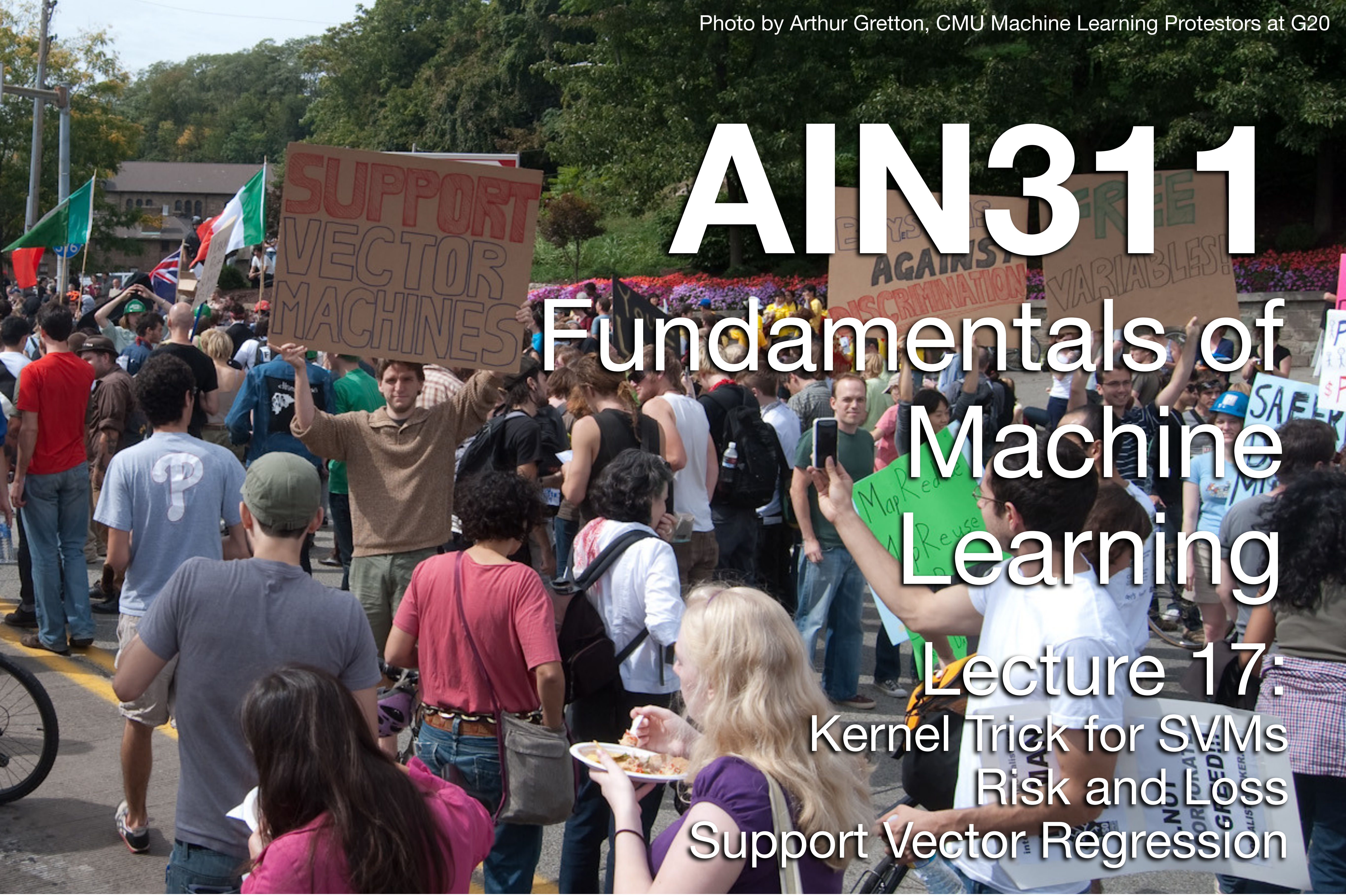


Photo by Arthur Gretton, CMU Machine Learning Protestors at G20

AIN311

Fundamentals of Machine Learning

Lecture 17:

