

Building classifiers in high dimensional space

Classifiers define decision surfaces in some feature space where the data is either initially represented or mapped to

Representing or mapping the data in a high dimensional space may ease the separability between the classes (see Cover's theorem)

but...

discrimination is not easier if the mapped points naturally lie into a lower dimensional manifold

the higher the dimension of the feature space the more parameters *may* need to be estimated

The blessing of dimensionality for kernel methods

Pierre Dupont

Pierre.Dupont@uclouvain.be

– Typeset by FoilT_EX –

Outline

- The curse of dimensionality
- The 3 core ideas of kernel methods, and specifically SVMs
- How to avoid the curse of dimensionality?
 - ⇒ some results from Vapnik's Statistical Learning Theory
- Why SVMs are interesting techniques but not the panacea?
- Kernels and regularized risk

The curse of dimensionality

If the number of parameters is too large with respect to the number of training samples there is a risk of **over-fitting** the training data

Over-fitting implies **poor generalization** to correctly classify *new* data

Additionally, sensitivity to noise and computational complexity *may* increase with the dimension of the feature space

This problem is known as the **curse of dimensionality**

However...

The 3 core ideas of kernel methods

- The so-called **kernel trick** allows to define an **implicit mapping** to a higher dimensional feature space with two interesting consequences
 - there is no need to compute anything in the higher dimensional space
 - the number of parameters to be estimated becomes **independent** of the dimension of the feature space
- The **capacity** of the class of discriminant functions considered matters more than the dimension of the space they lie into
Capacity is a measure of the complexity of a class of functions
The best known capacity concept is the Vapnik-Chervonenkis (VC) dimension
- Controlling the capacity of linear discriminants can be done by **maximizing the margin** of the hyperplane with respect to the training samples

Kernels + capacity control through margin maximization
 \Rightarrow the **blessing of dimensionality**

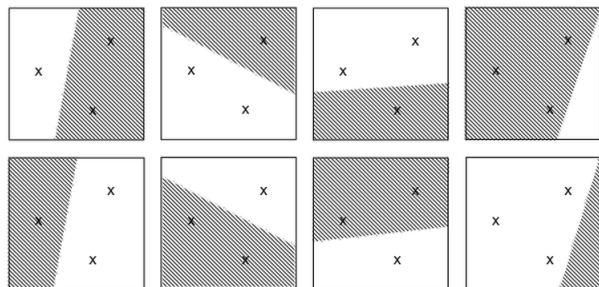
4

Shattering

A set of samples S is **shattered** by a function class \mathcal{F} if and only if for every possible +/- labeling of the samples in S there exists some function in \mathcal{F} which perfectly classify the samples

For instance, if the function class \mathcal{F} is the set of hyperplanes in \mathbb{R}^2 (= lines) and if we consider 3 samples, there are $2^3 = 8$ possible labellings

For any such labeling there is a hyperplane classifying correctly the samples



5

VC dimension

The **Vapnik-Chervonenkis dimension (VC-dim)** of a function class \mathcal{F} defined over an instance space X is the size of the largest subset of X shattered by \mathcal{F} . If arbitrarily large finite sets of X can be shattered by \mathcal{F} then $VC(\mathcal{F}) \equiv \infty$

We have seen a set of 3 points in \mathbb{R}^2 which can be shattered by hyperplanes even though they are sets of 3 points which cannot be shattered



No sets of 4 points can be shattered by a hyperplane in \mathbb{R}^2 , no matter how they are placed

\Rightarrow the VC-dim of hyperplanes in \mathbb{R}^2 is 3

More generally, the VC-dim of hyperplanes in \mathbb{R}^d is $d + 1$

6

Empirical Risk

Let $g(\mathbf{x})$ be a discriminant for a binary classification problem $\Omega = \{\omega_1, \omega_2\}$
 The decision function $f : X \rightarrow \{1, -1\}$ is defined as $f = \text{sign}(g(\mathbf{x}))$

Let $\mathbf{x}_1, \dots, \mathbf{x}_n$ be a training set of n samples with associated labels z_1, \dots, z_n
 By definition $z_i = 1$ if \mathbf{x}_i is labeled ω_1 , and $z_i = -1$ otherwise

The **zero-one loss function** $\frac{1}{2}|f(\mathbf{x}) - z|$ defines the correctness of the classification of any sample \mathbf{x} . The loss is 0 if \mathbf{x} is correctly classified, and 1 otherwise

