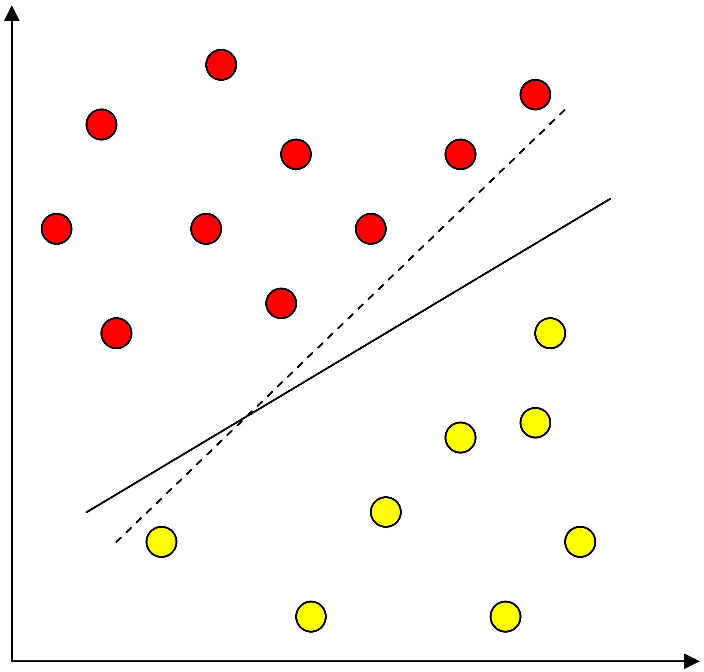
Support Vector Machines

Adrian Barbu

Decision Boundaries

- Binary classification
  - Labels 1 and -1
- Say classes are linearly separable
  - Logistic Regression
  - Generative classifiers



- Are all decision boundaries equally good?
- Which one is better?
- Which one is best?

Decision Boundary

- Boundary equation

$$w^T x + w_0 = 0$$

- w is the normal to the decision boundary
  - For class 1 (red):

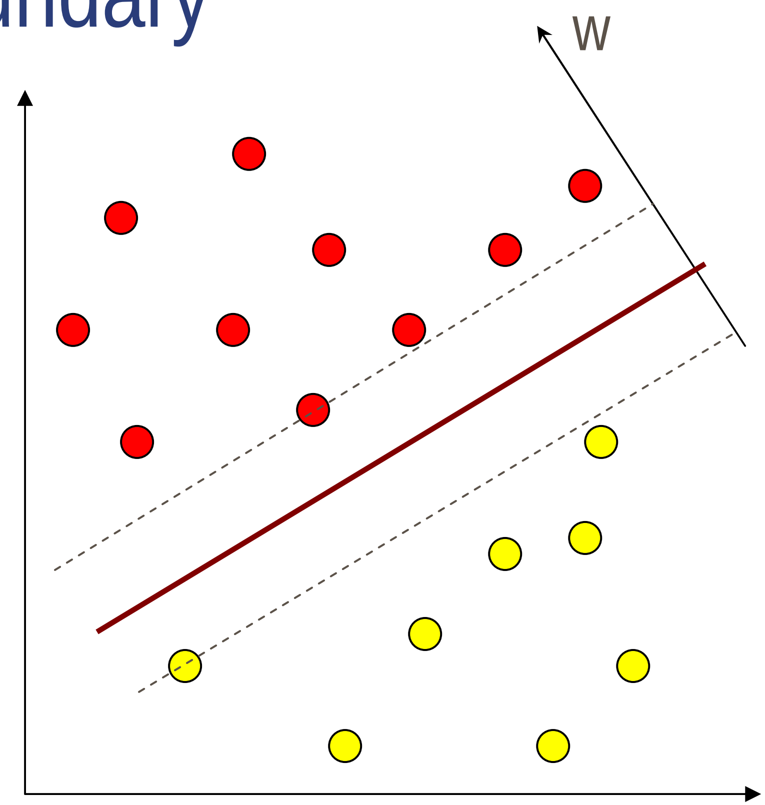
$$w^T x + w_0 > 0$$

  - For class -1(yellow)

$$w^T x + w_0 < 0$$

  - For all observations (x_i,y_i)

