Support Vector Machines

Léon Bottou

COS 424 – 4/1/2010

Agenda

Goals	Classification, clustering, regression, other.
Representation	Parametric vs. kernels vs. nonparametric Probabilistic vs. nonprobabilistic Linear vs. nonlinear Deep vs. shallow
Capacity Control	Explicit: architecture, feature selection Explicit: regularization, priors Implicit: approximate optimization Implicit: bayesian averaging, ensembles
Operational Considerations	Loss functions Budget constraints Online vs. offline
Computational Considerations	Exact algorithms for small datasets. Stochastic algorithms for big datasets. Parallel algorithms.

Summary

1. Maximizing margins.

2. Soft margins.

3. Kernels.

4. Kernels everywhere.

The curse of dimensionality

Polynomial classifiers in dimension d

Discriminant function: $f(\mathbf{x}) = \mathbf{w}^\top \Phi(\mathbf{x}) + b$.

Degree	Dim($\Phi(\mathbf{x})$)	$\Phi(\mathbf{x})$
1	d	$\Phi(\mathbf{x}) = [x_i]_{\ 1 \le i \le d}$
2	$\approx d^2/2$	$\Phi(\mathbf{x}) \mathrel{+}= [x_i x_j]_{\ 1 \le i \le j \le d}$
3	$\approx d^3/6$	$\Phi(\mathbf{x}) \mathrel{+}= [x_i x_j x_k]_{\ 1 \le i \le j \le k \le d}$
...		
n	$\approx d^n/n!$	

The number of parameters increases quickly.

Training such a classifier directly requires a number of examples that increases just as quickly as the number of parameters.

Beating the curse of dimensionality?

Capacity $\ll$ number of parameters

Assume the patterns $\mathbf{x}_1 \ldots \mathbf{x}_{2l}$ are known beforehand.
The classes are unknown.
Let $R = \max \|\mathbf{x}_i\|$.

We say that a hyperplane

$$\mathbf{w}^\top \mathbf{x} + b \quad \mathbf{w}, \mathbf{x} \in \mathbb{R}^d \quad \|\mathbf{w}\| = 1$$

separates patterns with margin Δ if

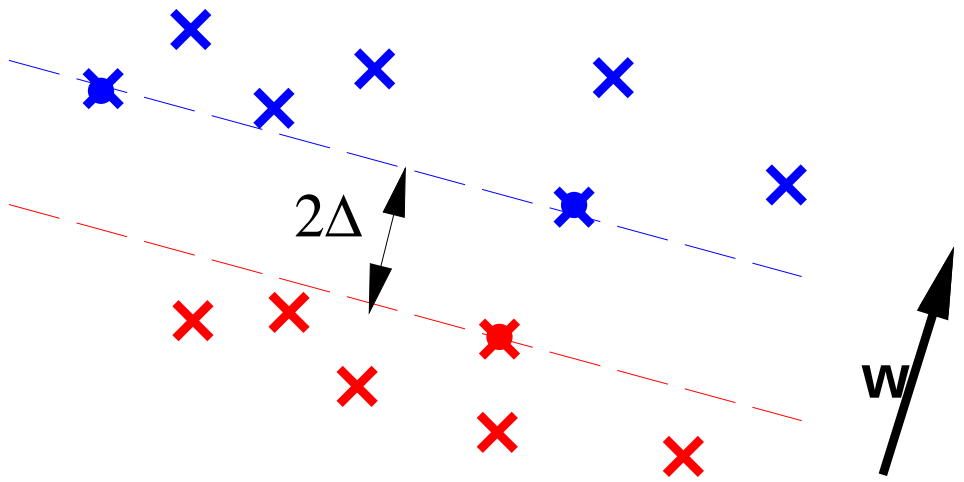
$$\forall i = 1 \ldots 2l \quad |\mathbf{w}^\top \mathbf{x}_i + b| \geq \Delta$$

The family of Δ-margin separating hyperplanes has

$$\log \mathcal{N}(\mathcal{F}, \mathcal{D}) \leq h \log \frac{2le}{h} \quad \text{with} \quad h \leq \min \left\{ \frac{R^2}{\Delta^2}, d \right\} + 1$$

Maximizing margins

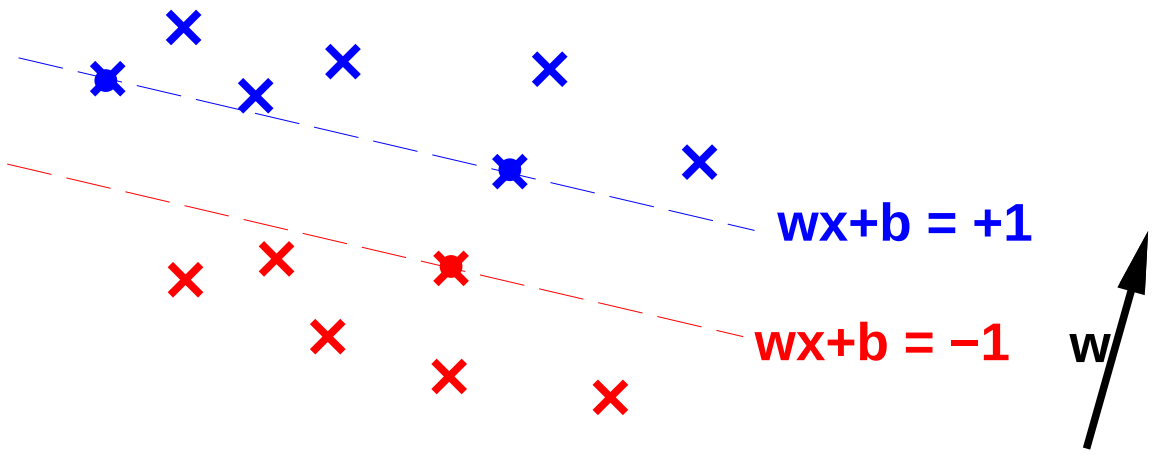
Patterns $\mathbf{x_i} \in \mathbb{R}^d$, classes $y_i = \pm 1$.



$$\max_{\mathbf{w},b,\Delta} \Delta \quad \text{subject to} \quad \|\mathbf{w}\| = 1 \quad \text{and} \quad \forall i \quad y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq \Delta$$

Maximizing margins

Classic formulation



$$\min_{\mathbf{w},b} \|\mathbf{w}\|^2 \quad \text{subject to} \quad \forall i \quad y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1$$

This is a quadratic programming problem with linear constraints.

Maximizing margins

Equivalence between the formulations

Let $\mathbf{w}' = \dfrac{\mathbf{w}}{\Delta}$ and $b' = \dfrac{b}{\Delta}$.

Constraint $y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq \Delta$ becomes $y_i(\mathbf{w'}^\top \mathbf{x}_i + b') \geq 1$.

Problem $\max_{\mathbf{w},b,\Delta} \Delta$ subject to $\|\mathbf{w}\| = 1$ becomes $\min_{\mathbf{w}',b'} \|\mathbf{w}'\|$

Both discriminant functions $\mathbf{w}^\top \mathbf{x} + b$ and $\mathbf{w'}^\top \mathbf{x} + b'$
describe the same decision boundary.

Primal and dual formulation

Karush-Kuhn-Tucker theory

– Refined theory for convex otimization under constraints.

– Construct a *dual optimization problem*
whose constraints are simpler,
and whose solution is related to the solution we seek.

Primal and dual formulation

Karush-Kuhn-Tucker theory

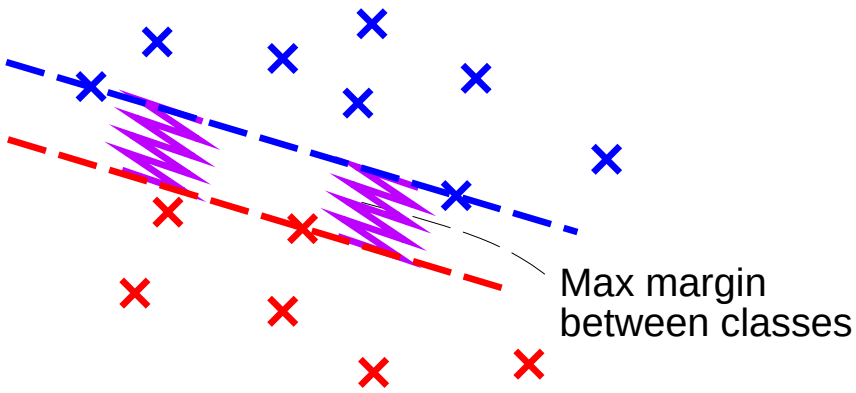
– Refined theory for convex otimization under constraints.