The **average training error** or **empirical risk** is defined as

$$R_{emp}[f] = \frac{1}{n} \sum_{i=1}^n \frac{1}{2} |f(\mathbf{x}_i) - z_i|$$

7

Bounding the Risk

The true **risk** or probability of misclassification for any test sample drawn from $P(\mathbf{x}, z)$ the (unknown) joint distribution of samples and class labels is defined as

$$R[f] = \int \frac{1}{2} |f(\mathbf{x}) - z| dP(\mathbf{x}, z)$$

Over-fitting occurs when a function f minimizing the empirical risk $R_{emp}[f]$ does not minimize the true risk

Fortunately, we can bound the risk if we know the VC-dim h of the function class \mathcal{F} which f belongs to

In particular, if $h < n$ (the number of training samples), then for all functions of \mathcal{F} , **independently** of the underlying distribution P , with probability at least $1 - \delta$

$$R[f] \leq R_{emp}[f] + \sqrt{\frac{1}{n} \left(h \left(\ln \frac{2n}{h} + 1 \right) + \ln \frac{4}{\delta} \right)}$$

8

Interpretation of the VC-bound

$$R[f] \leq R_{emp}[f] + \underbrace{\sqrt{\frac{1}{n} \left(h \left(\ln \frac{2n}{h} + 1 \right) + \ln \frac{4}{\delta} \right)}}_{\text{capacity or confidence term}}$$

The results holds only with probability (at least) $1 - \delta$ because the test data may be particularly difficult

When the training set size $n \rightarrow \infty$ the capacity term $\rightarrow 0$ and $R[f] \rightarrow R_{emp}[f]$

Considering a function class \mathcal{F} with low VC-dim h reduces the capacity term

However if the function class is too simple (too low VC-dim) it will be difficult to minimize $R_{emp}[f]$

This property can be seen as another formulation of the classical bias-variance trade-off

\Rightarrow there is an optimum to be found

9

Practical use of the VC-bound?

$$R[f] \leq R_{emp}[f] + \sqrt{\frac{1}{n} \left(h \left(\ln \frac{2n}{h} + 1 \right) + \ln \frac{4}{\delta} \right)}$$

The above bound is **not** tight because it derives from a worst-case analysis and must hold for any distribution $P(\mathbf{x}, z)$

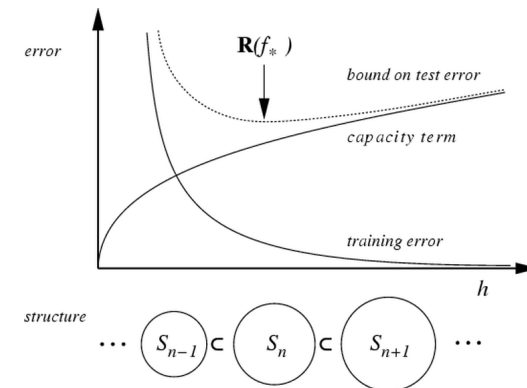
In practice, the distribution $P(\mathbf{x}, z)$ is unknown but not arbitrary

The interest of this bound is not so much its practical use but it motivates to best fit the data with the simplest possible class of functions

10

Structural risk minimization

Minimizing both $R_{emp}[f]$ and the capacity term by choosing the class of functions suitable for the amount of training data is the core of **structural risk minimization**



There is no curse of dimensionality but there is a **curse of capacity**

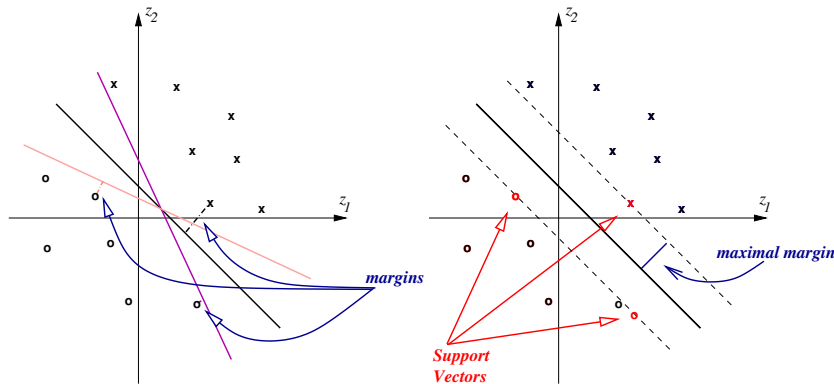
11

Support Vectors and Maximal Margin Hyperplane

When the data is linearly separable in some appropriate feature space, the separating hyperplane is not unique

