

CS195-5 : Introduction to Machine Learning

Lecture 16

Greg Shakhnarovich

October 11, 2006

Announcements

- Lectures: revisions will be posted.
- Midterm: next Friday, 10/27, in 367.
- Projects
 - Proposal: 2 pages, due no later than 11/22.
 - Some suggestions:
 - * <http://www.netflixprize.com> (potentially \$1M prize)
 - * NIST TREC evaluations trec.nist.gov
 - * Computer vision project (face recognition/analysis, object recognition, . . .)
- Plan for the rest of the term

Plan for today

- SVM:
 - non-separable case;
 - nonlinear classification with the kernel trick.

SVM: summary so far

- Assuming linearly separable case, we set up a quadratic program

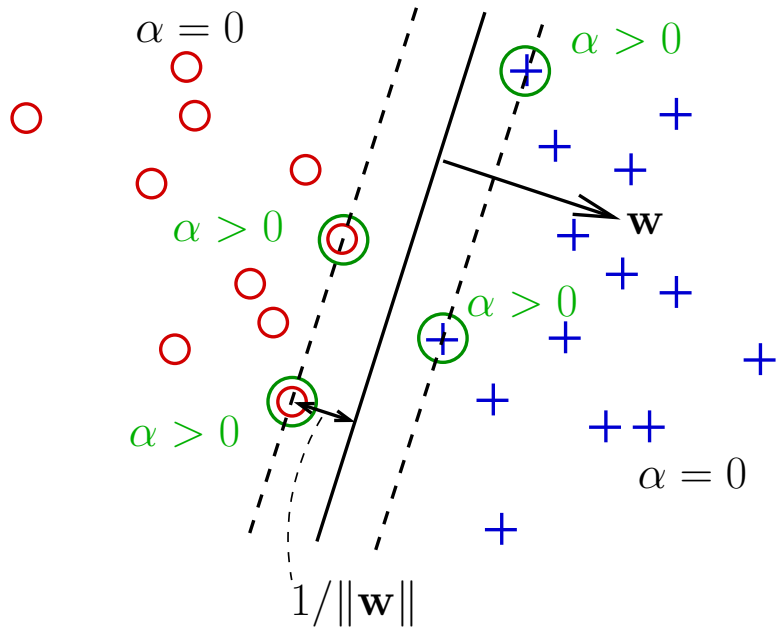
$$\max \left\{ \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j=1}^N \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \right\}$$

subject to $\sum_{i=1}^N \alpha_i y_i = 0$, $\alpha_i \geq 0$ for all $i = 1, \dots, N$.