– Construct a *dual optimization problem*

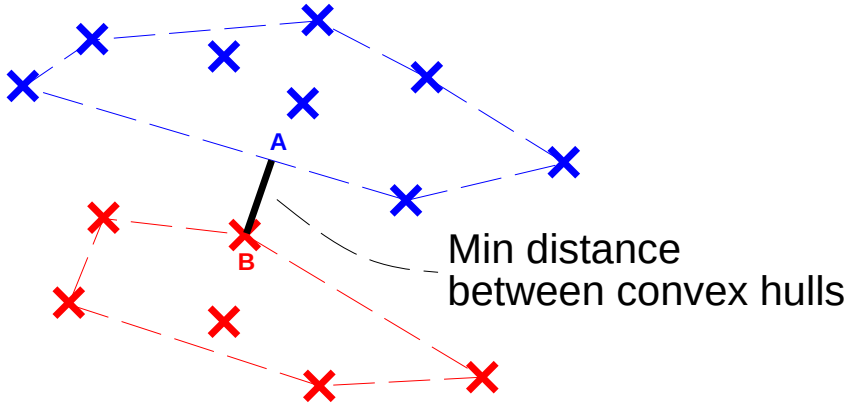
whose constraints are simpler,

and whose solution is related to the solution we seek.

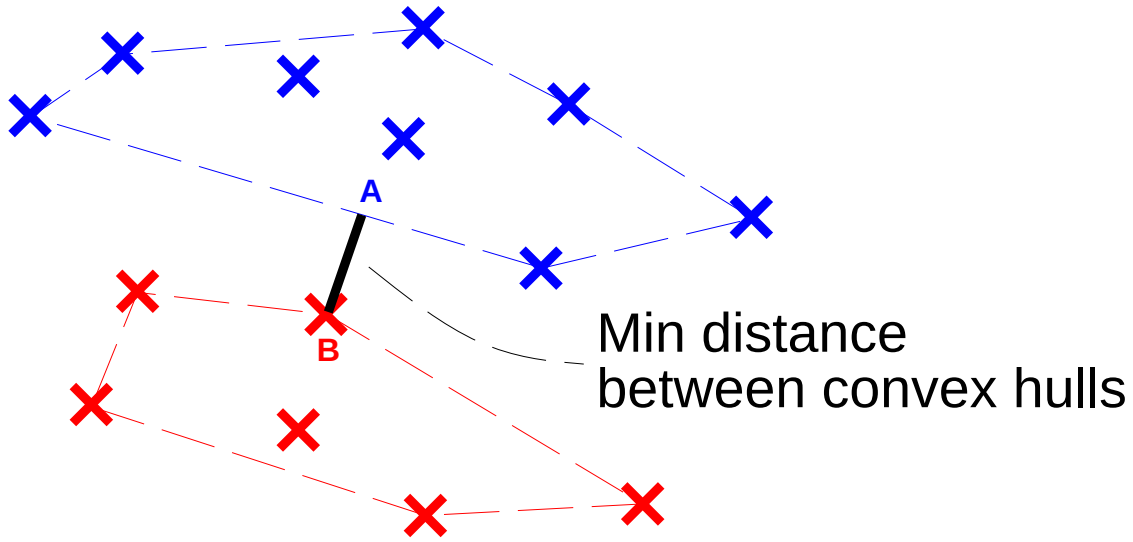
Primal formulation



Dual formulation



Dual formulation

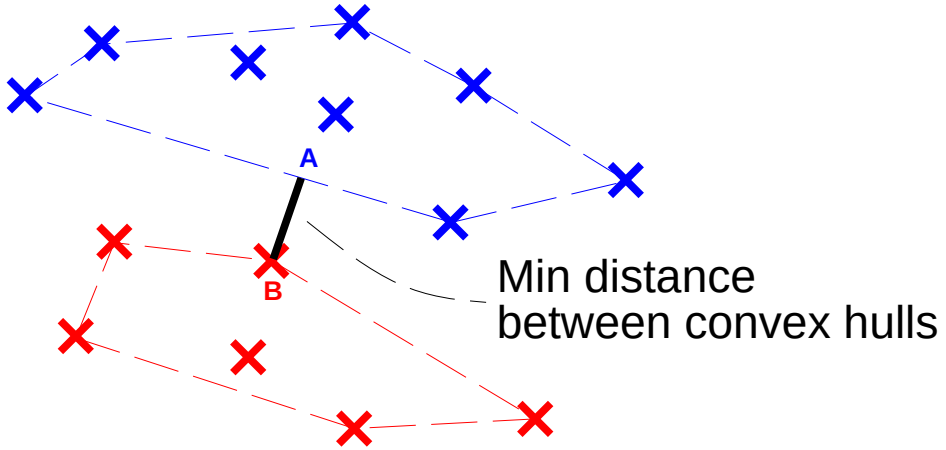


– Point A: $\sum_{i \in \text{Pos}} \beta_i \, \mathbf{x}_i$ subject to $\beta_i \geq 0$ and $\sum_{i \in \text{Pos}} \beta_i = 1$

– Point B: $\sum_{i \in \text{Neg}} \beta_i \, \mathbf{x}_i$ subject to $\beta_i \geq 0$ and $\sum_{i \in \text{Neg}} \beta_i = 1$

– Vector BA: $\sum_i y_i \, \beta_i \, \mathbf{x}_i$ subject to $\beta_i \geq 0$, $\sum_i \beta_i = 2$, and $\sum_i y_i \, \beta_i = 0$.

Dual formulation



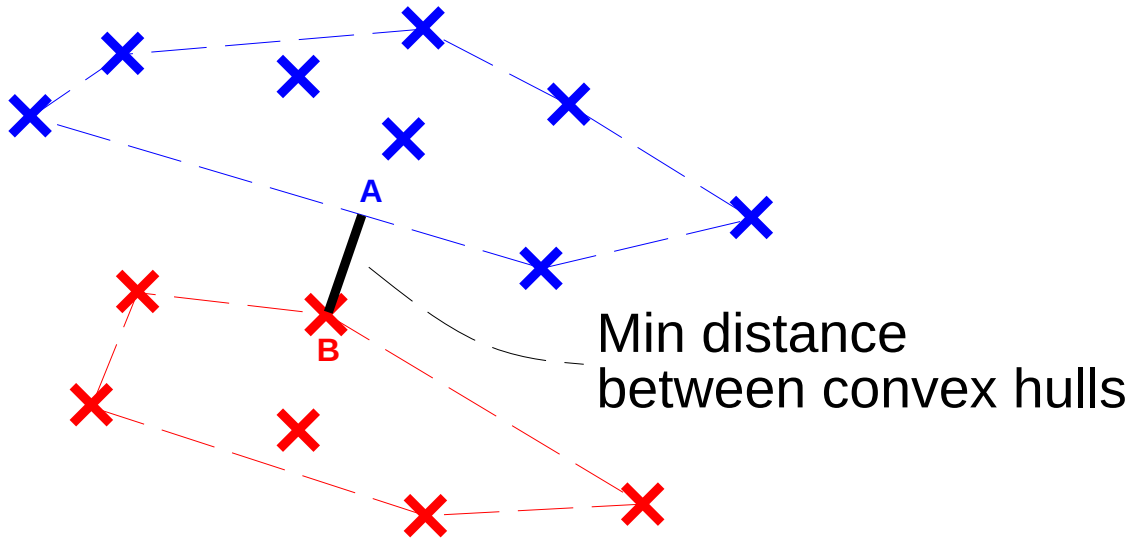
$$\min_{\boldsymbol{\beta}} \sum_{ij} y_i y_j \, \beta_i \beta_j \, \mathbf{x}_i^\top \mathbf{x}_j \quad \text{subject to} \quad \begin{cases} \forall i \quad \beta_i \geq 0 \\ \sum_i y_i \beta_i = 0 \\ \sum_i \beta_i = 2 \end{cases}$$

Then $\mathbf{w} = \sum_i \; y_i \, \beta_i \, \mathbf{x}_i$.

Then b is easy to find by projecting all examples on $\mathbf{w}$.

Dual formulation

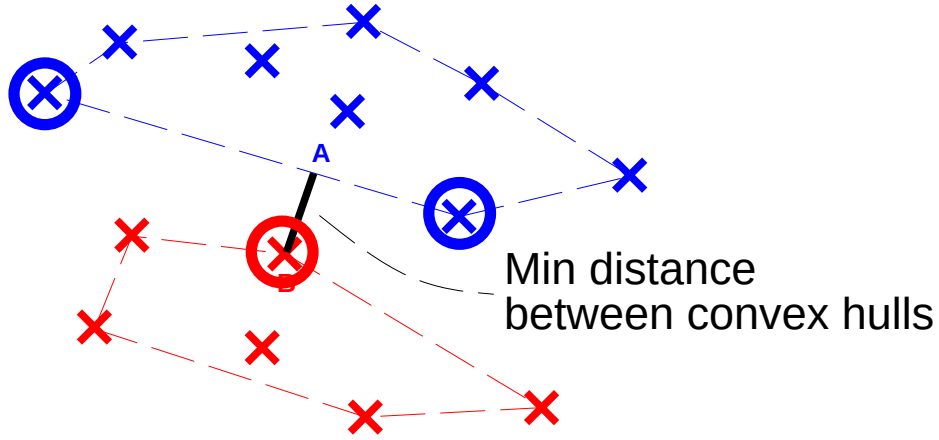
Classic formulation



$$\max_{\boldsymbol{\alpha}} \sum_i \alpha_i - \frac{1}{2} \sum_{ij} y_i y_j \, \alpha_i \alpha_j \, \mathbf{x}_i^\top \mathbf{x}_j \quad \text{subject to} \quad \begin{cases} \forall i \quad \alpha_i \geq 0 \\ \sum_i y_i \alpha_i = 0 \end{cases}$$

This is equivalent with $\alpha_i = \beta_i \Delta^{-2}$ but the proof is nontrivial.