$$(w^T x_i + w_0) y_i > 0$$



Decision Boundary and Margin

- Distance to the boundary of an observation (x_i, y_i)

  $$m_i = (w^T x_i + w_0) y_i > 0$$

  if $||w|| = 1$

- Define margin as m such that

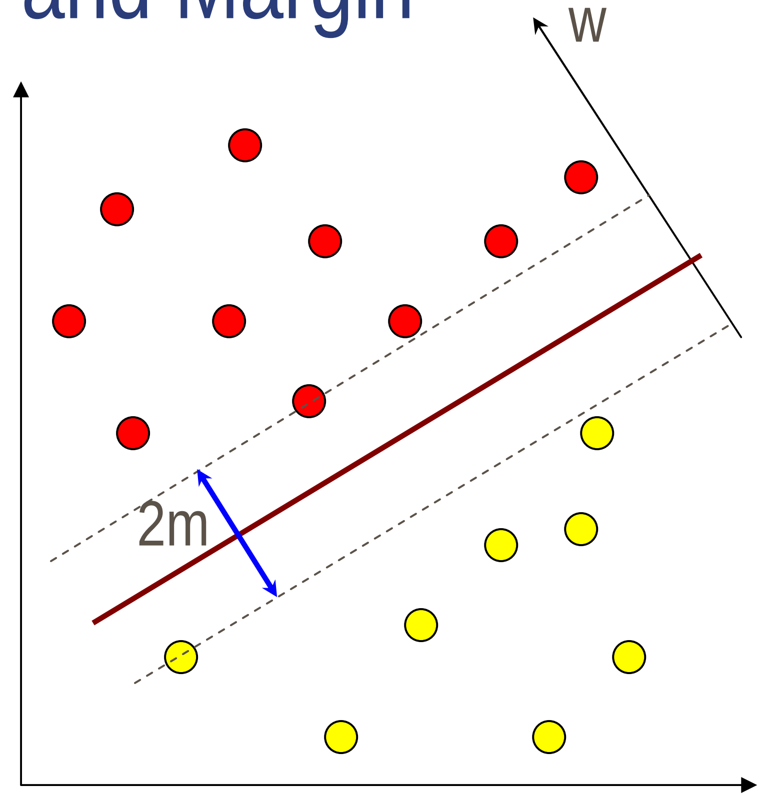
  $$\underline{(w^T x_i + w_0) y_i \geq m,}\ \forall i$$

  $$\underline{||w|| = 1}$$

- Want maximum margin

  $$\max_{w, w_0, ||w||=1} \quad m$$

  $$\text{s.t.} \quad (w^T x_i + w_0) y_i \geq m, \ \forall i$$



Maximum Margin Classifier

- Equivalently

$$\max_{w,w_0} \frac{m}{||w||}$$

$$\text{s.t.} \;\; (w^T x_i + w_0) y_i \geq m, \; \forall i$$

- Divide w and w_0 by m (same decision boundary)

$$\max_{w,w_0} \frac{1}{||w||}$$

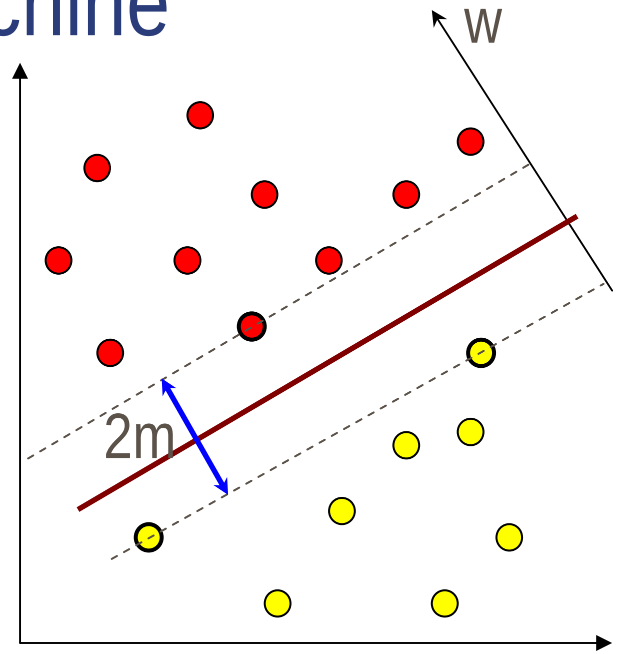
$$\text{s.t.} \;\; (w^T x_i + w_0) y_i \geq 1, \; \forall i$$