Kernel Trick for SVMs

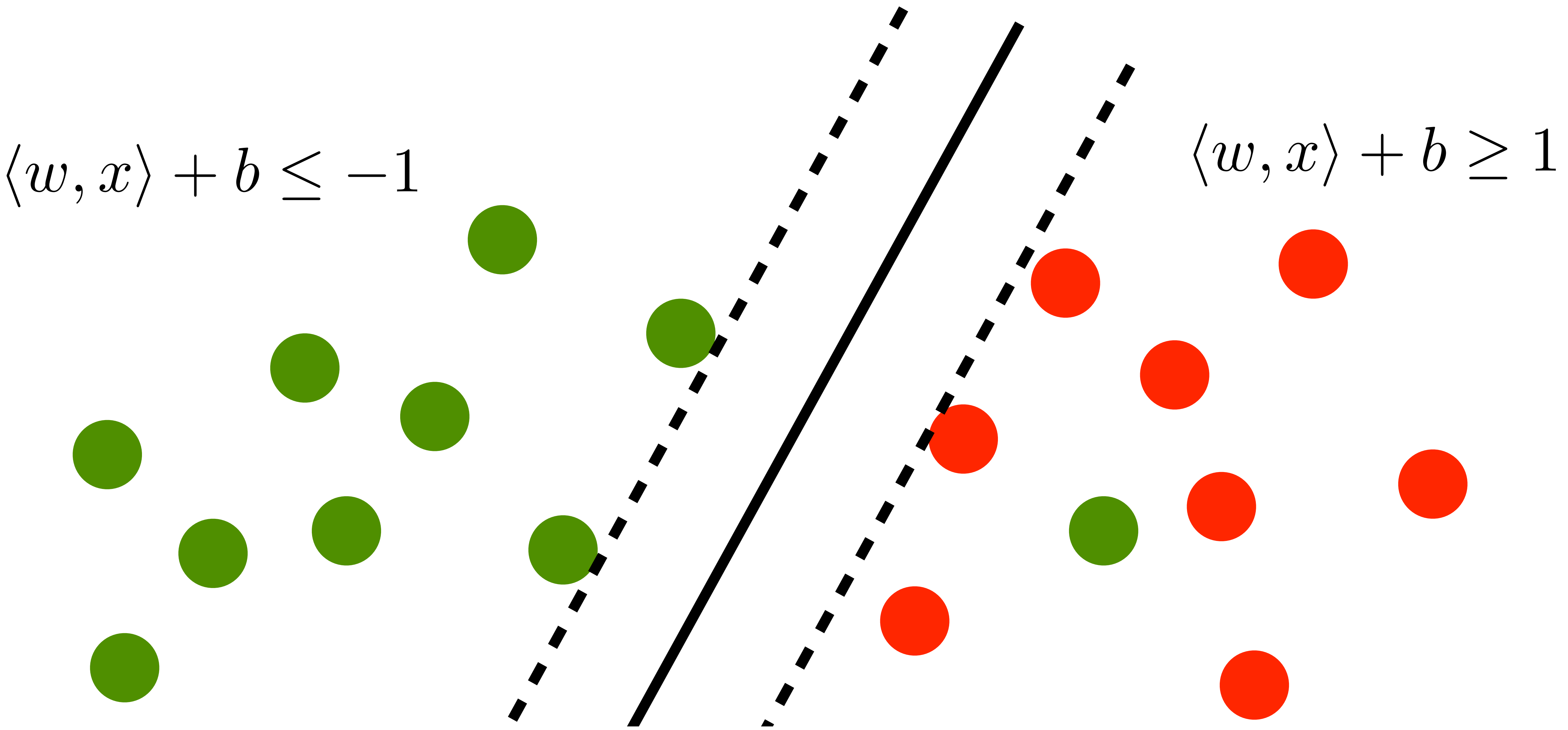
Risk and Loss

Support Vector Regression

HACETTEPE
UNIVERSITY
COMPUTER
VISION LAB

Erkut Erdem // Hacettepe University // Fall 2024

Last time... **Soft-margin Classifier**


$\langle w, x\rangle + b \leq -1$
$\langle w, x\rangle + b \geq 1$

minimum error separator
is impossible

Theorem (Minsky & Papert)
Finding the minimum error separating hyperplane is NP hard