Support Vectors Machines



$$\min_{\boldsymbol{\beta}} \sum_{ij} y_i y_j \, \beta_i \beta_j \, \mathbf{x}_i^\top \mathbf{x}_j \quad \text{subject to} \quad \begin{cases} \forall i \quad \beta_i \geq 0 \\ \sum_i y_i \beta_i = 0 \\ \sum_i \beta_i = 2 \end{cases}$$

The only non zero β_i are those corresponding to support vectors.

Leave-One-Out

Leave one out = n-fold cross-validation

– Compute classifiers f_i using training set minus example $(\mathbf{x}_i, y_i)$.

– Estimate test misclassification rate as $E_{LOO} = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\left\{ y_i f_i(\mathbf{x}_i) \leq 0 \right\}$.

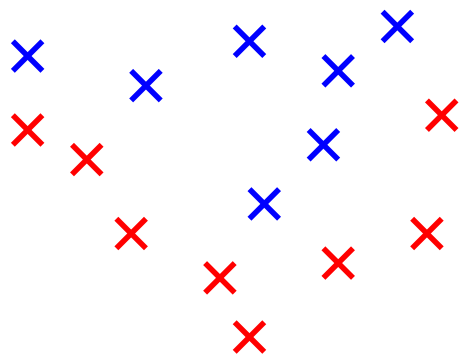
Leave one out for maximal margin classifier

– Removing a non support vector does not change the classifier.

$$E_{LOO} \leq \frac{\#\text{support vectors}}{\#\text{examples}}$$

– The important quantity is not the dimension
but is the number of support vectors.

Soft margins



When the examples are
not linearly separable,
the constraints $y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1$
cannot be satisfied.

Adding slack variables ξ_i

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \|\mathbf{w}\|^2 + C \sum_{i=1}^{n} \xi_i \quad \text{subject to} \quad \forall i \;\; y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \;, \quad \xi_i \geq 0$$

Parameter C controls the relative importance of:
– correctly classifying all the training examples,
– obtaining the separation with the largest margin.

Reduces to hard margins when $C = \infty$.

Soft margins and Hinge loss

The soft margin problem

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \quad \text{subject to} \quad \forall i \quad y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \ , \quad \xi_i \geq 0$$

is the same thing as

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\ell(y_i(\mathbf{w}^\top \mathbf{x}_i + b)) \quad \text{with} \quad \ell(z) = \max(0, 1 - z)$$

Soft Margins

Primal formulation

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \quad \text{subject to} \quad \forall i \;\; y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \;\; , \quad \xi_i \geq 0$$

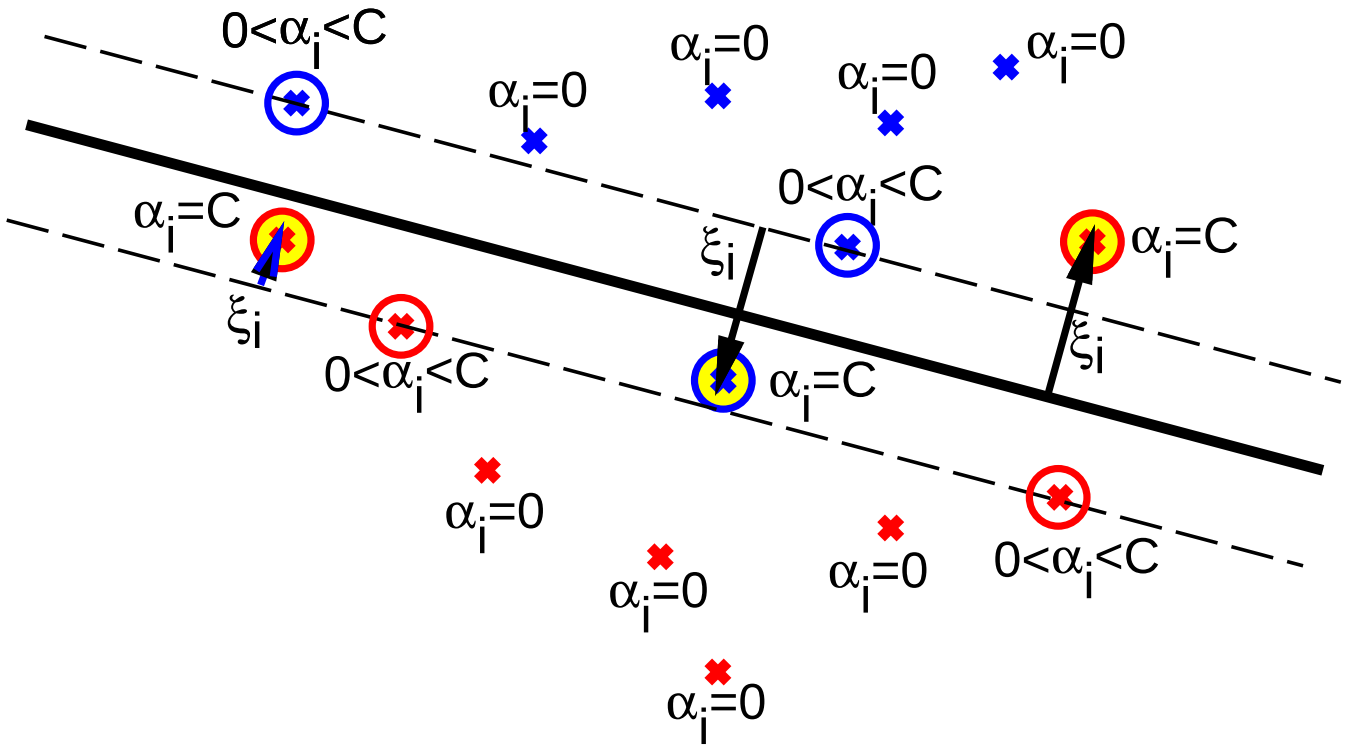
Dual formulation

$$\max_{\boldsymbol{\alpha}} \sum_i \alpha_i - \frac{1}{2}\sum_{ij} y_i y_j \, \alpha_i \alpha_j \, \mathbf{x}_i^\top \mathbf{x}_j \quad \text{subject to} \quad \begin{cases} \forall i \;\; 0 \leq \alpha_i \leq C \\ \sum_i y_i \alpha_i = 0 \end{cases}$$

The primal and dual solutions obey the relation $\mathbf{w} = \sum_{i=1}^{n} y_i \, \alpha_i \, \mathbf{x}_i$.

The threshold b is easy to find once $\mathbf{w}$ is known.

Soft Margins



Beyond linear separation

Reintroducing the $\Phi(\mathbf{x})$

– Define $K(\mathbf{x}, \mathbf{v}) = \Phi(\mathbf{x})^\top \Phi(\mathbf{v})$.

– Dual optimization problem

$$\max_{\boldsymbol{\alpha}} \sum_i \alpha_i - \frac{1}{2} \sum_{ij} y_i y_j \, \alpha_i \alpha_j \, K(\mathbf{x}_i, \mathbf{x}_j) \quad \text{subject to} \quad \begin{cases} \forall i \quad 0 \le \alpha_i \le C \\ \sum_i y_i \alpha_i = 0 \end{cases}$$

– Discriminant function

$$f(\mathbf{x}) = \mathbf{w}^\top \Phi(\mathbf{x}) + b = \sum_{i=1}^{n} y_i \, \alpha_i \, K(\mathbf{x}_i, \mathbf{x})$$

Curious fact

– We do not really need to compute $\Phi(\mathbf{x})$.
– The dot products $K(\mathbf{x}, \mathbf{v}) = \Phi(\mathbf{x})^\top \Phi(\mathbf{v})$ are enough.
– Can we take advantage of this?

Quadratic Kernel

Quadratic basis

$$\Phi(\mathbf{x}) = \left(\left[x_i \right]_i , \left[x_i^2 \right]_i , \left[\sqrt{2}\, x_i x_j \right]_{i<j} \right)$$

Dot product

$$\Phi(\mathbf{x})^\top \Phi(\mathbf{v}) = \sum_i x_i v_i + \sum_i x_i^2 v_i^2 + \sum_{i<j} 2\, x_i v_i x_j v_j$$

– Are there $\boldsymbol{d(d+3)/2}$ terms to add ?