- Equivalently

$$\min_{w,w_0} w^T w$$

$$\text{s.t.} \;\; (w^T x_i + w_0) y_i \geq 1, \; \forall i$$

Support Vector Machine



- Convex quadratic programming:

$$\min_{w,w_0} w^T w$$

$$\text{s.t. } (w^T x_i + w_0) y_i \geq 1, \ \forall i$$

  - Linear constraints
  - Margin is $m = \frac{1}{|w|}$
  - Only a few of the constraints are relevant → support vectors
- Constrained Optimization:
  - Can use a generic QP Optimization packages
  - Using **Lagrange duality**
    - More efficient optimization
    - Generalization: Kernel SVM

Lagrange Multipliers

- Consider the Primal optimization problem:

$$\min_{w} f(w)$$
$$\text{s.t. } g_i(w) \leq 0,\ i = 1, \ldots, k$$
$$h_j(w) = 0,\ j = 1, \ldots, l$$

- The Generalized Lagrangian

$$\mathcal{L}(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{j=1}^{l} \beta_j h_j(w)$$

the $\alpha_i \geq 0$ and β_j are the Lagrange Multipliers

- Lemma:

$$\max_{\alpha,\beta,\alpha_i \geq 0} \mathcal{L}(w, \alpha, \beta) = \begin{cases} f(w) & \text{if } g_i(w) \leq 0\ \forall i,\ h_j(w) = 0\ \forall j \\ \infty & \text{else} \end{cases}$$

Lagrange Duality

- We can reformulate the Primal Problem

$$\min_{w} \max_{\alpha,\beta,\alpha_i>0} \mathcal{L}(w,\alpha,\beta)$$

- The Dual Problem

$$\max_{\alpha,\beta,\alpha_i>0} \min_{w} \mathcal{L}(w,\alpha,\beta)$$

- Theorem 1 (weak duality)

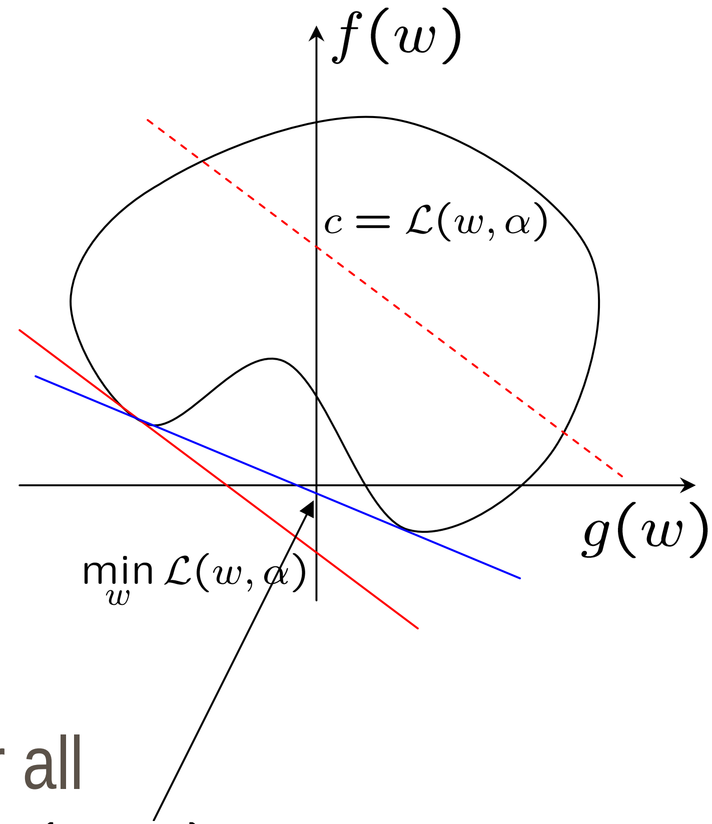
$$\max_{\alpha,\beta,\alpha_i>0} \min_{w} \mathcal{L}(w,\alpha,\beta) \leq \min_{w} \max_{\alpha,\beta,\alpha_i>0} \mathcal{L}(w,\alpha,\beta)$$

- Theorem 2 (strong duality)
  Equality holds if and only if $\mathcal{L}(w,\alpha,\beta)$ has a saddle point.