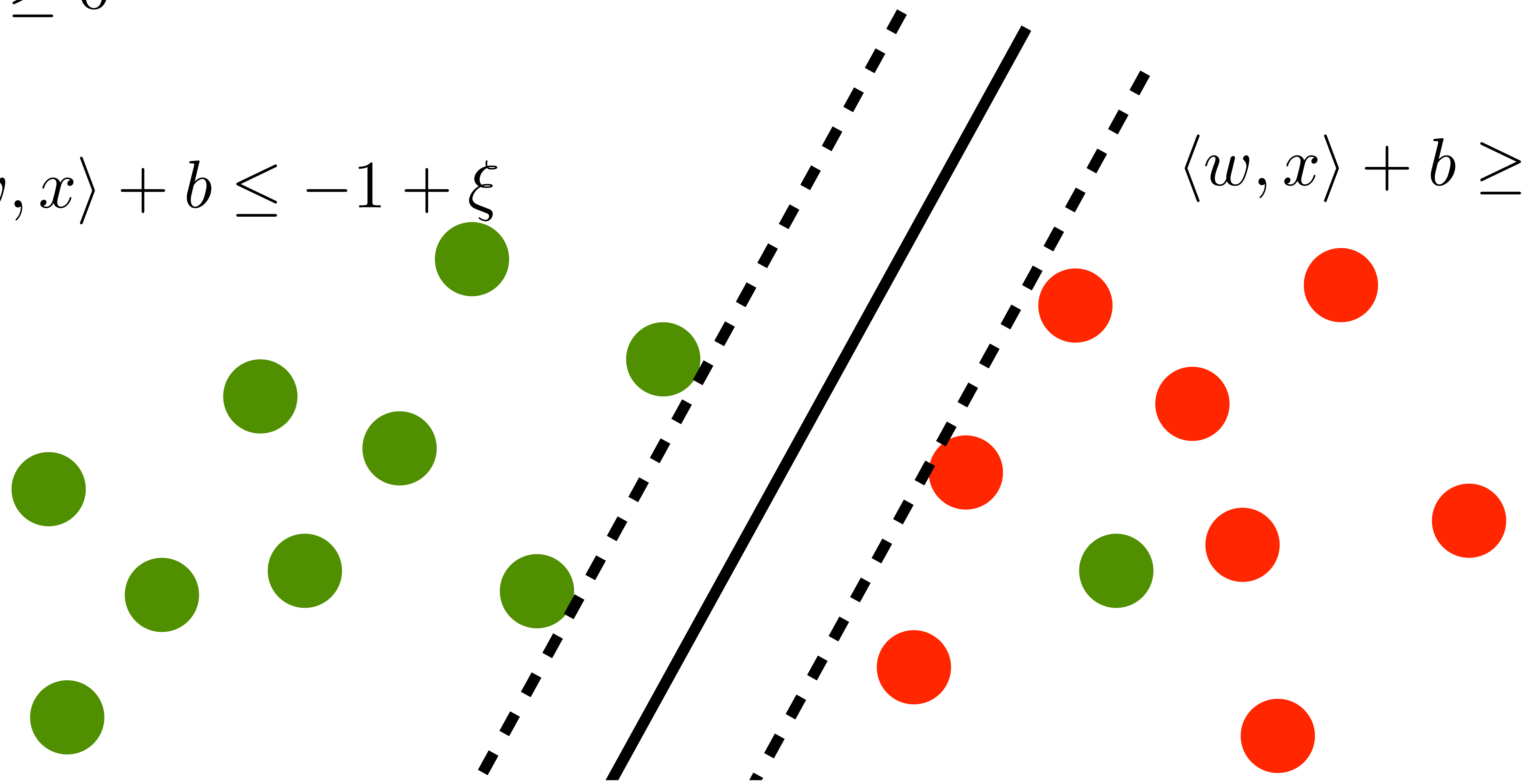
slide by Alex Smola

Last time... **Adding Slack Variables**

$\xi_i \geq 0$

$\langle w, x \rangle + b \leq -1 + \xi$

$\langle w, x \rangle + b \geq 1 - \xi$



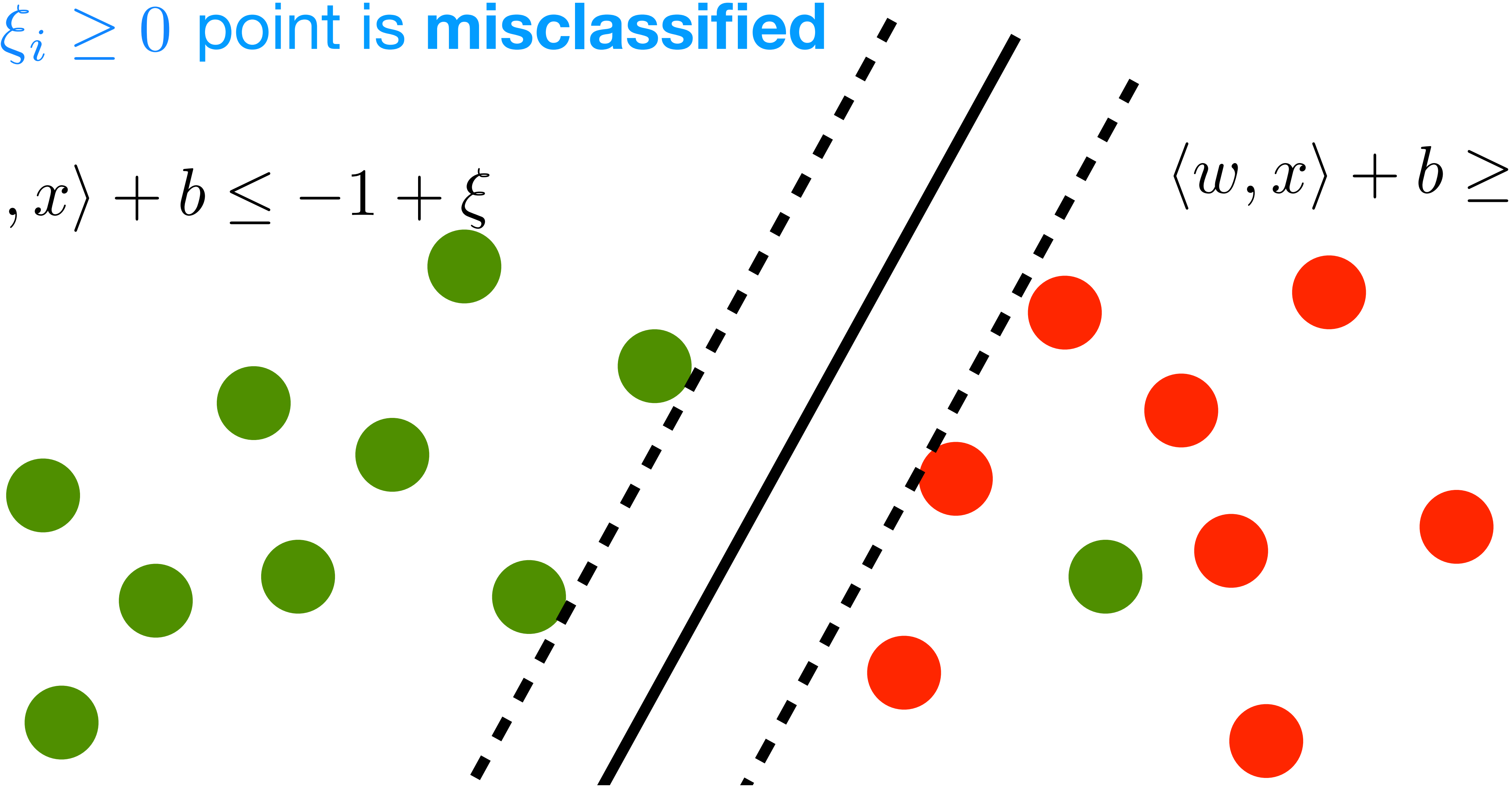
minimize amount of slack

Convex optimization problem

slide by Alex Smola

Last time... **Adding Slack Variables**

- for $0 < \xi \leq 1$ point is between the margin and **correctly classified**
- for $\xi_i \geq 0$ point is **misclassified**



$\langle w, x \rangle + b \leq -1 + \xi$

$\langle w, x \rangle + b \geq 1 - \xi$

minimize amount of slack

Convex optimization problem

adopted from Andrew Zisserman

Last time… **Adding Slack Variables**

- Hard margin problem

$$\underset{w,b}{\text{minimize}}\, \frac{1}{2}\left\|w\right\|^2 \text{ subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1$$