Quadratic Kernel

Quadratic basis

$$\Phi(\mathbf{x}) = \left(\left[x_i \right]_i , \left[x_i^2 \right]_i , \left[\sqrt{2}\, x_i x_j \right]_{i<j} \right)$$

Dot product

$$\begin{aligned}
\Phi(\mathbf{x})^\top \Phi(\mathbf{v}) &= \sum_i x_i v_i + \sum_i x_i^2 v_i^2 + \sum_{i<j} 2\, x_i v_i x_j v_j \\
&= \sum_i x_i v_i + \sum_{i,j} x_i v_i x_j v_j \\
&= \sum_i x_i v_i + \left(\sum_i x_i v_i \right)^2 \quad = \quad (\mathbf{x}^\top \mathbf{v}) + (\mathbf{x}^\top \mathbf{v})^2
\end{aligned}$$

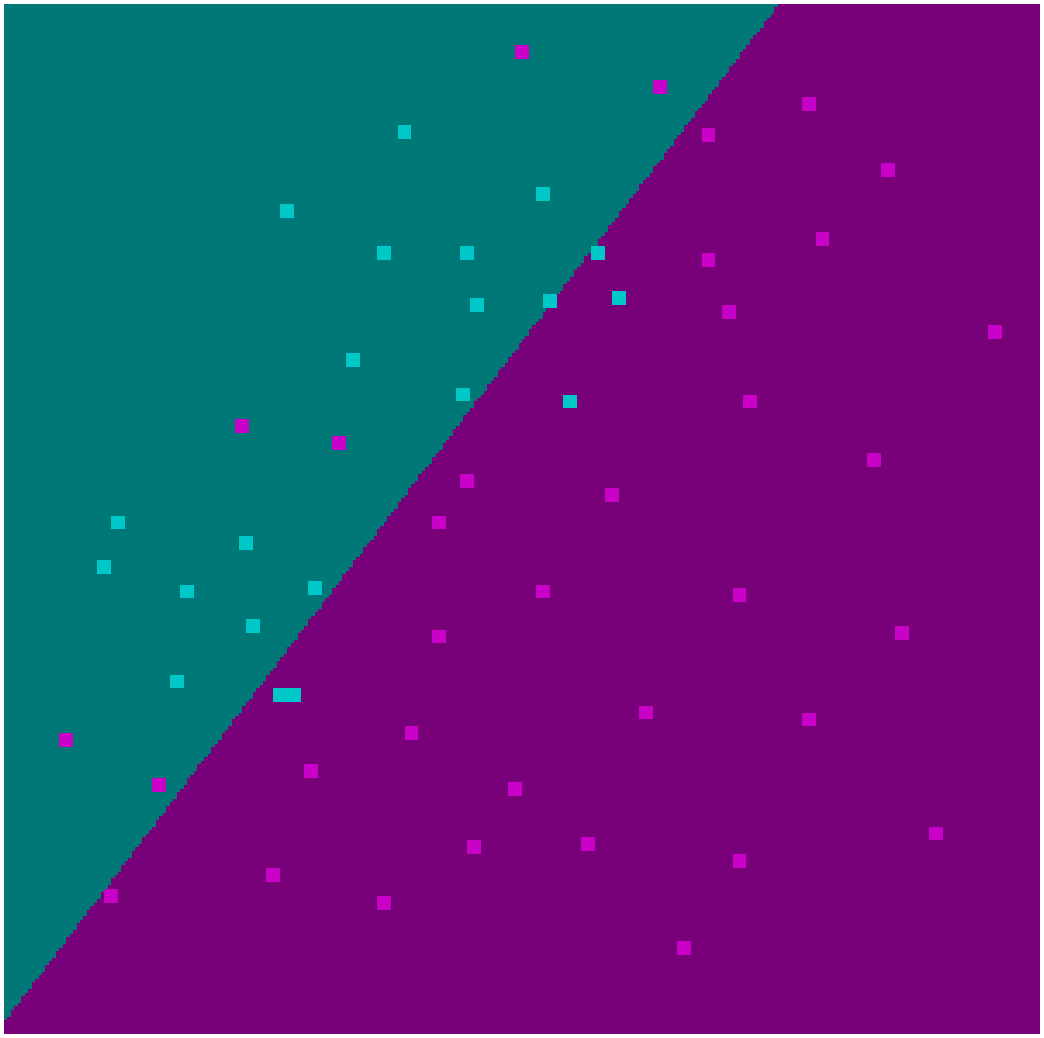
– There are only $\boldsymbol{d}$ terms to add !

Polynomial kernel

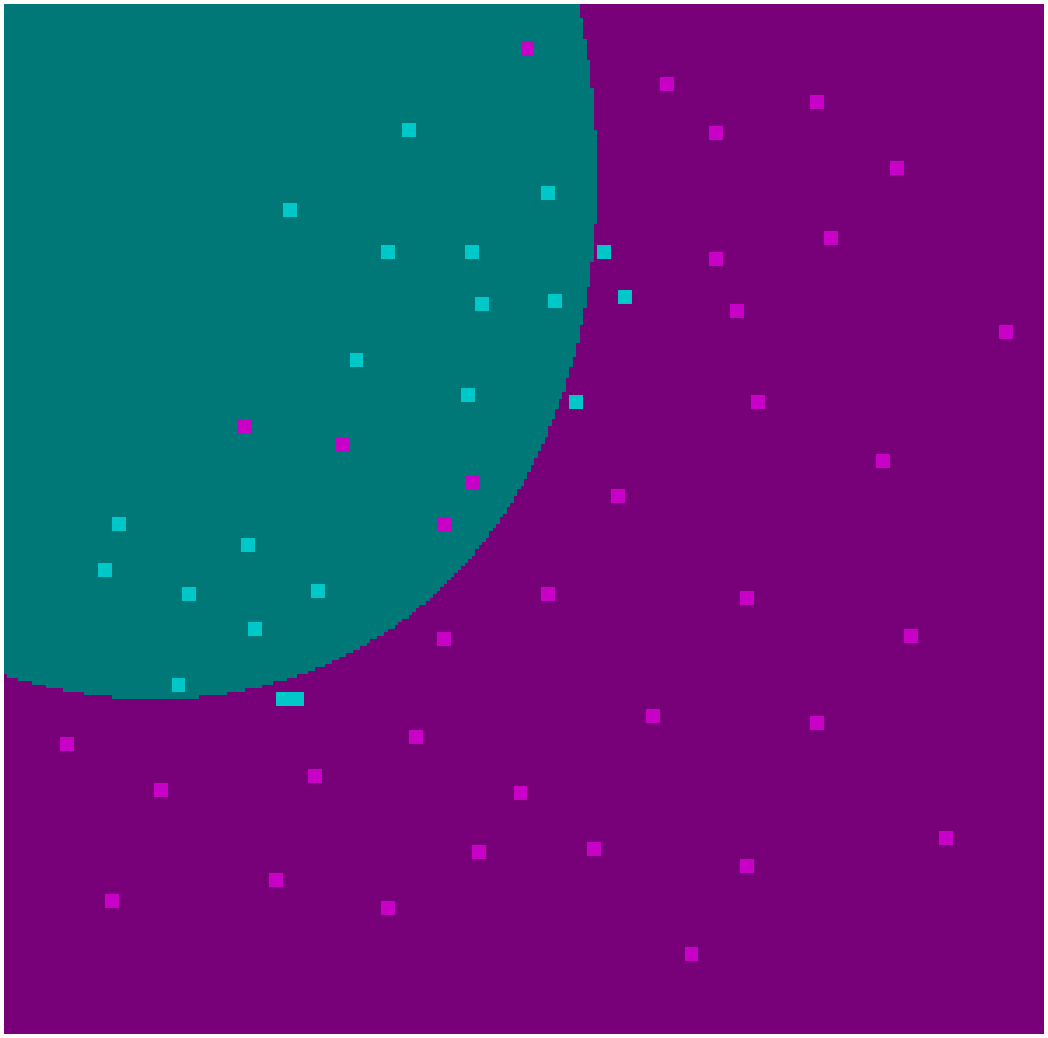
Degree	$\dim(\Phi(\mathbf{x}))$	$\Phi(\mathbf{x})^\top \Phi(\mathbf{v})$
1	d	$(\mathbf{x}^\top \mathbf{v})$
2	$\approx d^2/2$	$(\mathbf{x}^\top \mathbf{v}) + (\mathbf{x}^\top \mathbf{v})^2$
3	$\approx d^3/6$	$(\mathbf{x}^\top \mathbf{v}) + (\mathbf{x}^\top \mathbf{v})^2 + (\mathbf{x}^\top \mathbf{v})^3$
...		
n	$\approx d^n/n!$	$(1 + \mathbf{x}^\top \mathbf{v})^d$

The number of parameters increases exponentially quickly.
But the total computation remains nearly constant.