Example

- One dimensional w, no β: $\mathcal{L}(w,\alpha) = f(w) + \alpha g(w)$

$$\max_{\alpha \geq 0} \min_{w} \mathcal{L}(w,\alpha)$$



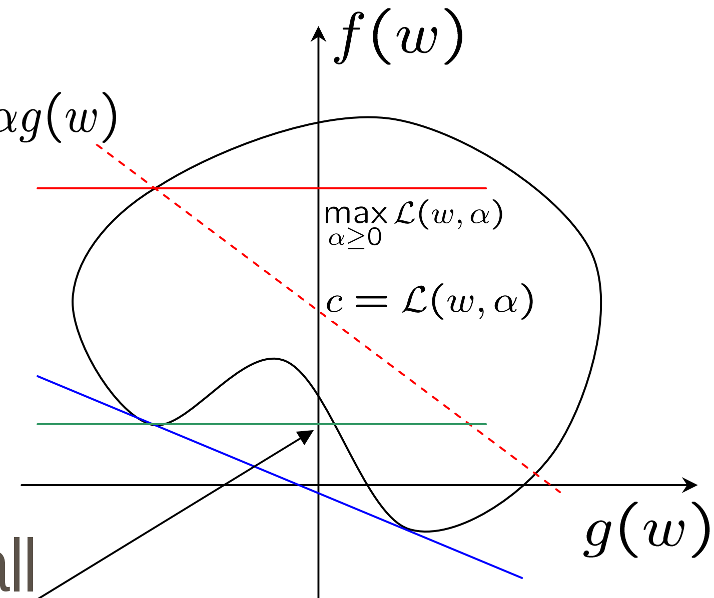
1. Fix α
   - Consider lines of slope -α
     $$y = -\alpha x + c$$
     if line passes through (g(w),f(w)) then intercept is $c = \mathcal{L}(w,\alpha)$
   - Push line down as much as possible
     Obtain $\min_{w} \mathcal{L}(w,\alpha)$
2. Find maximum of the intercept over all downward lines. Obtain $\max_{\alpha \geq 0} \min_{w} \mathcal{L}(w,\alpha)$

Example

$$\min_{w} \max_{\alpha \geq 0} \mathcal{L}(w, \alpha)$$

1. Fix w
   - Find α to maximize $\mathcal{L}(w, \alpha) = f(w) + \alpha g(w)$
     - If g(w)>0, $\max_{\alpha \geq 0} \mathcal{L}(w, \alpha) = \infty$
     - If g(w)<0, α=0
   - Obtain horizontal lines with intercept

     $$\max_{\alpha \geq 0} \mathcal{L}(w, \alpha)$$



2. Find minimum of the intercept over all horizontal lines. Obtain

$$\min_{w} \max_{\alpha \geq 0} \mathcal{L}(w, \alpha)$$

Observe $$\max_{\alpha \geq 0} \min_{w} \mathcal{L}(w, \alpha) \leq \min_{w} \max_{\alpha \geq 0} \mathcal{L}(w, \alpha)$$

The KKT Conditions

- Equality=saddle point
- Saddle point must satisfy the "Karush-Kuhn-Tucker" (KKT) conditions:

$$
\begin{aligned}
\frac{\partial}{\partial w_i}\mathcal{L}(w,\alpha,\beta) &= 0, i = 1, \ldots, M \\
\frac{\partial}{\partial \beta_j}\mathcal{L}(w,\alpha,\beta) &= 0, j = 1, \ldots, l \\
\alpha_i g_i(w) &= 0, i = 1, \ldots, k \\
g_i(w) &\le 0, i = 1, \ldots, k \\
\alpha_i &\ge 0, i = 1, \ldots, k
\end{aligned}
$$

- Theorem: If (w,α,β) satisfies the KKT conditions then it is a solution to the primal and dual problem.

