B. Schoelkopf, 14 April 2003, Berlin

www.kyb.tuebingen.mpg.de/~bs

72076 Tübingen, Germany

Max Planck Institute for Biological Cybernetics

Bernhard Schölkopf

Introduction to Kernel Methods
Acknowledgement
binary pattern recognition

\[ |h - (x)f|_\mathcal{Z} = (z', (x)f)\{1\oplus\} = \mathcal{C} \]

Special case: true output is \( h \).
Here, \((x)f \) is the loss incurred when predicting \((h', (x)f)\) is minimized.

\[ (h', x)p (h', (x)f) \int x = [f]h \]

We want to estimate a function such that \( \mathcal{C} \leftarrow x : f \) where

\[ (h', x)p \sim (h', x)p \]

\( \mathcal{C} \times x \in (m_h, m_x), \cdots (m_h, x) \)

Suppose we are given data

Learning Problem
How about problems that are not linearly separable?

Decision function: hyperplane with normal vector $m$. $m^T x + c = 0$.

Idea: classify points $x$ according to which of the two classes means is closer.

**An Example of a Pattern Recognition Algorithm**
Kernel Feature Spaces

Preprocess the inputs with

\[ \Phi : \mathcal{X} \rightarrow \mathcal{H} \]

where \( \mathcal{H} \) is a dot product space, and learn the mapping from \( \Phi(x) \) to \( y \).
Example: All Degree 2 Monomials
Dimension $10_{10}$

$\exists \ g \ N = 16 \times 16$, and $p = 5$

$\dim(\mathcal{H})$ grows like $N^p$.

How about patterns $x \in \mathbb{R}^N$ and product features of order $p$.
in \( \mathbb{R}^2 \), the dot product in can be computed from the dot product

\[
(x', x)_\mathcal{H} = \langle x', x \rangle = (\mathcal{Z}(x, x) + \mathcal{I}(x, x)) = \perp \mathcal{Z}(x', x, x) \mathcal{Z}(\mathcal{Z}(x' x', x), x') = \langle (x)\Phi, (x)\Phi \rangle
\]

The Kernel Trick
\( \langle (x)\Phi, (x)\Phi \rangle = \sum_{i=p}^{\ell N} (x^i \cdot (x^i \sum_{i=p}^{\ell N} = p \langle x, x \rangle) \)

More Generally: for \( x, \in \mathbb{R} \), \( p \in \mathbb{N} \),

in the dot product in \( \mathbb{R}^2 \), the dot product can be computed from the dot product

\( (x, x) : = \sum \langle x, x \rangle = \sum (x ; x + x \cdot x) = \sum \langle x, x \rangle (x ; x + x \cdot x) = \langle (x)\Phi, (x)\Phi \rangle \)

\( \mathbb{R} = p = \mathbb{N} \)

The Kernel Trick.
Special case of positive definite kernels: “Mercer kernels”

\( \langle (x) \phi, (x) \phi \rangle = (x, x) \mathcal{K} \)

there exists a map \( \phi \) into a dot product space such that \( \mathcal{H} \) is a so-called reproducing kernel Hilbert space.

\[
\mathcal{K}(x, x) \geq 0 \\
\text{for all } x, x' \in \mathcal{X} \text{ and any } a_1, \ldots, a_m \in \mathbb{R}
\]

\( \mathcal{K} \) is positive definite (pd), i.e., \( \mathcal{K} \) is symmetric, and for \( \mathcal{X} \) be a nonempty set. The following two are equivalent:

Positive Definite Kernels
Kernels are studied also in approximation theory (Micheli, 1986).

\[
\begin{align*}
(\|x - x\|_2^2) \exp(-\|x - x\|_2^2)/2 & = (x, x) \\
p(c + \langle x, x \rangle) & = (x, x)
\end{align*}
\]

Examples of common kernels:

- think of the kernel as a (nonlinear) similarity measure
- need not be a vector space
- from the kernel trick
- any algorithm that only depends on dot products can benefit from the kernel trick

The Kernel Trick — Summary
Compute the sign of the dot product between \( w \) and \( x - c \).