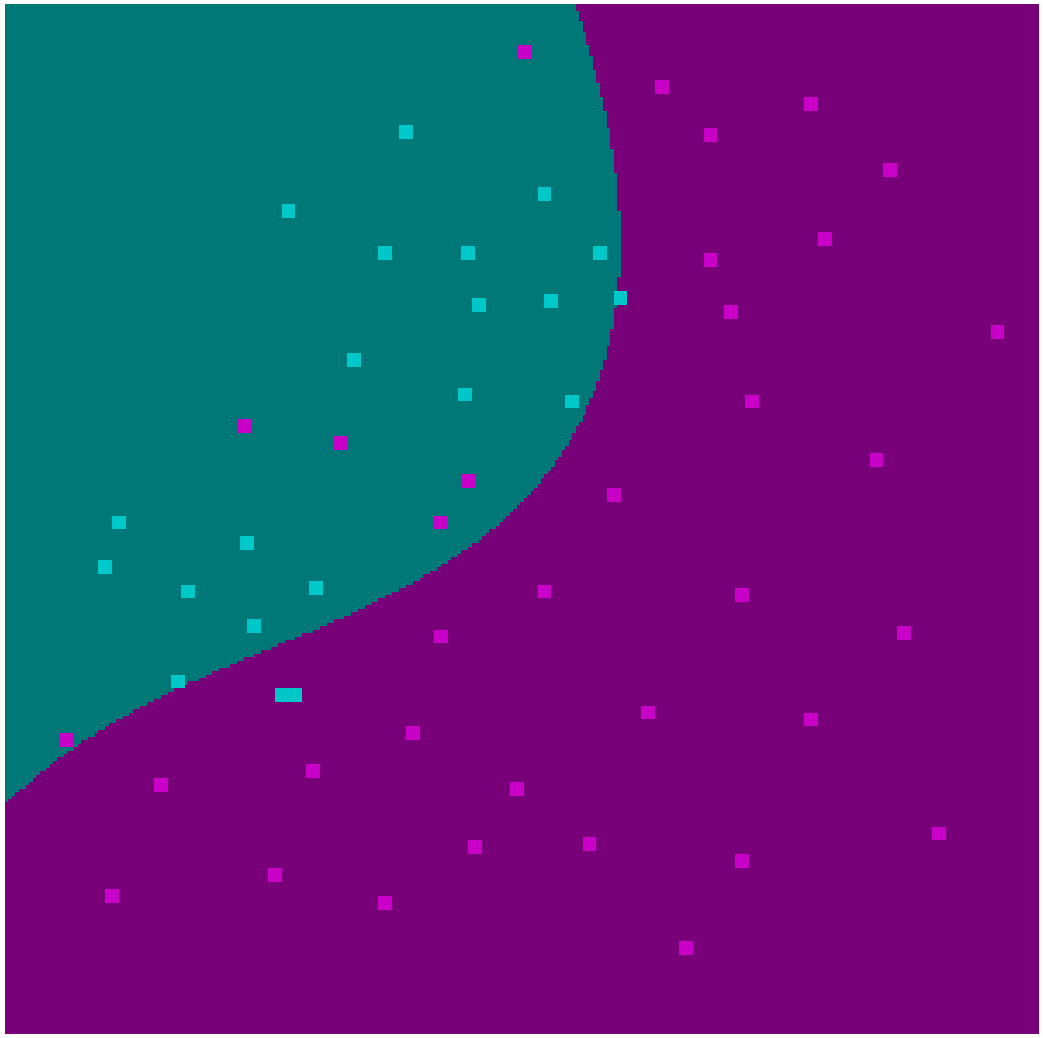
Linear



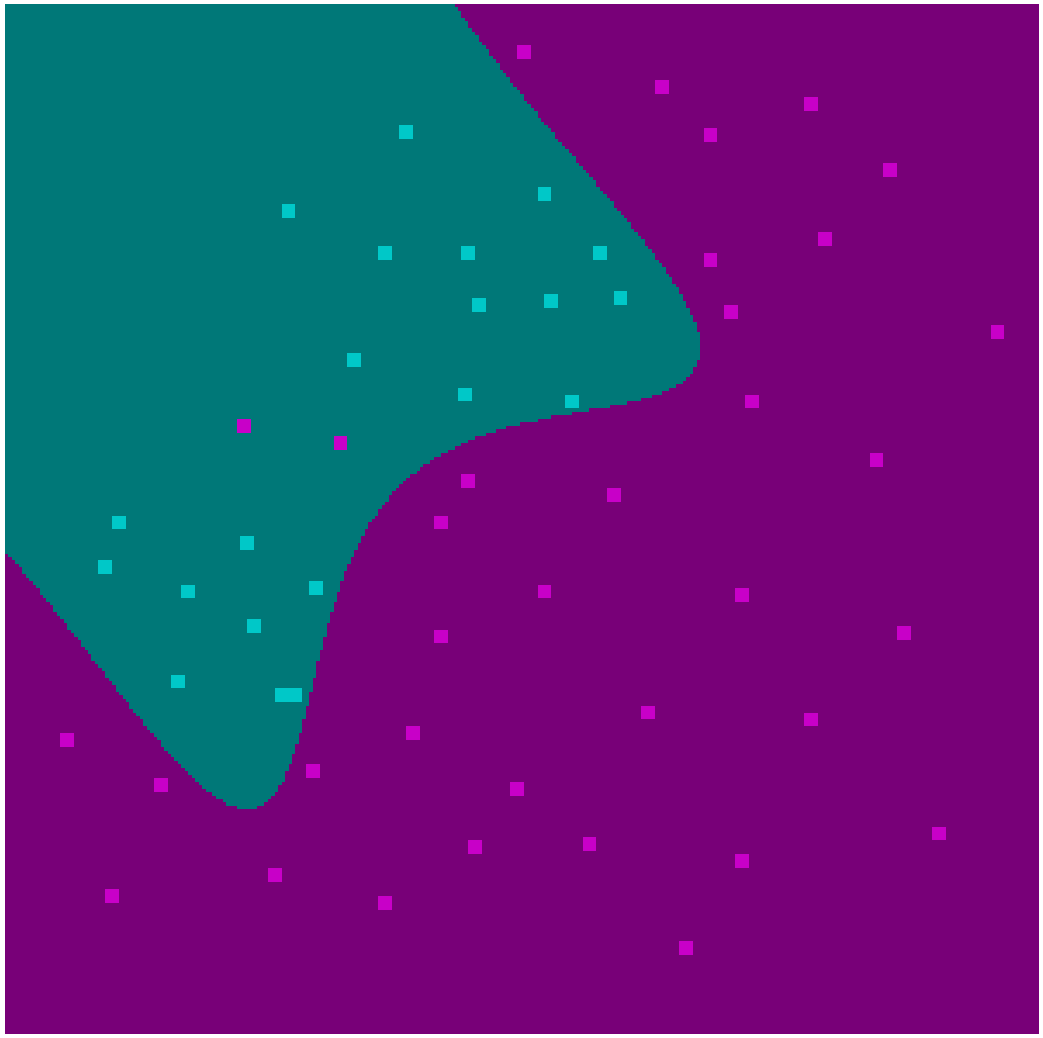
Quadratic



Polynomial degree 3



Polynomial degree 5



Polynomial kernels and more

Weighted polynomial kernel: $K_d(\mathbf{x}, \mathbf{v}) = \sum_{i=0}^{d} \frac{\gamma^i}{i!} (\mathbf{x}^\top \mathbf{v})^i$.

– This is a polynomial kernel.

– Coefficient γ controls the relative importance of terms of various degree.

Polynomial kernels and more

Weighted polynomial kernel: $K_d(\mathbf{x}, \mathbf{v}) = \sum_{i=0}^{d} \frac{\gamma^i}{i!} (\mathbf{x}^\top \mathbf{v})^i$.

– This is a polynomial kernel.
– Coefficient γ controls the relative importance of terms of various degree.

Exponential kernel: $K_\infty(\mathbf{x}, \mathbf{v}) = \sum_{i=0}^{\infty} \frac{\gamma^i}{i!} (\mathbf{x}^\top \mathbf{v})^i = e^{\gamma\, \mathbf{x}^\top \mathbf{v}}$

– This is non longer a polynomial kernel.
– The dimension of $\Phi(\mathbf{x})$ is infinite.
– The computation remains finite.

Radial Basis Function kernel

Radial Basis Functions

– Approximating functions with expressions of the form

$$f_w(\mathbf{x}) = \sum_i w_i F(\|\mathbf{x} - \mathbf{x}_i\|)$$

– Gaussian kernel

$$F(r) = e^{-\gamma r^2}$$

Radial Basis Kernel

– Running a SVM with kernel $K(\mathbf{x}, \mathbf{v}) = e^{-\gamma\|\mathbf{x}-\mathbf{v}\|^2}$
results in a discriminant function