Back to SVM

- SVM: $$\min_{w,w_0} w^T w$$ $$\text{s.t. } (w^T x_i + w_0) y_i \geq 1, \ \forall i$$

  Equivalently: $$\min_{w,w_0} \frac{1}{2} w^T w \qquad (1)$$ $$\text{s.t. } 1 - (w^T x_i + w_0) y_i \leq 0, \ \forall i$$

- The Lagrangian is:

$$\mathcal{L}(w, w_0, \alpha) = \frac{1}{2} w^T w - \sum_{i=1}^{N} \alpha_i [(w^T x_i + w_0) y_i - 1]$$

- The primal problem (1):

$$\min_{w,w_0} \max_{\alpha, \alpha_i > 0} \mathcal{L}(w, w_0, \alpha)$$

The Dual Problem

$$\max_{\alpha,\alpha_i>0} \min_{w,w_0} \mathcal{L}(w, w_0, \alpha)$$

- Minimize w.r.t. w and w_0

$$\frac{\partial}{\partial w}\mathcal{L}(w, w_0, \alpha) = w - \sum_{i=1}^{N} \alpha_i y_i x_i = 0$$

$$\frac{\partial}{\partial w_0}\mathcal{L}(w, w_0, \alpha) = \sum_{i=1}^{N} \alpha_i y_i = 0$$

Obtain $w = \sum_{i=1}^{N} \alpha_i y_i x_i$

- Plugging back in obtain

$$\mathcal{L}(w, w_0, \alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j (x_i^T x_j)$$

The Dual Problem

- We obtain the simplified dual problem:

$$\max_{\alpha, \alpha_i \geq 0} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j (x_i^T x_j)$$

$$\text{s.t.} \quad \sum_{i=1}^{N} \alpha_i y_i = 0$$

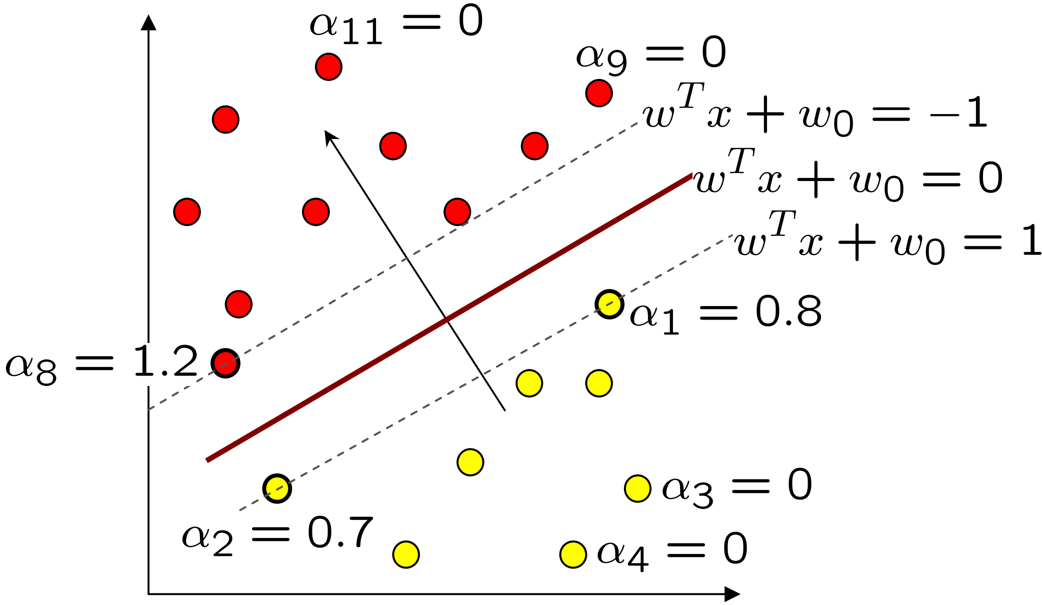
- Quadratic Optimization
  - Simpler, only in α, with one constrain
  - Can recover $w = \sum_{i=1}^{N} \alpha_i y_i x_i$
  - The kernel $x_i^T x_j$ can be generalized to nonlinear classification