- With slack variables

$$\underset{w,b}{\text{minimize}}\;\; \frac{1}{2}\left\|w\right\|^2 + C\sum_i \xi_i$$

$$\text{subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

Problem is always feasible. Proof:

$w = 0 \text{ and } b = 0 \text{ and } \xi_i = 1$ (also yields upper bound)

slide by Alex Smola

Soft-margin classifier

- Optimization problem:

$$\underset{w,b}{\text{minimize}} \quad \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

$$\text{subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

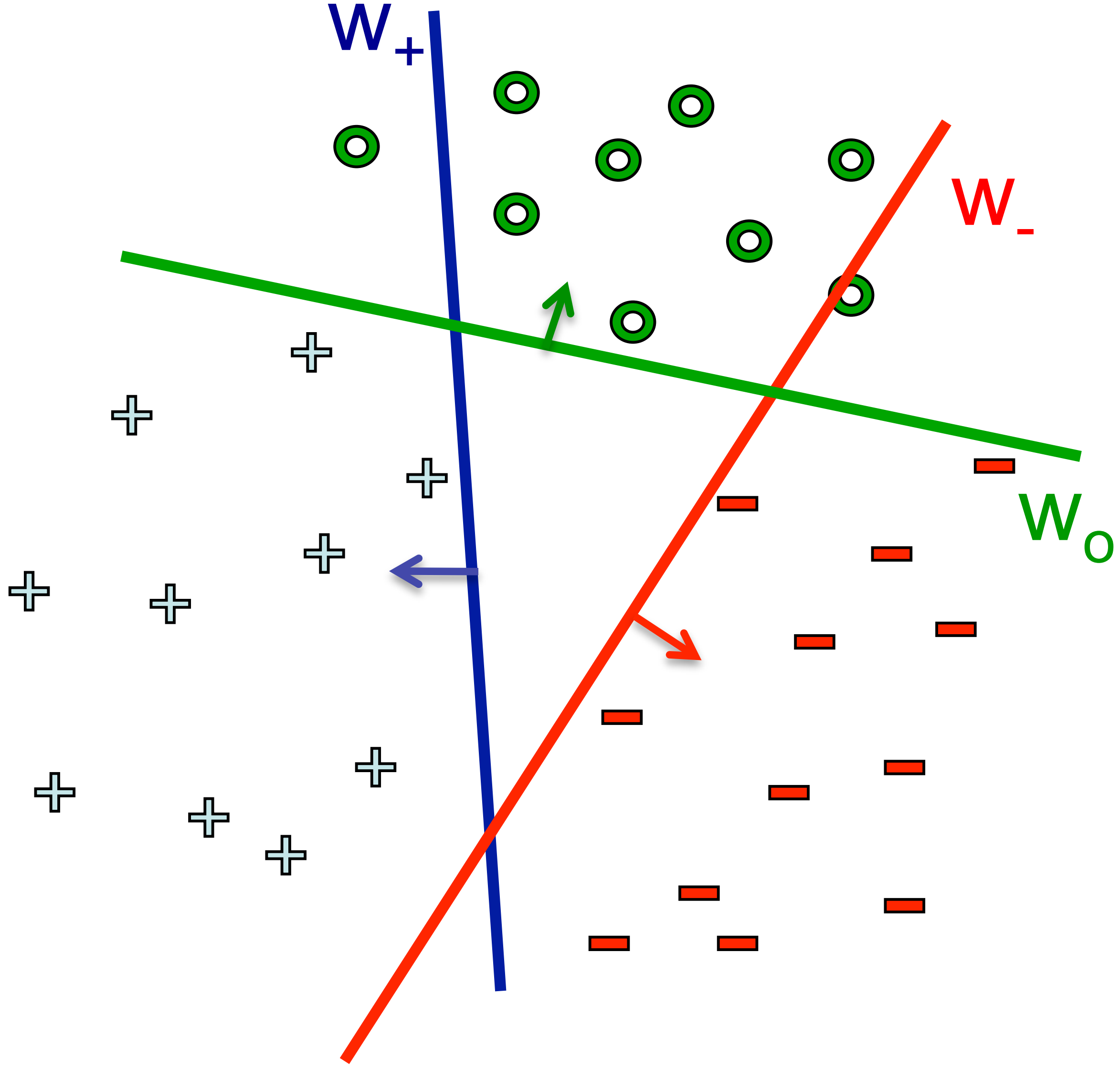
C is a **regularization** parameter:

- small *C* allows constraints to be easily ignored
  → **large margin**
- large *C* makes constraints hard to ignore
  → **narrow margin**
- $C = \infty$ enforces all constraints: **hard margin**

adopted from Andrew Zisserman

Last time… Multi-class SVM

- Simultaneously learn 3 sets of weights:
- How do we guarantee the correct labels?
- Need new constraints!