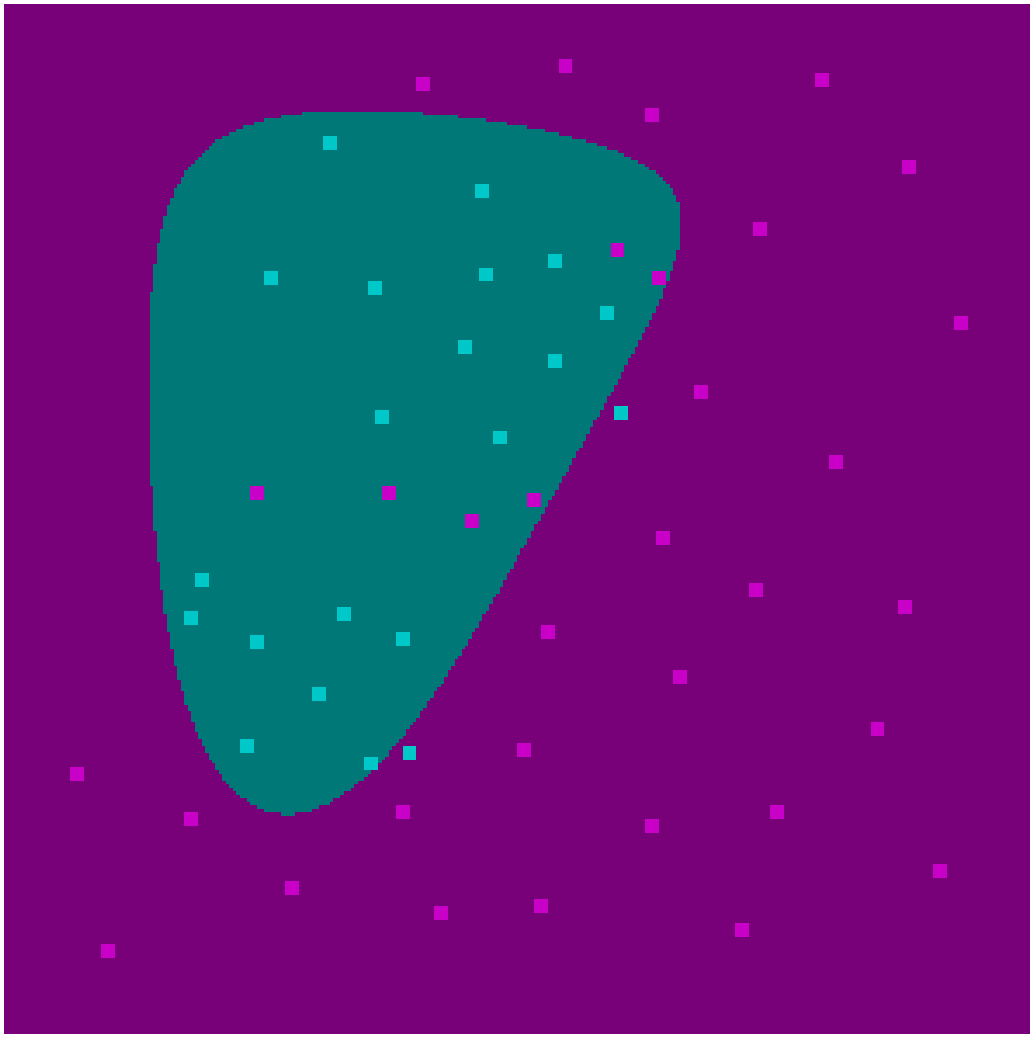
$$f_w(\mathbf{x}) = \sum_i y_i \alpha_i e^{-\gamma\|\mathbf{x}-\mathbf{x}_i\|^2}$$

Questions

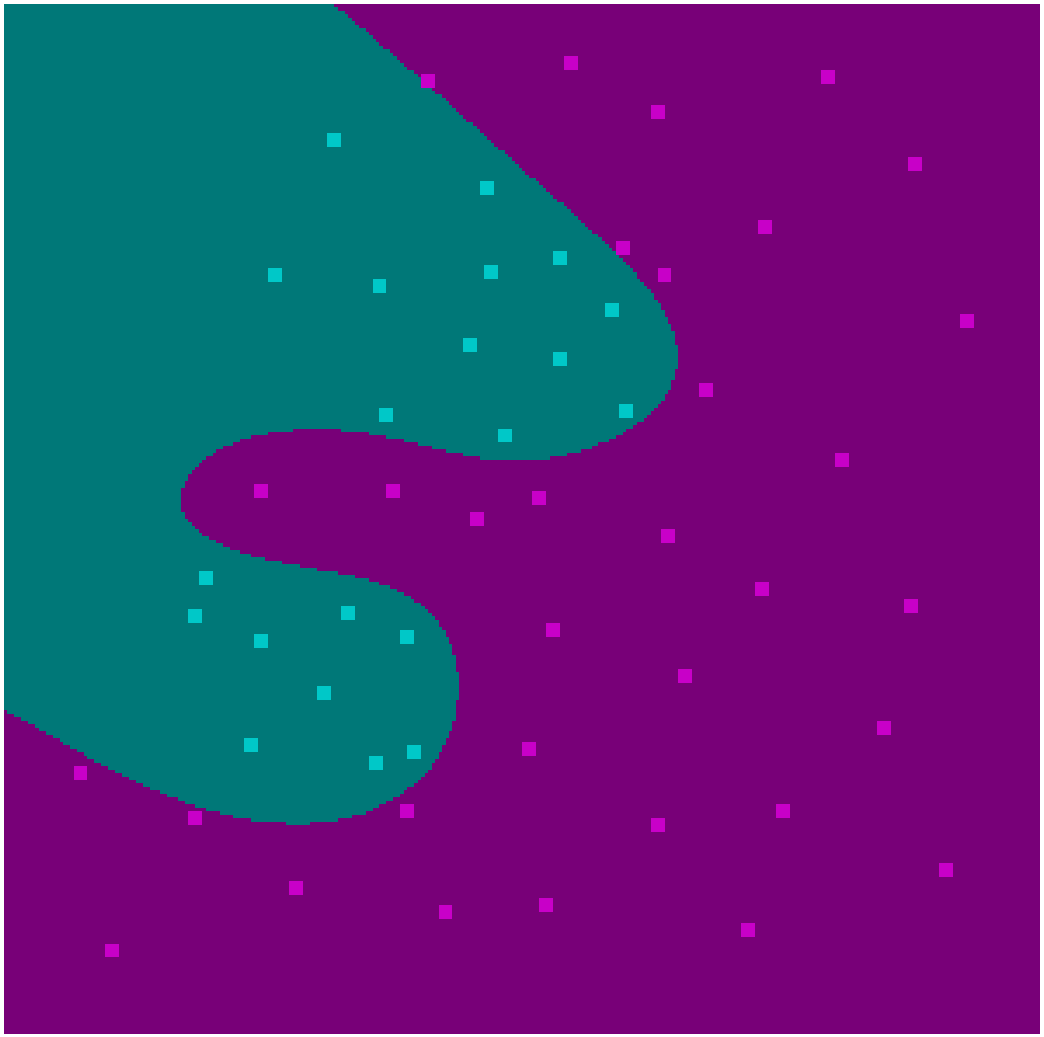
– Is there a function Φ that corresponds to this kernel?

– Does this work?

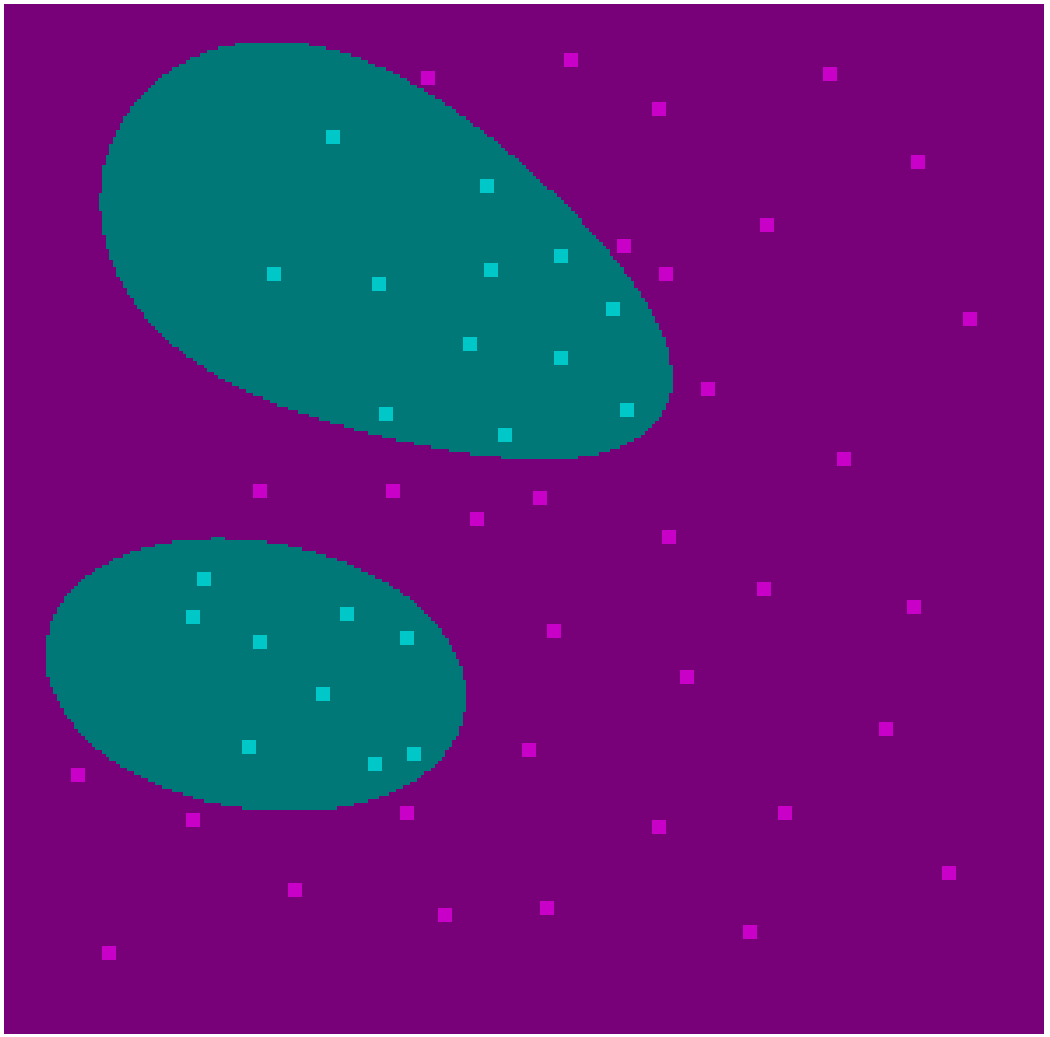
Radial Basis (gamma = 0.1)