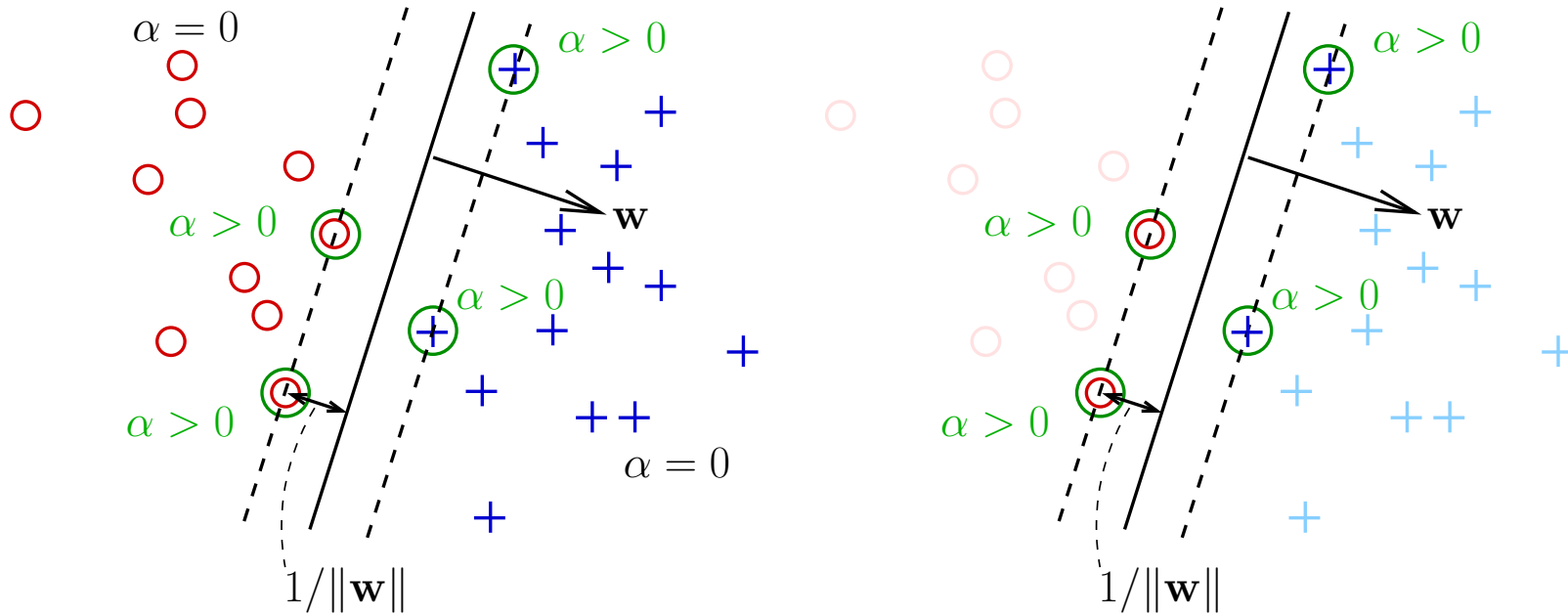
- Solving it for α we get the SVM classifier

$$\hat{y} = \text{sign} \left(\hat{w}_0 + \sum_{\alpha_i > 0} \alpha_i y_i \mathbf{x}_i^T \mathbf{x} \right).$$

SVM classification



SVM classification



Non-separable case

- What if the training data are not linearly separable? We can no longer require exact margin constraints.

- We rewrite the constraints with *slack variables* $\xi_i \geq 0$:

$$y_i (w_0 + \mathbf{w}^T \mathbf{x}_i) - 1 + \xi_i \geq 0.$$

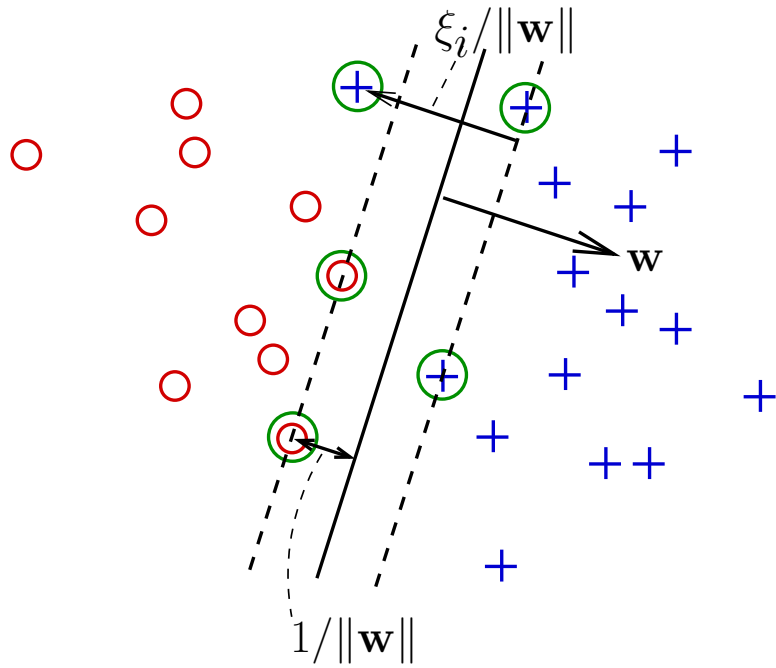
- Whenever the original constraint is satisfied, $\xi_i = 0$.