The "score" of the correct class must be better than the "score" of wrong classes:

$$w^{(y_j)} \cdot x_j + b^{(y_j)} > w^{(y)} \cdot x_j + b^{(y)} \qquad \forall j,\ y \neq y_j$$

slide by Eric Xing

Last time... **Multi-class SVM**

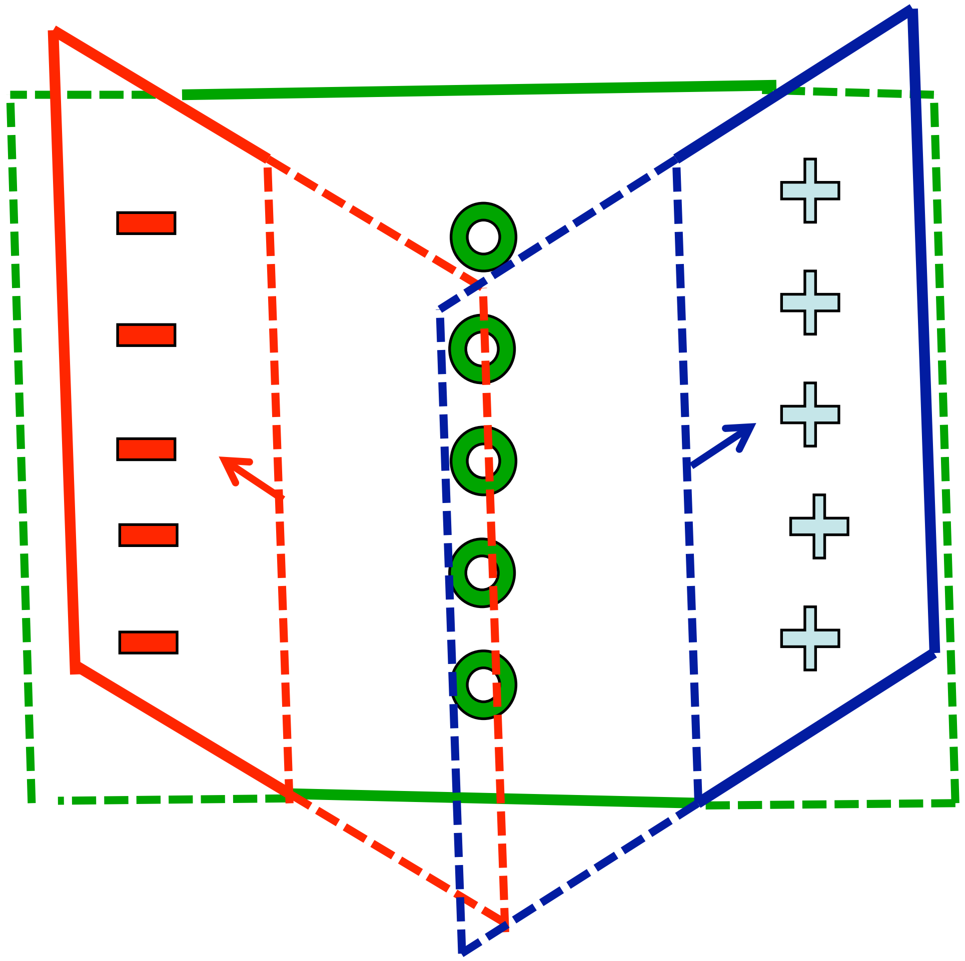
- As for the SVM, we introduce slack variables and maximize margin:

$$
\begin{aligned}
&\text{minimize}_{\mathbf{w},b} \quad \textstyle\sum_y \mathbf{w}^{(y)}.\mathbf{w}^{(y)} + C\sum_j \xi_j \\
&\mathbf{w}^{(y_j)}.\mathbf{x}_j + b^{(y_j)} \geq \mathbf{w}^{(y')}.\mathbf{x}_j + b^{(y')} + 1 - \xi_j, \ \forall y' \neq y_j, \ \forall j \\
&\hfill \xi_j \geq 0, \ \forall j
\end{aligned}
$$

To predict, we use:

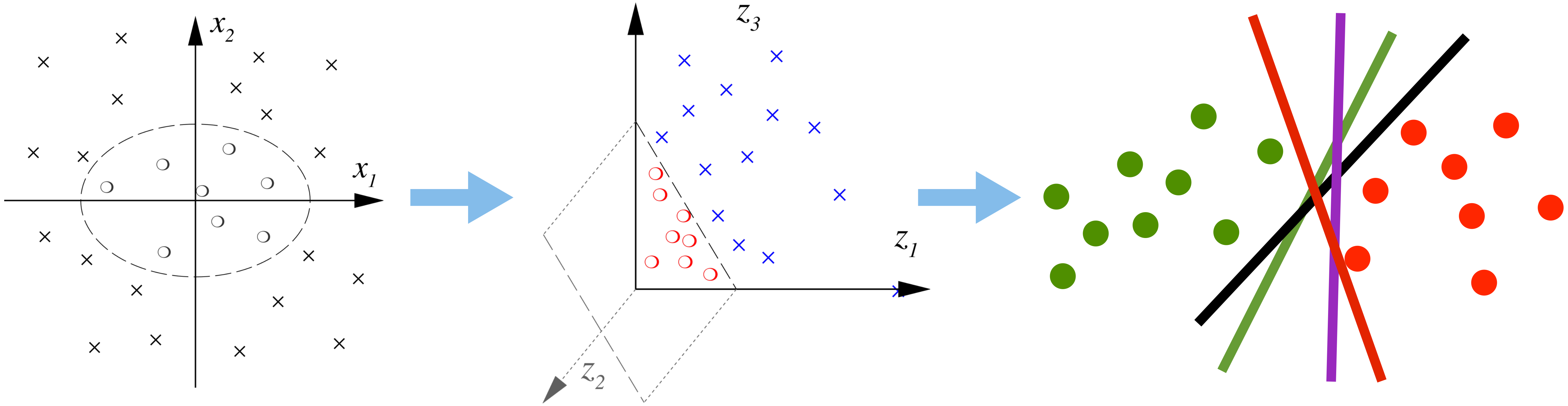
$$\hat{y} \leftarrow \arg\max_k \ w_k \cdot x + b_k$$