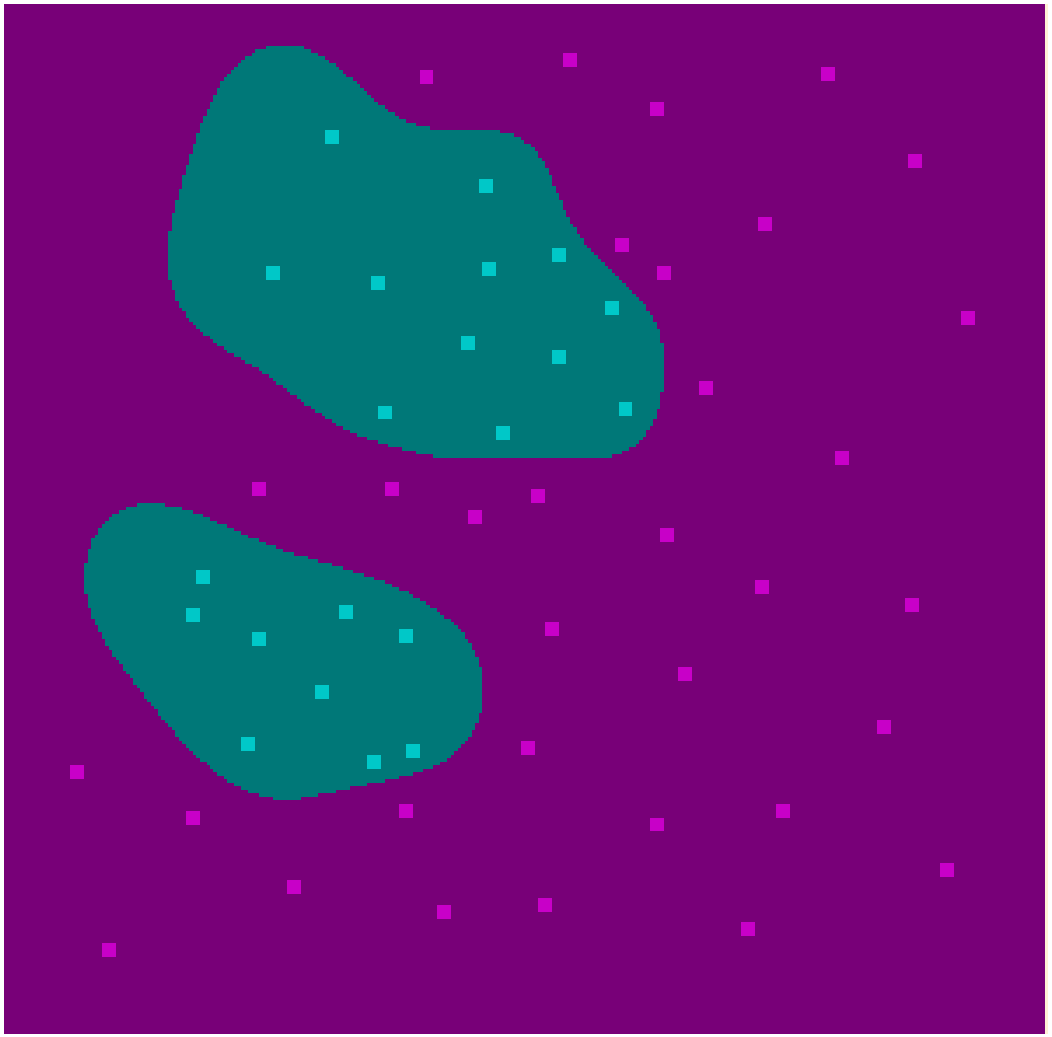
Radial Basis (gamma = 1)



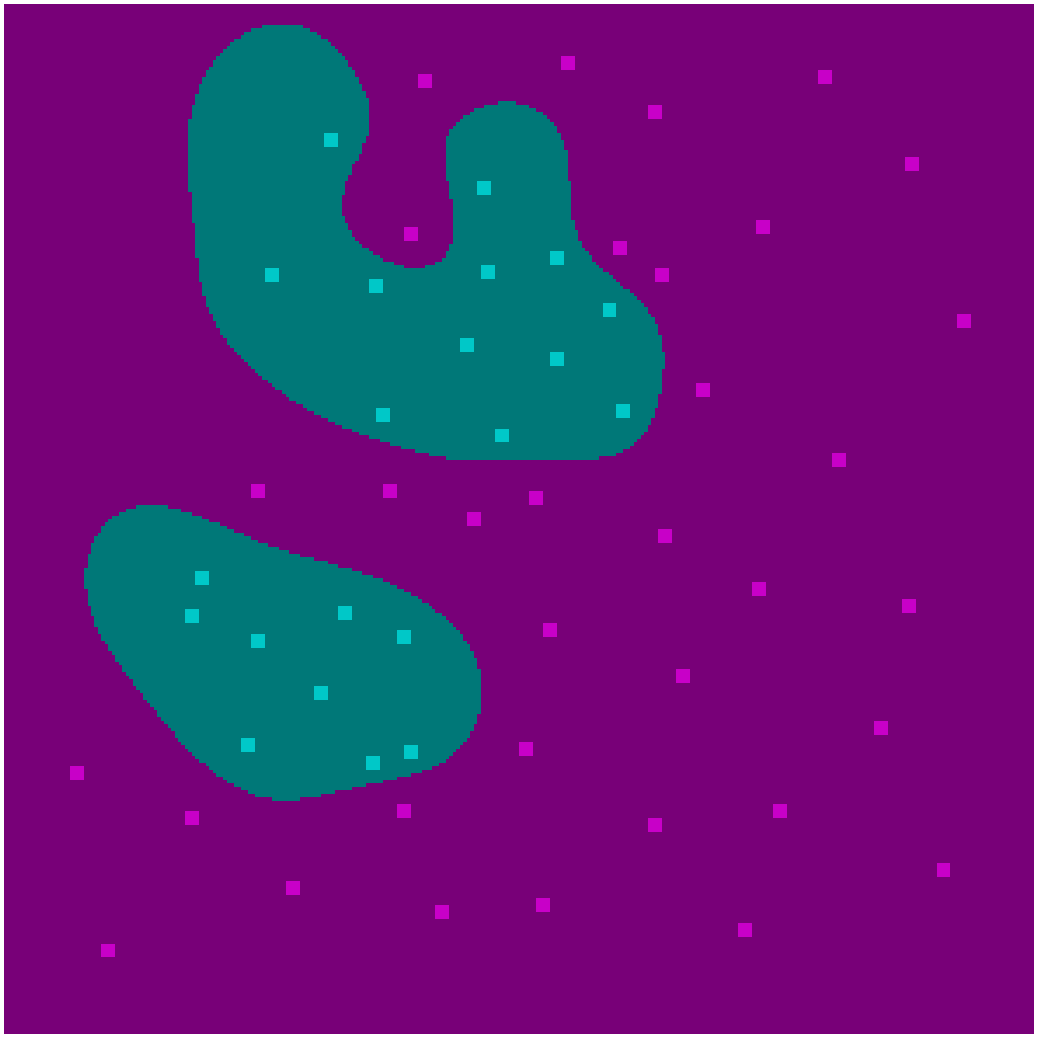
Radial Basis (gamma = 10)