The two class means is closer.

Classify points according to which of

\[ \langle x \rangle \Phi = x \]

An Example of a Kernel Algorithm
if \( \mathbf{z} \) is a density: Parzen windows interpretation

\[
\begin{pmatrix}
(\mathcal{L}(\mathbf{x}, \mathbf{x})y_{\mathcal{I}} \sum_{\mathcal{I}} \frac{+w}{z} - (\mathcal{L}(\mathbf{x}, \mathbf{x})y_{\mathcal{I}} \sum_{\mathcal{I}} \frac{-w}{z}) \mathbf{z} = q
\end{pmatrix}
\]

with the constant offset

\[
\begin{pmatrix}
q + (\mathcal{L}(\mathbf{x}, \mathbf{x})y_{\mathcal{I}} \sum_{\mathcal{I}} \frac{-w}{z} - (\mathcal{L}(\mathbf{x}, \mathbf{x})y_{\mathcal{I}} \sum_{\mathcal{I}} \frac{+w}{z}) \mathbf{z} =
\end{pmatrix}
\]

An Example of a Kernel Algorithm, etc.
Demo

unique solution found by convex OP

\[ (q + (x, x)^{T}y) \chi \sum \sin = (x)f \]

spars expansion of solution in terms of SVD

large margin separation

Support Vector Classifiers
\[
\begin{align*}
\sum_{i=1}^{4} \lambda_i \cdot k(x, x_i) &= \text{classification weights} \\
\sum_{i=1}^{4} \lambda_i \cdot k(x, x_i) + b &= \text{classification}
\end{align*}
\]
Examples, $28 \times 28$

Handwritten character benchmark (60000 training & 10000 test

**MNIST Benchmark**
<table>
<thead>
<tr>
<th>Translation Invariant SVM</th>
<th>Boosted LENET4</th>
<th>LENET4</th>
<th>Tangent Distance</th>
<th>SVM</th>
<th>3-nearest-neighbor</th>
<th>Linear classifier</th>
<th>Reference Test Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decoste and Scholkopf (2002)</td>
<td>0.56%</td>
<td>0.7%</td>
<td>1.1%</td>
<td>1.1%</td>
<td>1.4%</td>
<td>2.4%</td>
<td>8.4%</td>
</tr>
</tbody>
</table>
Note: the SVM system that holds the record on the MNIST set used a polynomial kernel of degree 9, corresponding to a feature space of dimensionality $\approx 3.2 \cdot 10^{20}$.

“Curse of Dimensionality”? Statistical Learning Theory: there is a curse of capacity, not of dimensionality.
\[ \mathcal{R}_{\text{emp}} \left( \hat{f} - (\hat{x})f \right) = \sum_{m} \frac{1}{m} = \left[ f \right]_{\mathcal{H}} \]

Training Error

Empirical risk minimization (ERM):


\[ \left( \hat{f}, x \right) \mathcal{P} \left( \hat{f} - (x)f \right) \int_{\mathcal{H}} = \left[ f \right]_{\mathcal{H}} \]

Drawn from \( \mathcal{P} \) drawn from \( \mathcal{P} \) drawn from \( \mathcal{P} \)

such that the expected classification error on a test set, also each pair generated from \( \mathcal{P} \), \( \hat{f} \), \( \mathcal{P} \), \( \mathcal{P} \)

Learn \( \mathcal{P} \) from \( \left\{ I \mathbb{P} \right\} \) from examples

Pattern Recognition
for all $\epsilon > 0$.

$$0 = \{ \epsilon < \sup_\mathcal{H} \left( \frac{1}{n} \sum_{i=1}^{n} [f]^\mathcal{H} - [f]_\mathcal{H} \right) \}_{\mathbb{P}\in\mathcal{H}} \lim_{n\to\infty}$$

sistent if and only if the convergence is uniform.

Vapnik and Chervonenkis showed that ERM is (nonsensically) cons-


tinuous of empirical risk mini-

mum.

