
- Impose an ϵ-tube : ϵ-insensitive loss $|y' - y|_\epsilon = max(0, |y' - y| - \epsilon)$



Support Vector Regression

SVR in the primal space
Given C and ϵ

$\min_{w,b,\xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i (\xi_i + \xi_i^*)$

s.c.

$\forall i = 1, \ldots n, y_i - f(x_i) \leq \epsilon + \xi_i$

$\forall i = 1, \ldots n, f(x_i) - y_i \leq \epsilon + \xi_i^*$

$\forall i = 1, \xi_i \geq 0, \xi_i^* \geq 0$

with $f(x) = w^T \phi(x) + b$

General case : ϕ is a feature map associated with a positive definite kernel k.

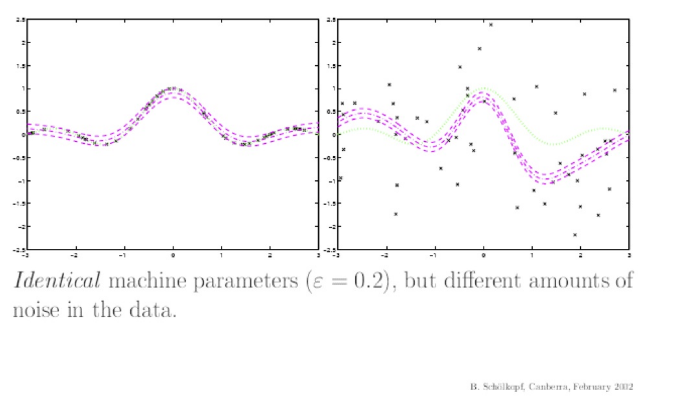
Solution in the dual

$\min_{\alpha,\alpha^*} \sum_{i,j}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)k(x_i, x_j) + \epsilon \sum_i(\alpha_i + \alpha_i^*) - \sum_i y_i(\alpha_i - \alpha_i^*)$
s.c. $\sum_i(\alpha_i - \alpha_i^*) = 0$ and $0 \leq \alpha_i \leq C$ and $0 \leq \alpha_i^* \leq C$
$w = \sum_{i=1}^n(\alpha_i - \alpha_i^*)\phi(x_i)$

Solution

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)k(x_i, x) + b$$

Support Vector Regression: example in 1D



Identical machine parameters ($\varepsilon = 0.2$), but different amounts of noise in the data.

B. Schölkopf, Canberra, February 2002

Outline

Reminder

Nonlinear SVM and Kernels

Methodology ctd': ROC curves

References

Receiver operating characteristic (ROC) curve for Binary Classification

Originally developed in signal detection theory in connection with radio signals, now much used for medical decision-making or any binary decision problem.

ROC curve

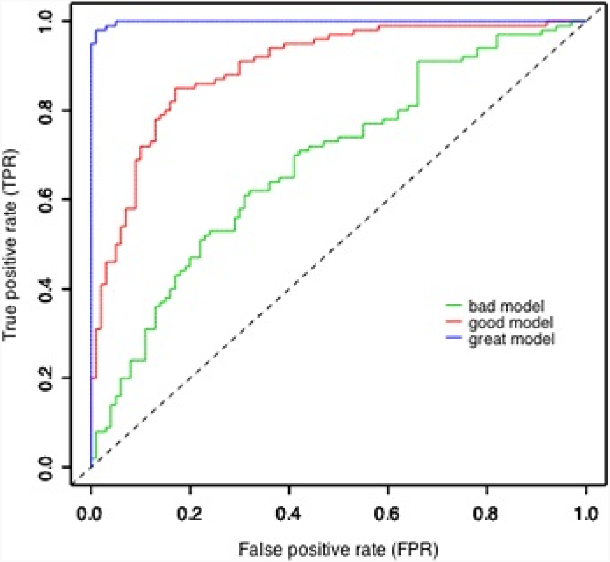
a plot of test sensitivity (True Positive Rate) as the y coordinate versus its 1-specificity or false positive rate (FPR) as the x coordinate. It is an effective method of evaluating the performance of diagnostic tests.

Value table

	Predicted YES	Predicted NO
POS	True Positive	False Negative
NEG	False Positive	True Negative

Let$h(x) = \text{sign}(f(x))$
Building a ROC curve consists in making the threshold s vary and define the point (FNR,TPR). To a single function f, we associate various points in the graph.

ROC Curve Illustration



True Positive Rate (sensitivity) = NB of positive examples correctly classified /NB of positive examples.

Area under the ROC Curve

Let c be a fixed classifier. Assume there is m positive examples and n negative examples in the test set. Let $f_1, \ldots, f_m$ be the outputs of c on the positive examples and $F_1, \ldots, F_n$ its output on the negative examples. The then AUC, A, is defined by:

$$A = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} 1_{f_i > F_j}}{mn}$$

which is the value of the Wilcoxon-Mann-Withney statistics.

Outline

Reminder

Nonlinear SVM and Kernels

Methodology ctd': ROC curves

References

References

- BOSER, Bernhard E., Isabelle M. GUYON, and Vladimir N. VAPNIK, 1992. A training algorithm for optimal margin classifiers. In: COLT 92: Proceedings of the Fifth Annual Workshop on Computational Learning Theory. New York, NY, USA: ACM Press, pp. 144-152.
- CORTES, Corinna, and Vladimir VAPNIK, 1995. Support-vector networks. Machine Learning, 20(3), 273-297.
- Article really cool (a bit of maths, preparation to M2) : A tutorial review of RKHS methods in Machine Learning, Hofman , Schoelkopf, Smola, 2005 (`https://www.researchgate.net/publication/228827159_A_Tutorial_Review_of_RKHS_Methods_in_Machine_Learning`)
-