The **maximal margin hyperplane** separates the data with the largest margin

For each separating hyperplane, there is an associated set of **support vectors**



Maximizing the margin is a good idea

Recall that the VC-dim of hyperplanes in \mathbb{R}^d is $d + 1$
 \Rightarrow the capacity term increases with the dimension of the space

Hopefully, for hyperplanes with margin ρ it was shown that the VC-dim h is bounded

$$h \leq \frac{R^2}{\rho^2} + 1$$

where R is the radius of the smallest hypersphere containing the data

The key advantage of this bound is that it is **independent** of the dimension d !!!

Maximizing the margin is a way to control the curse of capacity while working in very high dimensional spaces

Maximizing the margin is also a way to increase robustness to noise since perturbations around the training points do not affect much the decision boundary

Discussion

- The above property defines the VC-dim of **canonical hyperplanes** relative to a dataset (not all hyperplanes in \mathbb{R}^d)
- The maximal margin ρ needs to be defined a priori (not strictly equivalent to the SVM optimization problem)
- A similar and more general result holds for another capacity concept: the **fat shattering dimension**
- **Overfitting** is still possible depending on the **kernel choice** (see later...)

Mercer kernels

A kernel k is a symmetric function with $k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle = k(\mathbf{x}', \mathbf{x})$ where ϕ is a mapping from the original input space X to a feature space Y

Mercer Conditions: A symmetric function $k : X \times X \rightarrow \mathbb{R}$ is a kernel if for any finite subset $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ of X the **gram matrix**

$$\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)]_{i,j=1}^n$$

is positive semi-definite (has non-negative eigenvalues)

$k(\mathbf{x}, \mathbf{x}')$ can be thought of as a similarity measure between \mathbf{x} and \mathbf{x}' which generalizes the simple dot product $\langle \mathbf{x}, \mathbf{x}' \rangle$

Implicit mapping induced by a kernel

If k satisfies the Mercer conditions, there exists a mapping ϕ such that

$$k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$$

We can directly specify k rather than ϕ

⇒ there is an **implicit mapping** to a new feature space

Linear kernel $k(\mathbf{x}, \mathbf{x}') = \langle \mathbf{x}, \mathbf{x}' \rangle$ (ϕ maps \mathbf{x} to itself)

Polynomial kernel $k(\mathbf{x}, \mathbf{x}') = (\langle \mathbf{x}, \mathbf{x}' \rangle + c)^b$ with $b \in \mathbb{N}, c \geq 0$

Gaussian Radial Basis Function kernel $k(\mathbf{x}, \mathbf{x}') = \exp\left(\frac{-\|\mathbf{x} - \mathbf{x}'\|^2}{\sigma^2}\right)$ with $\sigma \neq 0$

Sigmoid kernel $k(\mathbf{x}, \mathbf{x}') = \tanh(\kappa \langle \mathbf{x}, \mathbf{x}' \rangle + \vartheta)$ with $\kappa > 0$ and $\vartheta < 0$

The kernel trick

Any learning algorithm that uses the data only via dot products can rely on this implicit mapping by replacing $\langle \mathbf{x}, \mathbf{x}' \rangle$ by $k(\mathbf{x}, \mathbf{x}')$

16

Hard margin SVMs

The SVM estimation problem (i.e. finding a maximal margin hyperplane in the feature space) may be formulated (in its dual form) as

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j z_i z_j k(\mathbf{x}_i, \mathbf{x}_j)$$

The number of parameters only depends on the number of training samples n , not the dimension of the input or the feature space

The decision function is defined as:

$$f(\mathbf{x}) = \text{sign} \left[\sum_{\mathbf{x}_i \in SV} \alpha_i z_i k(\mathbf{x}_i, \mathbf{x}) + w_0 \right]$$

which only depends on the (so-called support) vectors \mathbf{x}_i such that $\alpha_i \neq 0$

17

SVMs pro's

- SVMs are "theoretically motivated" by Vapnik's statistical learning theory
- estimation is a convex optimization problem (no multiple local minima)
- primal-dual formulation and the duality gap to measure "distance" to optimum
- sparse solution: only the support vectors matter in the decision function
- state of the art results on many different datasets
- the kernel trick allows to build classifiers for structured data such as strings, trees, graphs, probability distributions, etc
- relatively few meta-parameters: C (soft margin formulation), σ (RBF kernel), the kernel itself, ...