Does this imply that ERM will give us the optimal result in the

law of large numbers: for every

\( f \in \mathcal{F} \exists \epsilon \in \mathcal{H} \).

Convergence of Means to Expectations
Risk Function class

\[ R \{ f \} = \mathcal{F} \]

Function class

Consistency and Uniform Convergence
To have a low test error, we need a low training error and low capacity.

(Williamson et al., 1998; Alon et al., 1997;

\( e.g. \) Vapnik and Chervonenkis, 1974; Vapnik, 1998;

Shawe-Taylor et al., 1998;

(e.g.) the entropy numbers are well-behaved

\( \bullet \) the VC-dimension is finite

\( \bullet \) the VC-entropy grows sublinearly with \( m \)

\( \bullet \) the VC-entropy of capacity concepts of the function class, e.g.

Vapnik, Chervonenkis and others give conditions for uniform con-

\[ \text{Capacity} \]
Justifications for Large Margins

\[ \text{Compression} / \text{MDL} \ (\text{vonLuxburg et al., 2002}) \]
\[ \text{Rademacher Averages} \ (\text{Koltchinskii et al., 2001; Mendelson, 2001; Bousquet, 2002}) \]
\[ \text{Algorithmic Stability} \ (\text{Bousquet and Elisseeff, 2002}) \]
\[ \text{Bayesian MAP Estimation} \ (\text{Kimeldorf and Wahba, 1970; Poggio and Girosi, 1990}) \]
\[ \text{Regularization Theory} \ (\text{Caponi, 1998; Smola and Schölkopf, 1998}) \]
\[ \text{Shawe-Taylor et al., 1998}) \]
\[ \text{Fat-Shattering Dimension and Data Dependent SVM} \ (\text{Cunlus, 1997}) \]
\[ \text{radius of the smallest sphere containing the data} \ (\text{Vapnik, 1999}) \]
\[ \text{V-C-Dimension:} \ y \leq \frac{R^2}{\lambda} \]

Where \( \lambda \) is the margin and \( R \) is the

\[ \frac{m}{(\ln g)^2} \geq 2 \log(2) R g \]

Theorem. For all \( f \in \mathcal{F} \) with probability \( \geq 1 - \varepsilon \),

\[ (w x) f, \ldots, (I x) f \]

\[ \leftarrow \]

\[ w, \ldots, w \]  
\[ x, \ldots, x \]  

Sender

Receiver

Given the training inputs, using functions from \( \mathcal{F} \), the compression coefficient of the training labels.

Denote by \( \mathcal{C} \) the compression coefficient of the training labels.

Given a finite function class \( \mathcal{F} \).

---

Compression Bound
\{ \text{hyperplanes with } \| \mathbf{n} \| = 1 \}\}

\mathcal{L} \text{ of } \land \triangledown \text{-cover can be done by computing a }\land \triangledown \text{-cover by von Luxburg et al. 2002.}

Hence only need to transmit with accuracy \( \frac{\rho}{\sigma} \) and still correctly separate the data. Can perturb \( \land \triangledown \text{ by } \land \triangledown \text{ with } \arcsin > |\land \triangledown| \text{ and still correctly separate.}

Maximum Margin vs. MID — 2D Case
axes of ellipsoid can be computed from kernel eigenvalues.

Ellipsoid setting: different directions imply different...
Abalone (m = 500)
Wisconsin breast cancer (m = 200)
Datasets: USPS (m = 500)

Experiments: Selecting $\varphi$ in a Gaussian kernel
... ●
feature extraction ●
quantitative estimation / novelty detection ●
classification ●

2. "Learning module"

\[
\((\mathbb{I} x, \Phi) = (x) f\]

"representer theorem, function class" ●

thus can construct geometric algorithms —

\[
\langle (x) \Phi, (x) \Phi \rangle = \langle x, x \rangle \Phi \text{ in associated feature space where } \Phi \text{ data representation}
\]

similarity measure \( \kappa(x, x) \) where \( x \in X \) ●

I. "Kernel module"

Further Kernel Algorithms — Design Principles
$p(\lambda \cdot x) = (\lambda, x)\gamma$