- The updated objective:

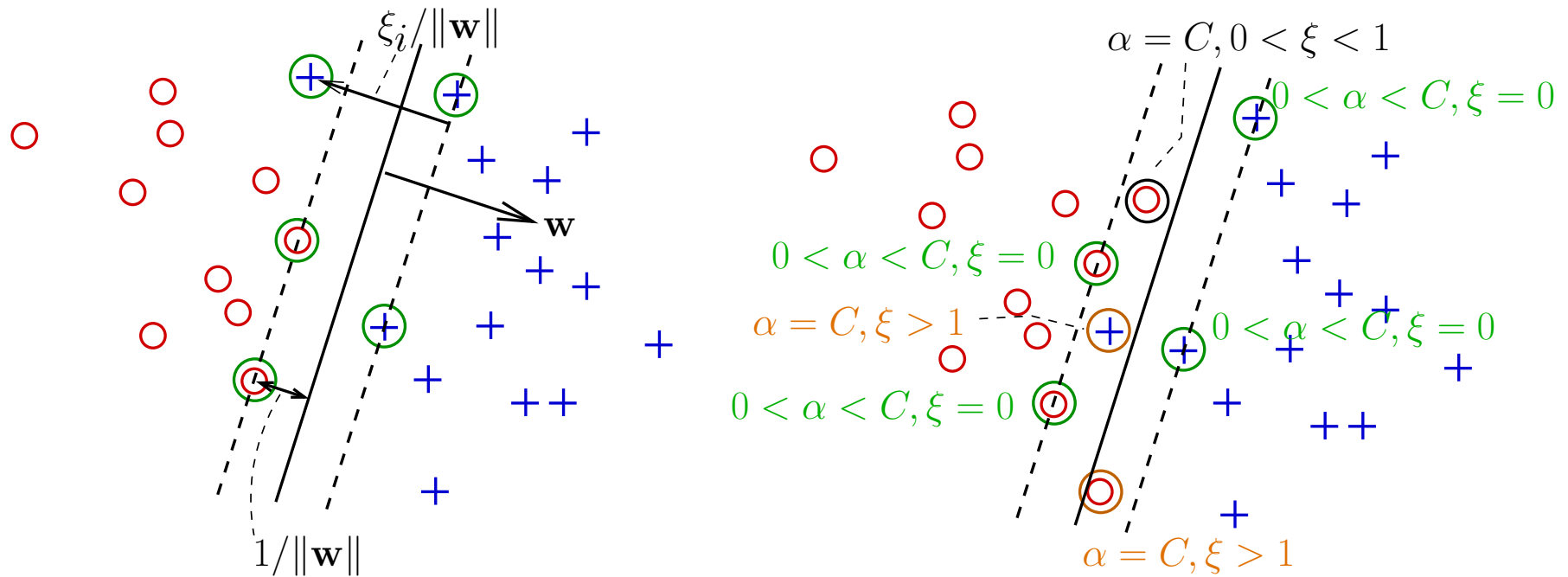
$$\min_{\mathbf{w}} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i.$$

- The parameter C determines the penalty paid for violating margin constraints.
- This is applicable even when the data *are* separable!

SVM with slack variables



SVM with slack variables



- Support vectors: points with $\alpha > 0$
- If $0 < \alpha < C$: SVs on the margin, $\xi = 0$.
- If $0 < \alpha = C$: over the margin, either misclassified ($\xi > 1$) or not ($0 < \xi \leq 1$).