Now can we learn it? →



slide by Eric Xing

Last time… Kernels



- Original data
- Data in feature space (implicit)
- Solve in feature space using kernels

slide by Alex Smola

Last time… **Quadratic Features**

Quadratic Features in $\mathbb{R}^2$

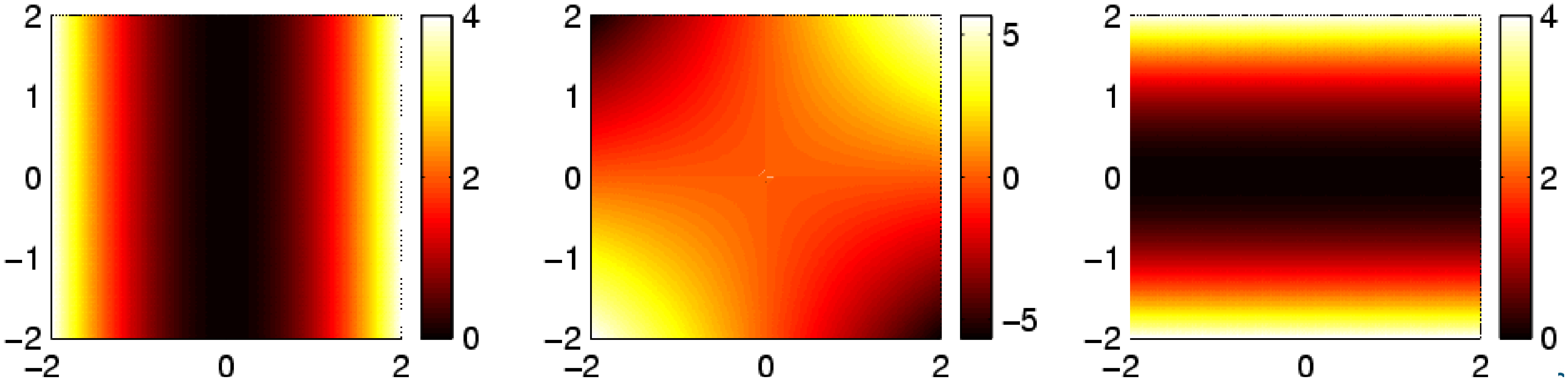
$$\Phi(x) := \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right)$$

Dot Product

$$\begin{aligned}\langle \Phi(x), \Phi(x')\rangle &= \left\langle \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right), \left({x'_1}^2, \sqrt{2}x'_1x'_2, {x'_2}^2\right)\right\rangle \\ &= \langle x, x'\rangle^2.\end{aligned}$$

Insight

Trick works for any polynomials of order via $\langle x, x'\rangle^d$.



slide by Alex Smola

Last time.. **Computational Efficiency**

Problem

- Extracting features can sometimes be very costly.
- Example: second order features in 1000 dimensions. This leads to $5 \cdot 10^5$ numbers. For higher order polynomial features much worse.

Solution

Don't compute the features, try to compute dot products implicitly. For some features this works ...

Definition

A kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a symmetric function in its arguments for which the following property holds

$$k(x, x') = \langle \Phi(x), \Phi(x') \rangle \text{ for some feature map } \Phi.$$

If $k(x, x')$ is much cheaper to compute than $\Phi(x)$ ...

slide by Alex Smola

Last time.. **Example kernels**

Examples of kernels $k(x, x')$

Linear	$\langle x, x' \rangle$
Laplacian RBF	$\exp\left(-\lambda \lVert x - x' \rVert\right)$
Gaussian RBF	$\exp\left(-\lambda \lVert x - x' \rVert^2\right)$
Polynomial	$\left(\langle x, x' \rangle + c \rangle\right)^d, c \geq 0,\ d \in \mathbb{N}$
B-Spline	$B_{2n+1}(x - x')$
Cond. Expectation	$\mathbf{E}_c[p(x \vert c) p(x' \vert c)]$

Simple trick for checking Mercer's condition

Compute the Fourier transform of the kernel and check that it is nonnegative.

slide by Alex Smola

Today

- The Kernel Trick for SVMs
- Risk and Loss
- Support Vector Regression

The Kernel Trick for SVMs

slide by Alex Smola

The Kernel Trick for SVMs

- Linear soft margin problem

$$\underset{w,b}{\text{minimize}} \;\; \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

$$\text{subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

- Dual problem

$$\underset{\alpha}{\text{maximize}} - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \langle x_i, x_j\rangle + \sum_i \alpha_i$$

$$\text{subject to } \sum_i \alpha_i y_i = 0 \text{ and } \alpha_i \in [0, C]$$

- Support vector expansion

$$f(x) = \sum_i \alpha_i y_i \langle x_i, x\rangle + b$$

slide by Alex Smola

The Kernel Trick for SVMs

- Linear soft margin problem

$$\underset{w,b}{\text{minimize}} \quad \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

$$\text{subject to } y_i\left[\langle w, \phi(x_i)\rangle + b\right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