Linear PCA

$\Phi$ kernel PCA

$\gamma$

$\mathbb{R}^2$

$(\lambda \cdot x) = (\lambda, x)\gamma$

Demo

(Schoelkopf et al., 1998)
Information retrieval

• Outlier detection (Schölkopf et al., 2001)

• Network intrusion detection (Wang and Cai, 2001)

• Jet engine condition monitoring (Hajton et al., 2001)

\[
((\frac{2}{\pi \| \mathbf{h} - \mathbf{x} \|^2 \cdot \exp}) = (\mathbf{h} \cdot \mathbf{x})^y \text{ using } y)
\]

| $\sigma$ | 0.65, 0.03 | 0.24, 0.03 | 0.59, 0.47 | 0.54, 0.43 | SYS/OLS
|---|---|---|---|---|---|
| $\epsilon$ | 0.1, 0.1 | 0.1, 0.1 | 0.5, 0.5 | 0.5, 0.5 | Width $\epsilon$

One-class SVM
A convenient way of learning the \( \omega_i \) is to work in the kernel PCA basis.

\[
\|(\Phi_i) \Phi - ((x) \Phi) \Omega \|_\mathcal{X} \text{ar}_{\text{min}} = \omega_i
\]

compute the pre-image

This can be evaluated in various ways. E.g., given an \( x \), we can

\[
\langle \cdot, (\Phi_i \Phi) \rangle_{\mathcal{X}} \text{Conv} = (\cdot) \Omega
\]

Estimate a dependence \( \mathcal{H} \leftarrow \mathcal{H} : \Omega \text{ and training data}\)

\[
(\omega_i, x) \quad (\Phi_i \Phi) \quad (\Phi_i \Phi) \quad \omega_i \quad \mathcal{H}
\]

Given two sets \( X \) and \( Y \), and kernels \( \mathcal{K} \) and \( \mathcal{K} \), and training data

\[
\text{Kernell Dependency Estimation (Weston et al., 2002)}
\]
(from Weston et al. (2002))

a mistake (73 mistakes for k-NN, 23 for KDE).

Shown are all digits where at least one of the two algorithms makes

Application to Image Completion
- model selection, e.g., via alignment (Christmann et al., 2001)
- functional calculus for kernel matrices (Scholkopf et al., 2002)
- global kernels from local ones (Rydoner and Lafferty, 2002)
- complex kernels from simple ones (Haussler, 1999; Barrall and Scholkopf, 2002)
- local kernels (e.g., Zien et al., 2000)
- kernels based on generative models (Jacobova and Haussler, 1999; Seeger, 1999)
- kernal design
- \ldots
- theory of empirical inference: sharper capacity measures and bounds (Barrall, Bouv.

\ldots
- optimization and implementation: OP, SDP (Langer et al., 2002), online ver-
- Hammerling et al., 2002), canonical correlations (Bach and Jordan, 2002, Kuss, 2002)
- \ldots
- algorithms/tasks: KDE, feature selection (Weston et al., 2001), multi-label-problems

\underline{Kernell Machines Research}
Conclusion

- crucial ingredients of SV algorithms: kernels that can be represented as dot products, and large margin regularizers
- kernels allow the formulation of a multitude of geometrical algorithms (Parzen windows, SV pattern recognition, SV quantile estimation, kernel PCA, ... ) that work very well in practice
- kernels unify three aspects of empirical inference: similarity measures, function classes, and data representations. The choice of a kernel is crucial, and it is not a problem of statistics.

References


Introduction. A new approach to pattern recognition, approximation, and operator

R.

= N

M

' %

G

D

D

D

m

I

1998.

B. Schölkopf and A. J. Smola, eds., Kernel Methods for Pattern Recognition, Approximation and Operator


S. Mézières. Backward average and phase transitions in Chirn-Cardell classes. IEE Transactions on Information

1998.