Non-separable case: solution

$$\min_{\mathbf{w}} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i.$$

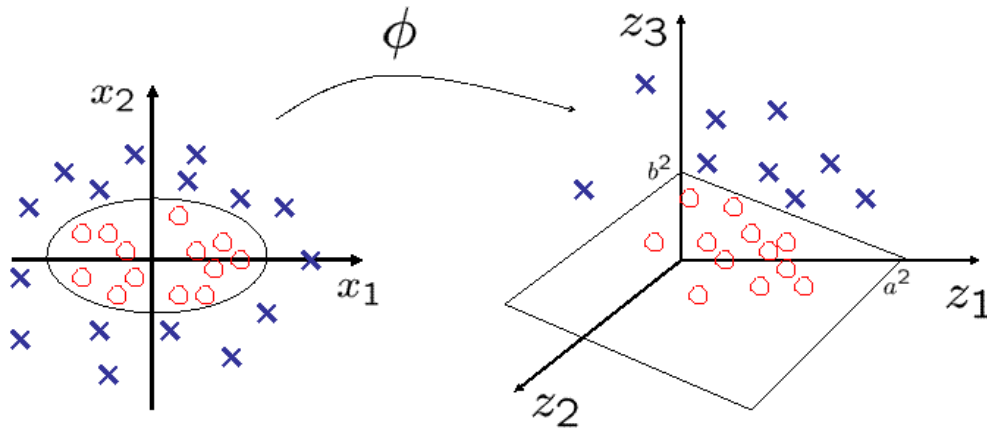
- We can solve this using Lagrange multipliers
 - Introduce additional multipliers for the ξ s.
- The resulting dual problem:

$$\max \left\{ \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j=1}^N \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \right\}$$

subject to $\sum_{i=1}^N \alpha_i y_i = 0$, $0 \leq \alpha_i \leq C$ for all $i = 1, \dots, N$.

Nonlinear features

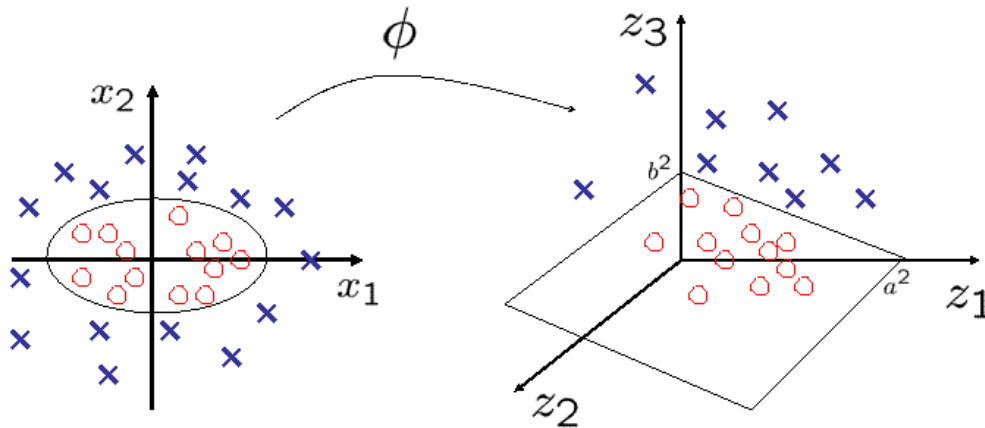
- As with logistic regression, we can move to nonlinear classifiers by mapping data into nonlinear *feature space*.



$$\phi : [x_1, x_2]^T \rightarrow [x_1^2, \sqrt{2}x_1x_2, x_2^2]^T$$

Nonlinear features

- As with logistic regression, we can move to nonlinear classifiers by mapping data into nonlinear *feature space*.



$$\phi : [x_1, x_2]^T \rightarrow [x_1^2, \sqrt{2}x_1x_2, x_2^2]^T$$

- Elliptical decision boundary in the input space becomes linear in the feature space $\mathbf{z} = \phi(\mathbf{x})$:

$$\frac{x_1^2}{a^2} + \frac{x_2^2}{b^2} = c \Rightarrow \frac{z_1}{a^2} + \frac{z_2}{b^2} = c.$$

Example of nonlinear mapping

- Consider the mapping: $\phi : [x_1, x_2]^T \rightarrow [1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1x_2]^T$.
- The (linear) SVM classifier in the feature space:

$$\hat{y} = \text{sign} \left(\hat{w}_0 + \sum_{\alpha_i > 0} \alpha_i y_i \phi(\mathbf{x}_i)^T \phi(\mathbf{x}) \right)$$