- Dual problem

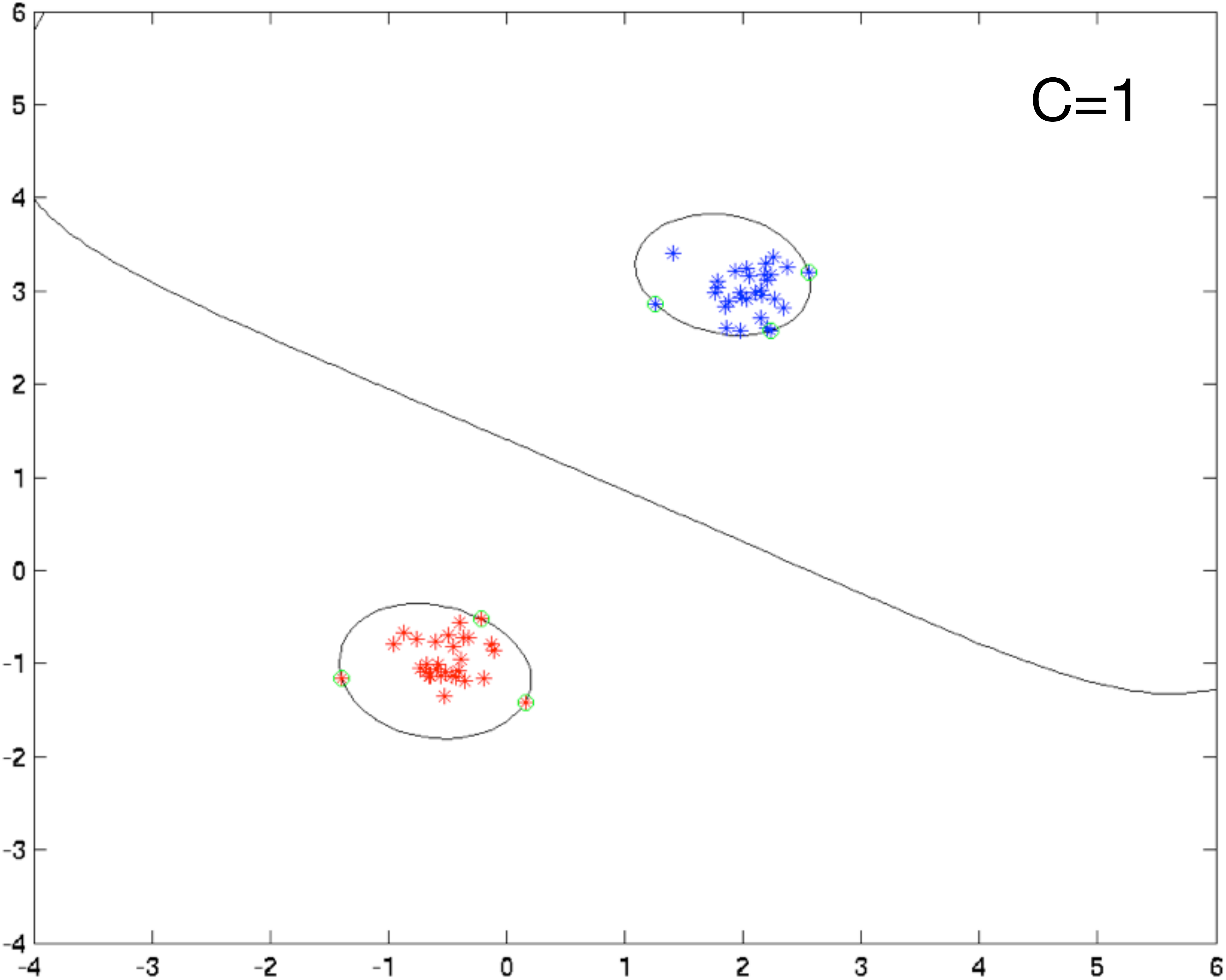
$$\underset{\alpha}{\text{maximize}} - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j k(x_i, x_j) + \sum_i \alpha_i$$