Support Vectors

- KKT conditions:

$$\alpha_i[(w^T x_i + w_0)y_i - 1] = 0$$

- Points (x_i, y_i) for which $\alpha_i > 0$ are the support vectors (SV)



Support Vector Machines

- The weights can be obtained from the Lagrange multipliers

$$w = \sum_{i \in SV} \alpha_i y_i x_i$$

  - Linear combination of a small number of data points
  - Only the important data is memorized
- Given a new feature vector z obtain the classification

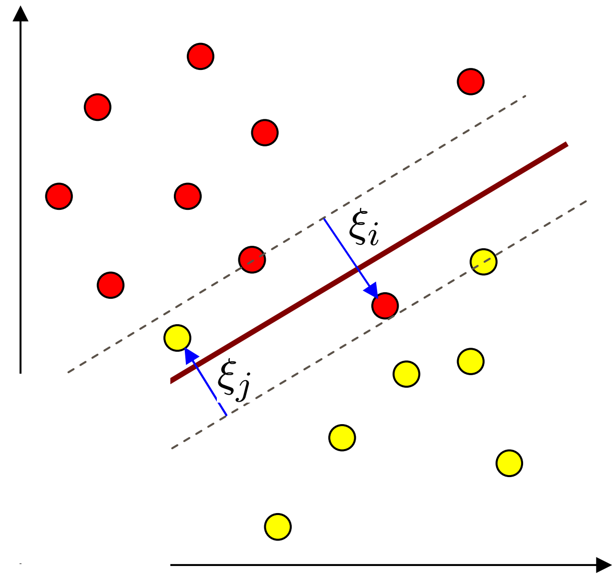
$$\hat{y} = \mathrm{sgn}\Big(\sum_{i \in SV} \alpha_i y_i (x_i^T z) + w_0 \Big)$$

  - Can be faster than computing w (if the number of SV is small)
  - w_0 can be used to change the detection rate-false alarm tradeoff

Non-Separable Problems

- Pay penalty for misclassified cases

$$\xi_i = 1 - y_i(w^T x_i + w_0) > 0$$



- The Soft Margin SVM

$$\min_{\xi_i \geq 0, w, w_0} \frac{1}{2} w^T w + C \sum_{i=1}^{N} \xi_i$$

$$\text{s.t. } 1 - (w^T x_i + w_0) y_i \leq \xi_i, \ \forall i$$

  - Slack variables ξ_i
  - C = tradeoff parameter

Dual Problem for Soft Margin SVM

- Repeat the same tricks.
- Obtain

$$\max_{\alpha} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j (x_i^T x_j)$$

$$\text{s.t. } \; 0 \leq \alpha_i \leq C, \forall i$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0$$

- Observe upper bound on α_i
- Use a QP package to solve it

Non-Linear Decision Boundary

Key idea:

- Map x to a larger feature space where it becomes linear
  - Use $\phi(x)$ instead of x as the feature vector
  - Most non-linear problems can be made linear in a larger space

- But
  - The new feature space can be very large, e.g. 10^{20} dimensional
  - More features, more computationally expensive

- But
  - The Kernel trick avoids the computational problem

The Kernel Trick

- SVM: $$\max_{\alpha} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j (x_i^T x_j)$$
  $$\text{s.t. } 0 \le \alpha_i \le C, \forall i$$
  $$\sum_{i=1}^{N} \alpha_i y_i = 0$$
- Only depends on the products $x_i^T x_j$
- With transformation, becomes $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$
  This is the Kernel K
- Often can be computed without transformation
- Classification
  $$\hat{y} = \text{sgn}\Big(\sum_{i \in SV} \alpha_i y_i K(x_i, z) + w_0 \Big)$$