- The dot product in the feature space:

$$\begin{aligned} \phi(\mathbf{x})^T \phi(\mathbf{z}) &= 1 + 2x_1z_1 + 2x_2z_2 + x_1^2z_1^2 + x_2^2z_2^2 + 2x_1x_2z_1z_2 \\ &= (1 + \mathbf{x}^T \mathbf{z})^2. \end{aligned}$$

Dot products and feature space

- We defined a non-linear mapping into feature space

$$\phi : [x_1, x_2]^T \rightarrow [1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1x_2]^T$$

and saw that $\phi(\mathbf{x})^T \phi(\mathbf{z}) = K(\mathbf{x}, \mathbf{z})$ using the *kernel*

$$K(\mathbf{x}, \mathbf{z}) = (1 + \mathbf{x}^T \mathbf{z})^2.$$

- I.e., we can calculate dot products in the feature space implicitly, without ever writing the feature expansion!

The kernel trick

- Replace dot products in the SVM formulation with kernel values.
- The optimization problem:

$$\max \left\{ \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j=1}^N \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \right\}$$

– Need to compute pairwise kernel values for training data.

- The classifier:

$$\hat{y} = \text{sign} \left(\hat{w}_0 + \sum_{\alpha_i > 0} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) \right)$$

– Need to compute $K(\mathbf{x}_i, \mathbf{x})$ for all SVs \mathbf{x}_i .

Mercer's kernels

- What kind of function K is a valid kernel, i.e. such that there exists a feature space $\Phi(\mathbf{x})$ in which $K(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^T \phi(\mathbf{z})$?
- Theorem due to Mercer (1930s): K must be
 - Continuous;
 - symmetric: $K(\mathbf{x}, \mathbf{z}) = K(\mathbf{z}, \mathbf{x})$;
 - positive definite: for any $\mathbf{x}_1, \dots, \mathbf{x}_N$, the *kernel matrix*

$$K = \begin{bmatrix} K(\mathbf{x}_1, \mathbf{x}_1) & K(\mathbf{x}_1, \mathbf{x}_2) & K(\mathbf{x}_1, \mathbf{x}_N) \\ \cdot & \cdot & \cdot \\ K(\mathbf{x}_N, \mathbf{x}_1) & K(\mathbf{x}_N, \mathbf{x}_2) & K(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix}$$

must be positive definite.

Some popular kernels

- The linear kernel:

$$K(\mathbf{x}, \mathbf{z}) = \mathbf{x}^T \mathbf{z}.$$

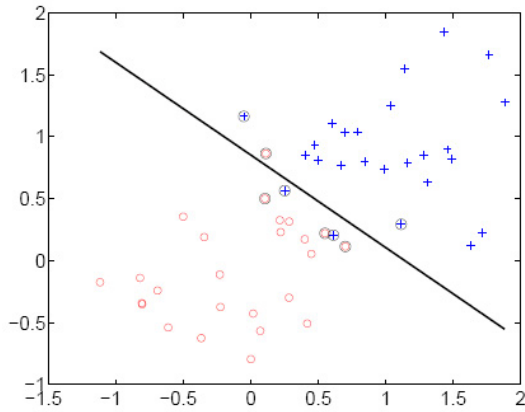
This leads to the original, linear SVM.

- The polynomial kernel:

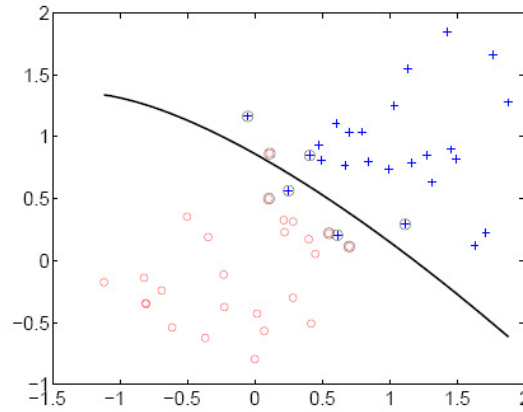
$$K(\mathbf{x}, \mathbf{z}; c, d) = (c + \mathbf{x}^T \mathbf{z})^d.$$

We can write the expansion explicitly, by concatenating powers up to d and multiplying by appropriate weights.