18

SVMs are interesting but not the panacea

- many aspects are not new:
 - The kernel trick is nearly a century old (Mercer 1909) but it was used only much later to build a classifier (Boser, Guyon and Vapnik; COLT'92)
 - The Ho and Kashyap algorithm (1965) estimates a hyperplane with a large margin (with a minimum-squared error criterion and without the kernel trick)
 - SVMs with RBF kernels are close to RBF networks (identical decision functions, different estimation procedures: k-means vs prototypes selected as support vectors)
- Computational cost of the training procedure becomes prohibitive for very large datasets (but chunking can help)
- The kernel choice is critical

This is a practical concern but also a theoretical issue:

The VC-bound applies in the (implicit) feature space !!!

19

Over-fitting induced by the kernel

Consider a RBF kernel $k(\mathbf{x}, \mathbf{x}') = \exp\left(\frac{-\|\mathbf{x}-\mathbf{x}'\|^2}{\sigma^2}\right)$ with $\sigma \rightarrow 0$

The gram matrix $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)]_{i,j=1}^n$ tends to $c\mathbf{I}$

In other words, training points are only considered (very) similar to themselves
 \Rightarrow fitting the training set is easy but generalization is likely to be poor

The "ideal" kernel is such that any pair of points $(\mathbf{x}, \mathbf{x}')$ are considered similar if and only if they should be associated to the same class label z

\Rightarrow the design of this kernel would require the knowledge of $P(\mathbf{x}, z)$ to minimize the true risk $R[f]$

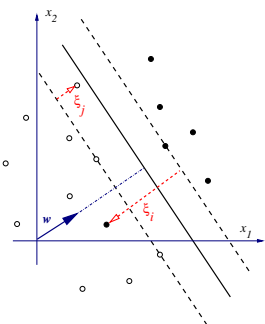
Regularized risk

True risk minimization can be approximated by minimizing a regularized risk

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

where $\Omega[f]$ penalizes the lack of smoothness of the function f and λ is a regularization constant

Maximizing the margin of classification by a hyperplane in feature space is equivalent to minimizing $\Omega[f] = \frac{1}{2}\|\mathbf{w}\|^2$



This setting corresponds to soft-margin SVMs with $R_{emp}[f]$ approximated by a function of the slack variables ξ_i

$$\min_{\mathbf{w}, \xi} \underbrace{\frac{1}{2}\|\mathbf{w}\|^2}_{\text{margin maximization}} + \frac{C}{n} \underbrace{\sum_{i=1}^n \xi_i}_{\text{margin error}}$$

Kernel choice is a regularization choice

Representer theorem (see [Schölkopf and Smola, 2002], chap. 4)

Let \mathcal{H} denote the feature space associated to a kernel k and $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ be a labeled data set

Each minimizer $f \in \mathcal{H}$ of the regularized risk $R_{emp}[f] + \lambda\Omega[f]$ admits a representation of the form:

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

In other words, the kernel choice is a regularization choice

The RBF kernel can be shown to penalize derivatives of all orders, and thus enforce more or less smoothness depending on σ

Take home message

- the curse of capacity matters more than the curse of dimensionality
- maximizing the margin is a good idea to control the capacity of the function class considered and to build classifiers robust to noise
- there is no free lunch in the kernel choice but each kernel corresponds to a regularization operator

References

- [Boser et al., 1992] Boser, B., Guyon, I., and Vapnik, V. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pages 144–152, Pittsburgh, PA, USA.
- [Cristianini and Shawe-Taylor, 2000] Cristianini, N. and Shawe-Taylor, J. (2000). *Support Vector Machines and other kernel-based learning methods*. Cambridge University Press.
- [Ho and Kashyap, 1965] Ho, Y.-C. and Kashyap, R. (1965). An algorithm for linear inequalities and its applications. *IEEE Transactions on Electronic Computers*, EC 14:683–688.
- [Mercer, 1909] Mercer, J. (1909). Functions of positive and negative type and their connection to the theory of integral equations. *Philosophical Transactions of The Royal Society London, A* 209:415–446.

24

- [Schölkopf and Smola, 2002] Schölkopf, B. and Smola, A. (2002). *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, Cambridge, MA.
- [Shawe-Taylor and Cristianini, 2004] Shawe-Taylor, J. and Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*. Cambridge University Press.
- [Vapnik, 2000] Vapnik, V. (2000). *The Nature of Statistical Learning Theory*. Springer, 2nd edition.

25