Example

- Say $x = (x_1, x_2)^T$
- Take the transformation

$$\phi(x) = (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1 x_2)^T$$

- Using the usual inner product, obtain

$$K(x, x') = \phi(x)^T \phi(x') = (1 + x^T x')^2$$

- This can generalize in any dimension

Example of Kernel Functions

- Linear Kernel:

$$K(x, x') = x^T x'$$

- Polynomial Kernel

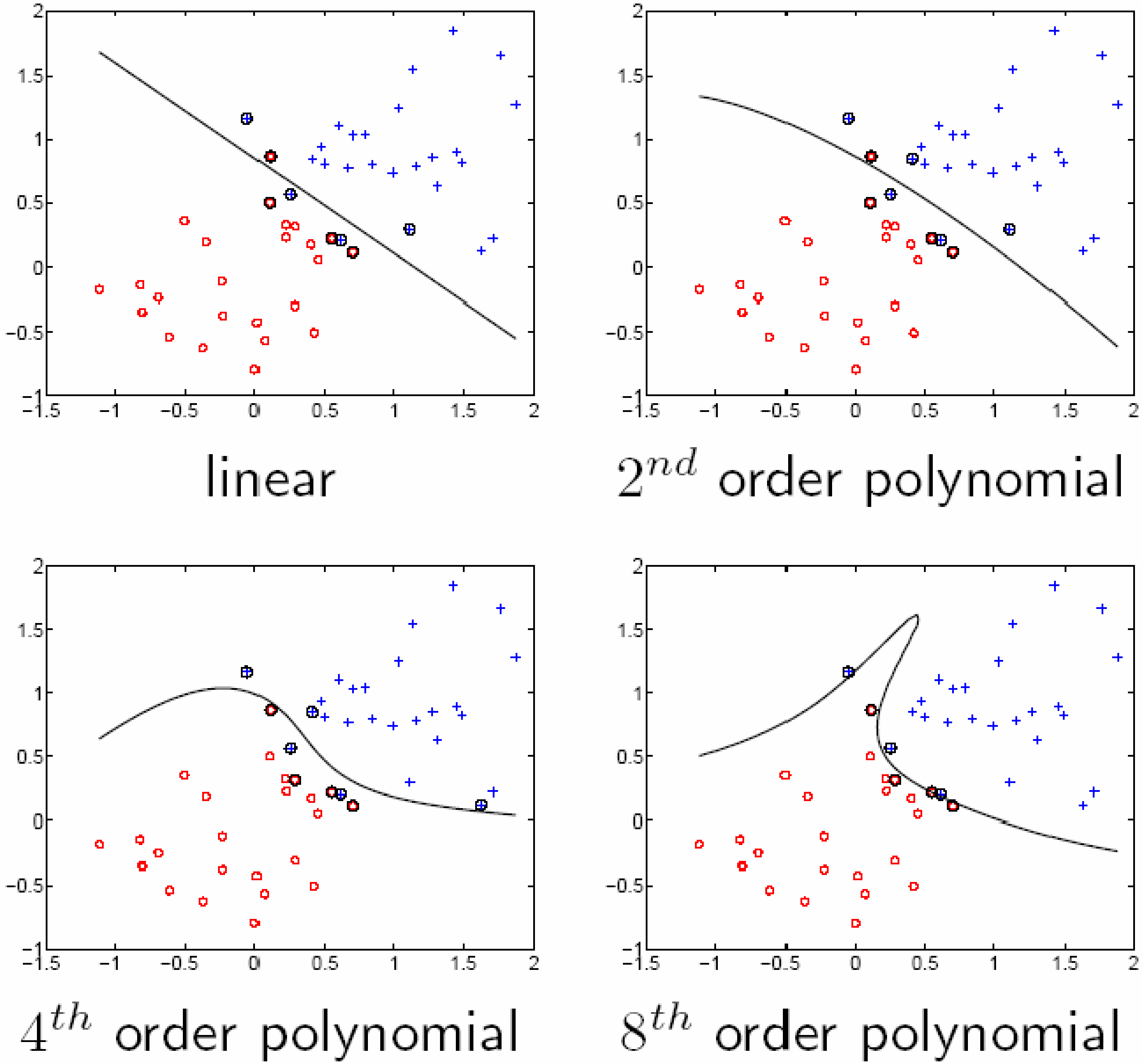
$$K(x, x') = (1 + x^T x')^p, p = 2, 3, \ldots$$