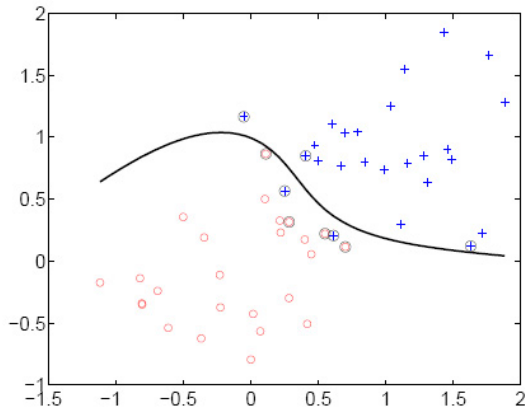
Example: SVM with polynomial kernel



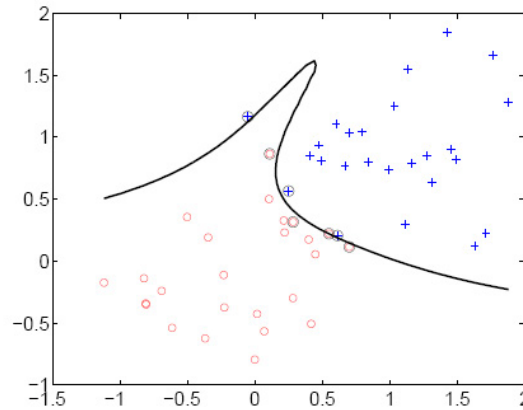
linear



2nd order polynomial



4th order polynomial



8th order polynomial

(using $C < \infty$)

Compare to the effect of model order in regression or logistic regression.

Radial basis function kernel

$$K(\mathbf{x}, \mathbf{z}; \sigma) = \exp\left(-\frac{1}{\sigma^2} \|\mathbf{x} - \mathbf{z}\|^2\right).$$

- The RBF kernel is a measure of similarity between two examples.
 - The feature space is infinite-dimensional!
- What is the role of parameter σ ?

Radial basis function kernel

$$K(\mathbf{x}, \mathbf{z}; \sigma) = \exp\left(-\frac{1}{\sigma^2} \|\mathbf{x} - \mathbf{z}\|^2\right).$$

- The RBF kernel is a measure of similarity between two examples.
 - The feature space is infinite-dimensional!
- What is the role of parameter σ ? Consider $\sigma \rightarrow 0$.