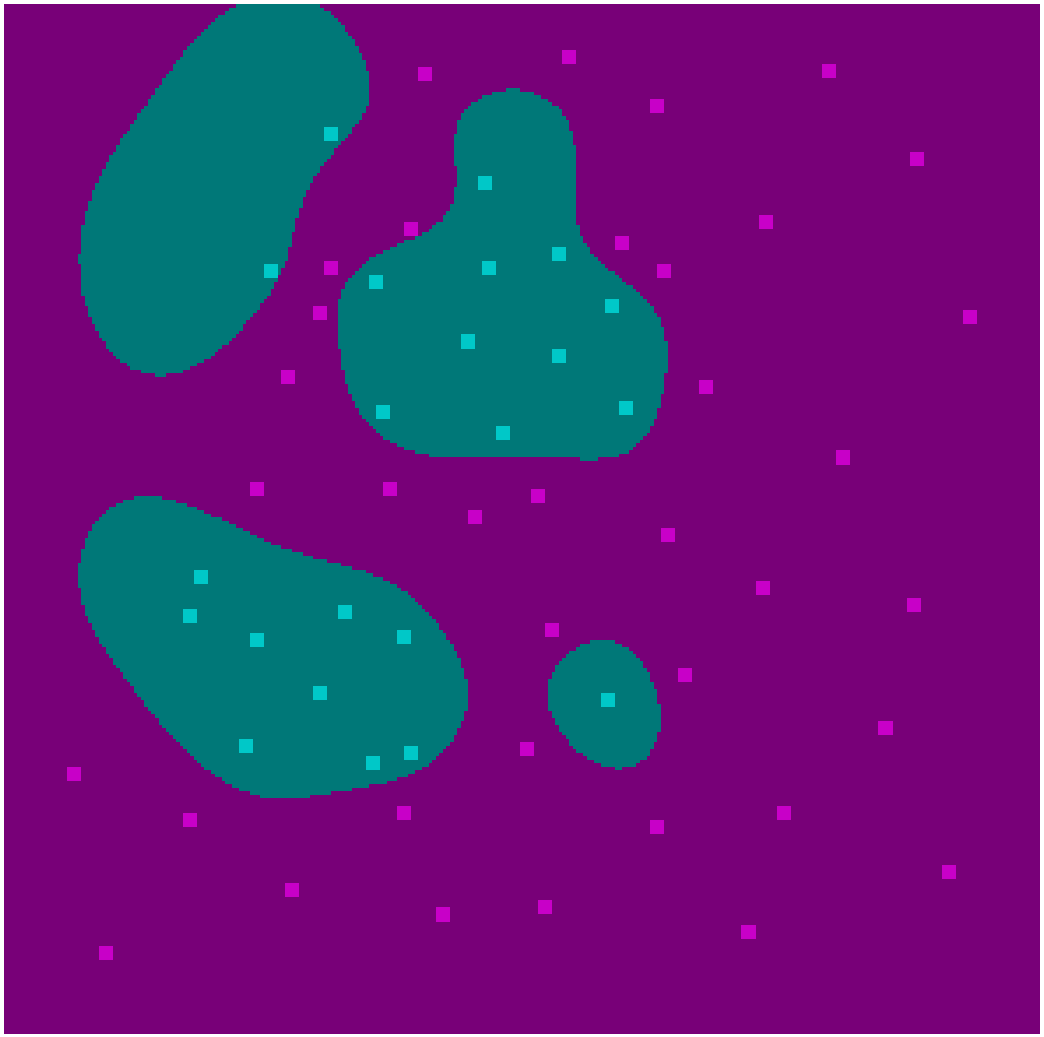
Radial Basis (gamma = 100)



Radial Basis (gamma = 100)



Radial Basis (gamma = 100)



Mercer kernel

Definition

– Kernel $K(\mathbf{x}, \mathbf{v})$ is a Mercer kernel iff it is

1. symmetric: $\forall \mathbf{x}, \mathbf{v} \quad K(\mathbf{x}, \mathbf{v}) = K(\mathbf{v}, \mathbf{x})$
2. positive: $\forall k \quad \forall \mathbf{x}_1 \ldots \mathbf{x}_k \quad \forall c_1 \ldots c_k \quad \sum_{i,j=1}^{k} c_i c_j K(\mathbf{x}_i, \mathbf{x}_j) \geq 0$

Mercer theorem

– For any Mercer kernel $K(\mathbf{x}, \mathbf{v})$
there exists a vectorial space Ω
and a function $\Phi : \mathbf{x} \mapsto \Phi(\mathbf{x}) \in \Omega$
such that $K(\mathbf{x}, \mathbf{v}) = \Phi(\mathbf{x})^\top \Phi(\mathbf{v})$.

Practical consequences

– We can create models by specifying basis functions $\Phi(\mathbf{x})$.
– We can also create models by specifying kernels $K(\mathbf{x}, \mathbf{v})$.

Usual and customary kernels

	$K(\mathbf{x}, \mathbf{v})$	**Decision boundary**	**Dim**(Φ-space)
linear	$\mathbf{x}^\top \mathbf{v}$	hyperplanes	n
quadratic	$\mathbf{x}^\top \mathbf{v} + \mathbf{x}^\top \mathbf{v}^2$	conics	$\frac{n(n+3)}{2}$
d-polynomial	$(1 + \mathbf{x}^\top \mathbf{v})^d$	?	$\equiv \frac{n^d}{d}$
gaussian:	$\exp(-\gamma \|\|\mathbf{x} - \mathbf{v}\|\|^2)$	smooth	∞

More kernels

	$K(\mathbf{x}, \mathbf{v})$
spline	$1 + \mathbf{x}^\top \mathbf{v} + \prod_{j=1}^{d} \int_{-R}^{R} [x_j - t]_+ [v_j - t]_+ \, dt$
multilayer perceptron	$\tanh(\alpha \mathbf{x}^\top \mathbf{v} - \beta)$
sum	$\sum_j \lambda_j K_j(\mathbf{x}, \mathbf{v}) \qquad \lambda_j \geq 0$
tensor product	$\prod_j K_j(x_j, v_j)$

Exotic kernels (1)

Input space needs not be a vector space.

Kernels defined on histograms and p.d.f.

	$K(\mathbf{x}, \mathbf{v})$
Kullback	$\exp(-\beta(D(\mathbf{x}\|\mathbf{v}) + D(\mathbf{v}\|\mathbf{x})))$
Jensen	$\exp(-\beta(D(\mathbf{x}\|\frac{\mathbf{x}+\mathbf{v}}{2}) + D(\mathbf{v}\|\frac{\mathbf{x}+\mathbf{v}}{2})))$
Hellinger	$\exp\left(-\beta \int \sqrt{x(t)} - \sqrt{v(t)}\, dt\right)$

Exotic kernels (2)

Input space needs not be a vector space.

Kernels defined on sequences.

	$K(\mathbf{x}, \mathbf{v})$
Fisher	$\left[\frac{\partial \log L}{\partial \lambda}(\mathbf{x})\right]^\top \left[\frac{\partial \log L}{\partial \lambda}(\mathbf{v})\right]$ where $L(.)$ is the likelihood of a H.M.M.
string	number of common substrings of length d
rational	defined by certain finite state automatons

Kernels everywhere

Kernel Principal Component Analysis.
Compute principal subspaces in feature space.

- Eigenvectors in Φ-space defined as:

$$E_p = \sum_i \alpha_{i,p} \Phi(\mathbf{x}_i)$$

- Cannot find pre-images $\mathbf{e}_k$ such that $E_k = \Phi(\mathbf{e}_k)$.
  But can extract components in principal subspace:

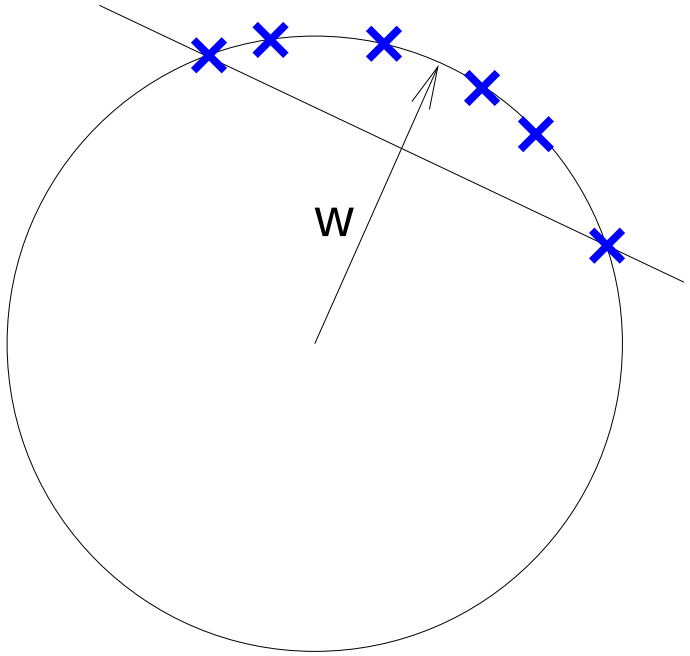
$$s_k(\mathbf{x}) = \sum_i \alpha_{i,k} K(\mathbf{x}, \mathbf{x}_i)$$

- Related to Isomap, LLE, Spectral Clustering.

Kernels everywhere

One Class Support Vector Machines.

Locate the support of the data distribution.



Assume $||x_i|| = 1$

Min$||w||^2$ with $\forall i,\ w.x_i \geq 1$.

Best done in Φ-space of course.

Example: Novelty detection.

Kernels everywhere

More kernel algorithms.

Kernelizing a standard algorithm was in fashion

SVR	Support Vector Regression
KLDA	Kernel Linear Discriminant Analysis
LS-SVM	Least Square Support Vector Machine
KLR	Kernel Logistic Regression
...	

and led to the rediscovery of old algorithms:

Aizerman-Braverman Potential Functions	
→	Kernel Perceptron, Kernel Adatron, etc.
Gaussian Processes	
→	Kriging

Conclusion

Soft-margin SVM

– a classifier using the hinge loss
– with a kernel representation
– and capacity control using regularization.

Obvious variants

– change the loss
– change the representation
– change the regularizer. . .

Outlook

Success stories

– Text categorization

– Classification tasks in general
  the best classifier can change a lot,
  but the SVM is rarely far away.

Weak points

– Computationally costly with noisy data

– L2 regularization works poorly when irrelevant inputs abound.