  - The transformed feature vector contains all monomials up to degree p, with appropriate weights.
- Radial Basis (Gaussian) Kernel

$$K(x, x') = \exp(-\frac{1}{R^2}||x - x'||^2), R > 0$$

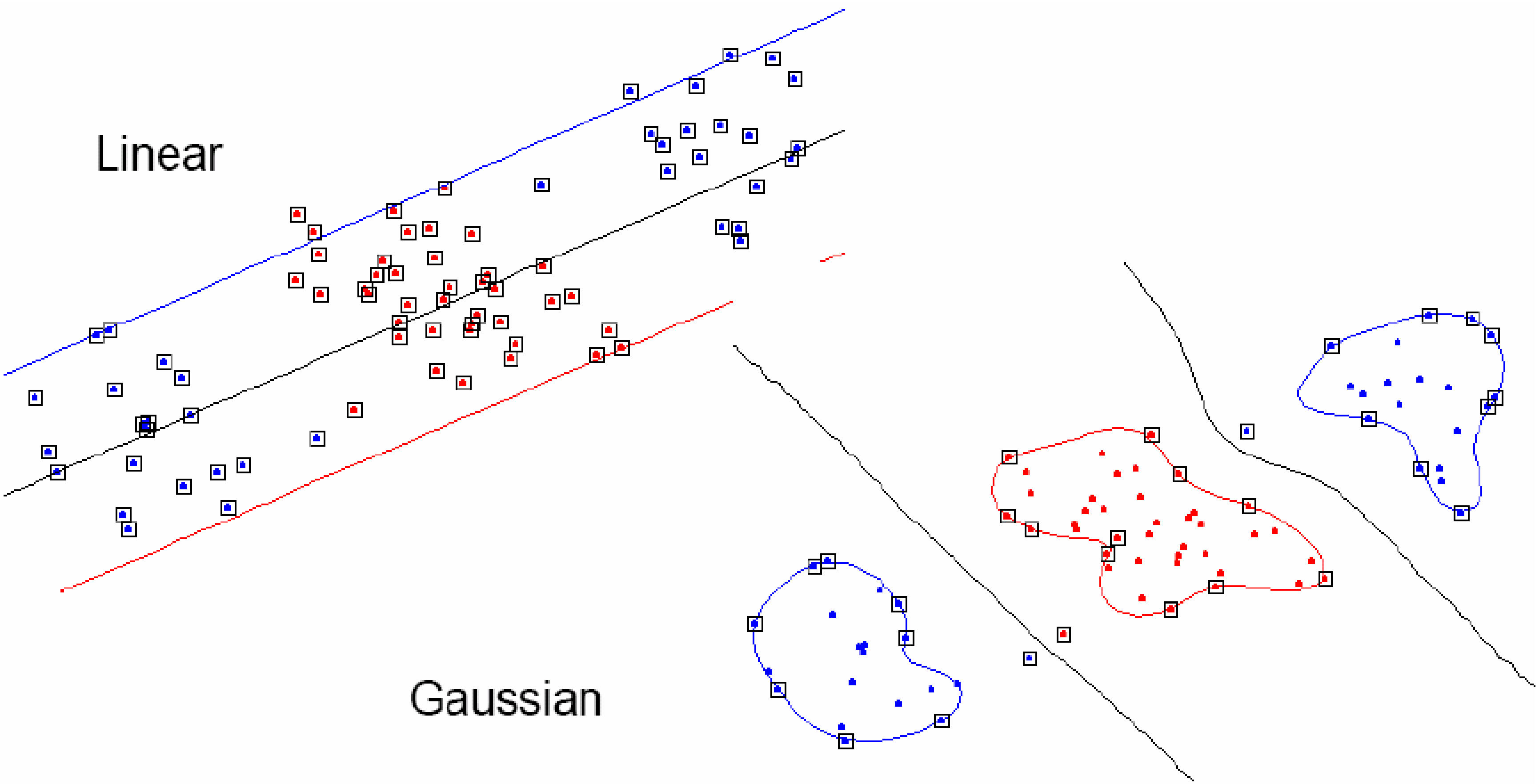
  - Larger R, smoother decision boundary

Examples



linear
2^{nd} order polynomial
4^{th} order polynomial
8^{th} order polynomial

Examples



- Must select appropriate R to avoid overfitting.

Cross-Validation Error

- The Leave-One-Out cross-validation depends only on the number of support vectors:

$$\text{Leave-One-Out CV Error} = \frac{|SV|}{N}$$

- Want a small number of SVs
- Tune parameters (p,R,C) to minimize |SV|.