Radial basis function kernel

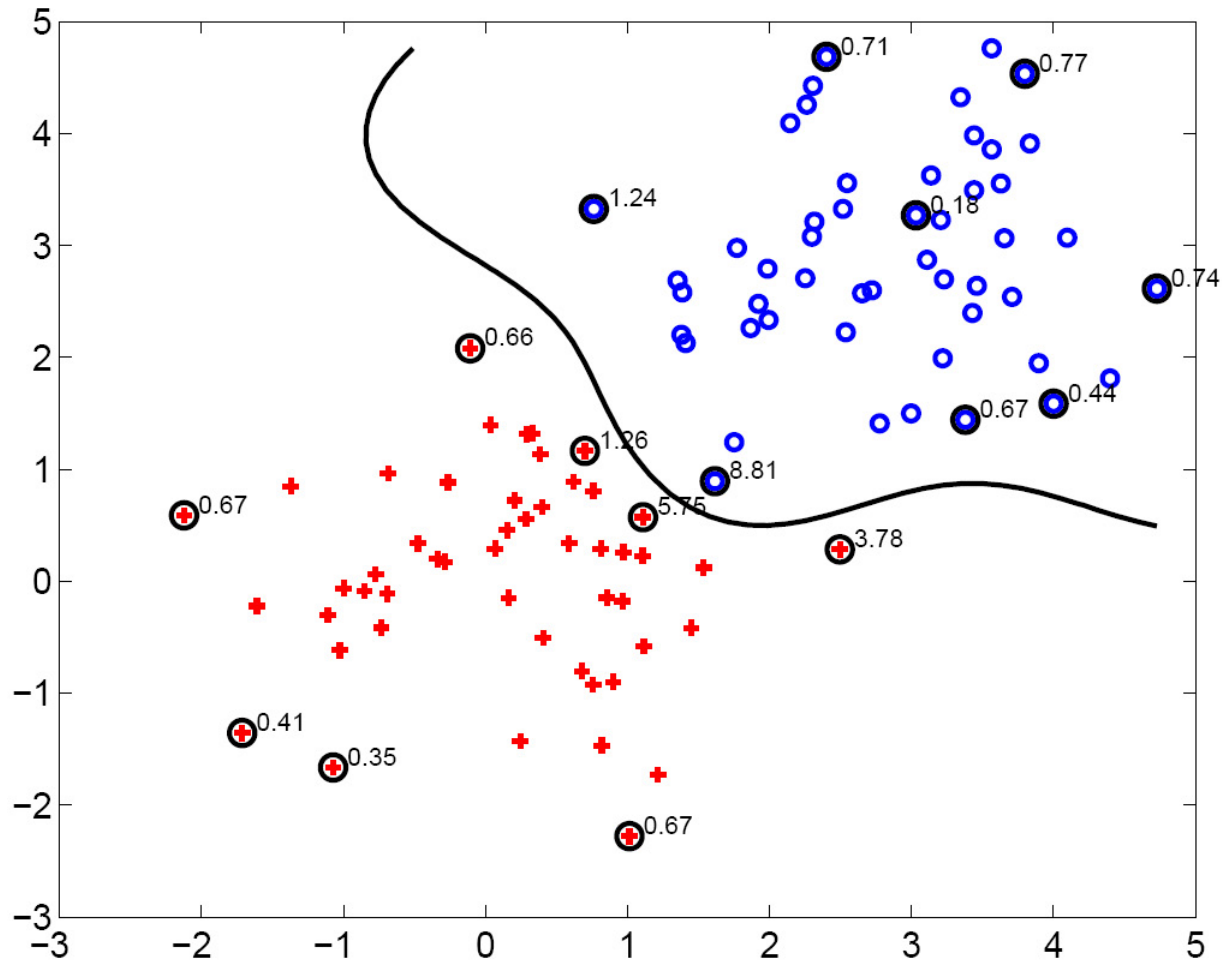
$$K(\mathbf{x}, \mathbf{z}; \sigma) = \exp\left(-\frac{1}{\sigma^2}\|\mathbf{x} - \mathbf{z}\|^2\right).$$

- The RBF kernel is a measure of similarity between two examples.
 - The feature space is infinite-dimensional!
- What is the role of parameter σ ? Consider $\sigma \rightarrow 0$.

$$K(\mathbf{x}_i, \mathbf{x}; \sigma) \rightarrow \begin{cases} 1 & \text{if } \mathbf{x} = \mathbf{x}_i, \\ 0 & \text{if } \mathbf{x} \neq \mathbf{x}_i. \end{cases}$$

- All examples become SVs \Rightarrow likely overfitting.

SVM with RBF (Gaussian) kernels



- Why are some SV here not close to the boundary?..

SVM: summary

- Two main ideas:
 - large margin classification,
 - the kernel trick.
- Complexity of classifier depends on the number of SVs.
 - Controlled indirectly by C and kernel parameters.
- One of the most successful ML techniques!
- A crucial component: good QP solver.
- Recommended off-the-shelf package: SVM^{light}
<http://svmlight.joachims.org>

Next time

More on the use of kernels in machine learning.