$$\text{subject to } \sum_i \alpha_i y_i = 0 \text{ and } \alpha_i \in [0, C]$$

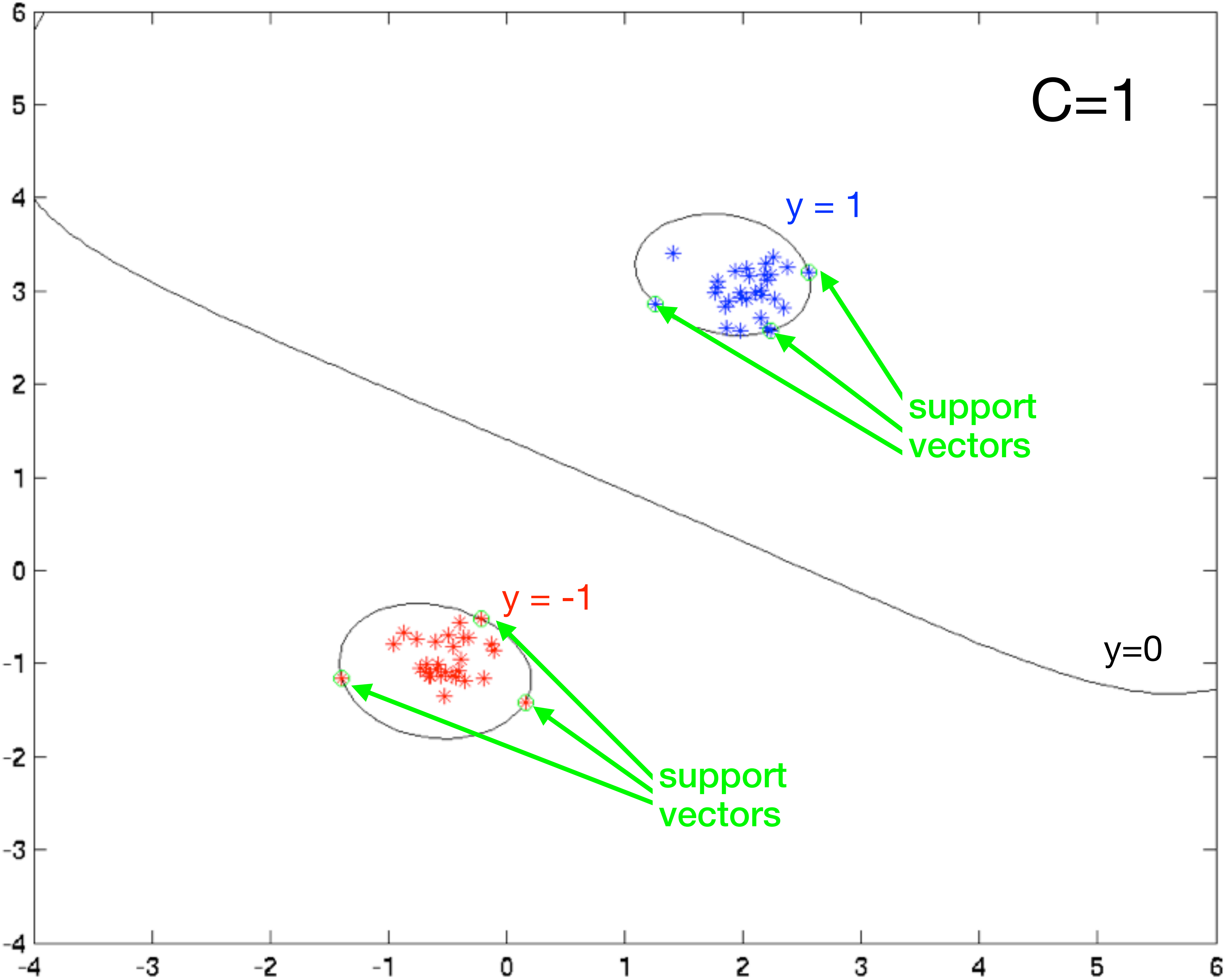
- Support vector expansion

$$f(x) = \sum_i \alpha_i y_i k(x_i, x) + b$$

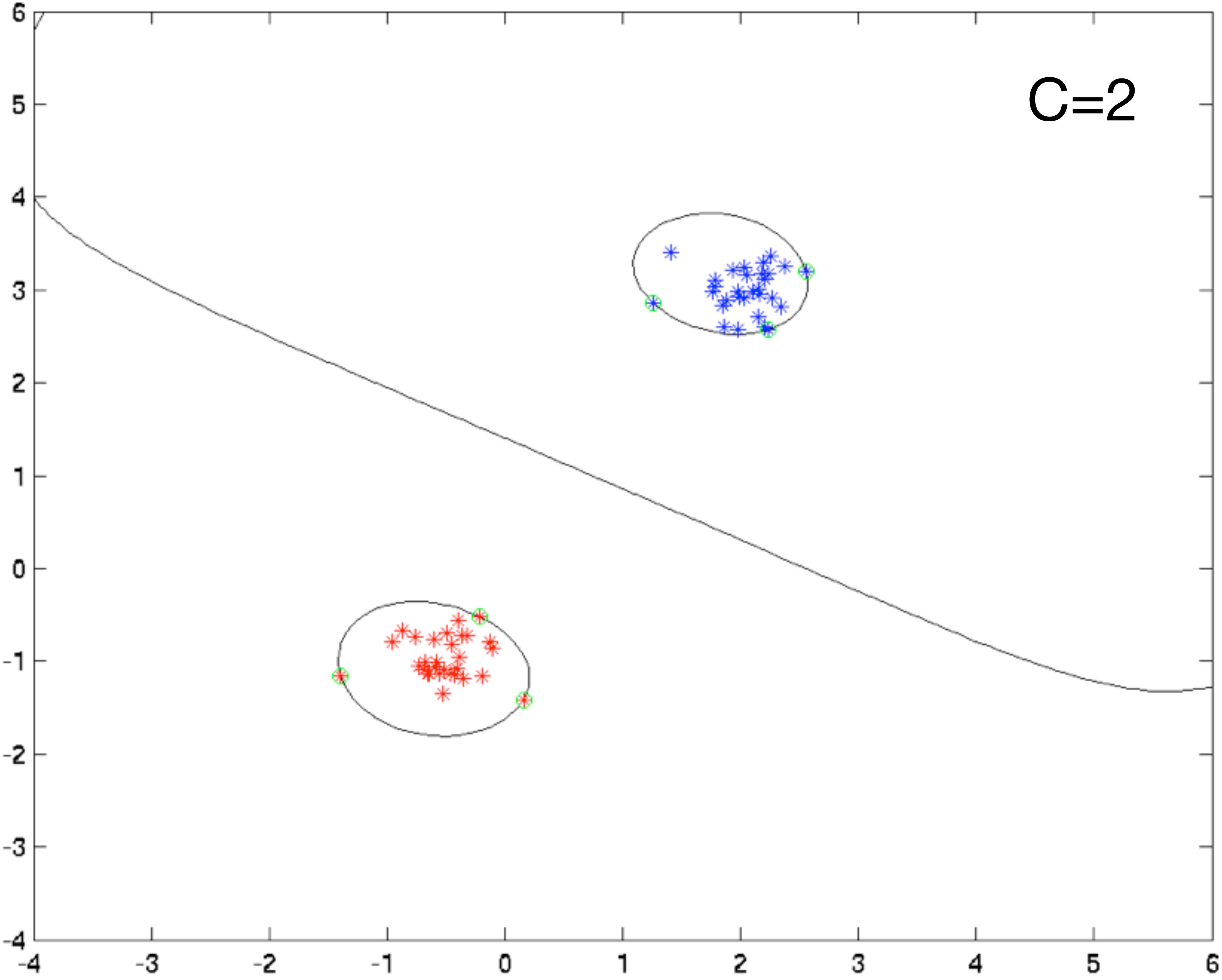
slide by Alex Smola



slide by Alex Smola

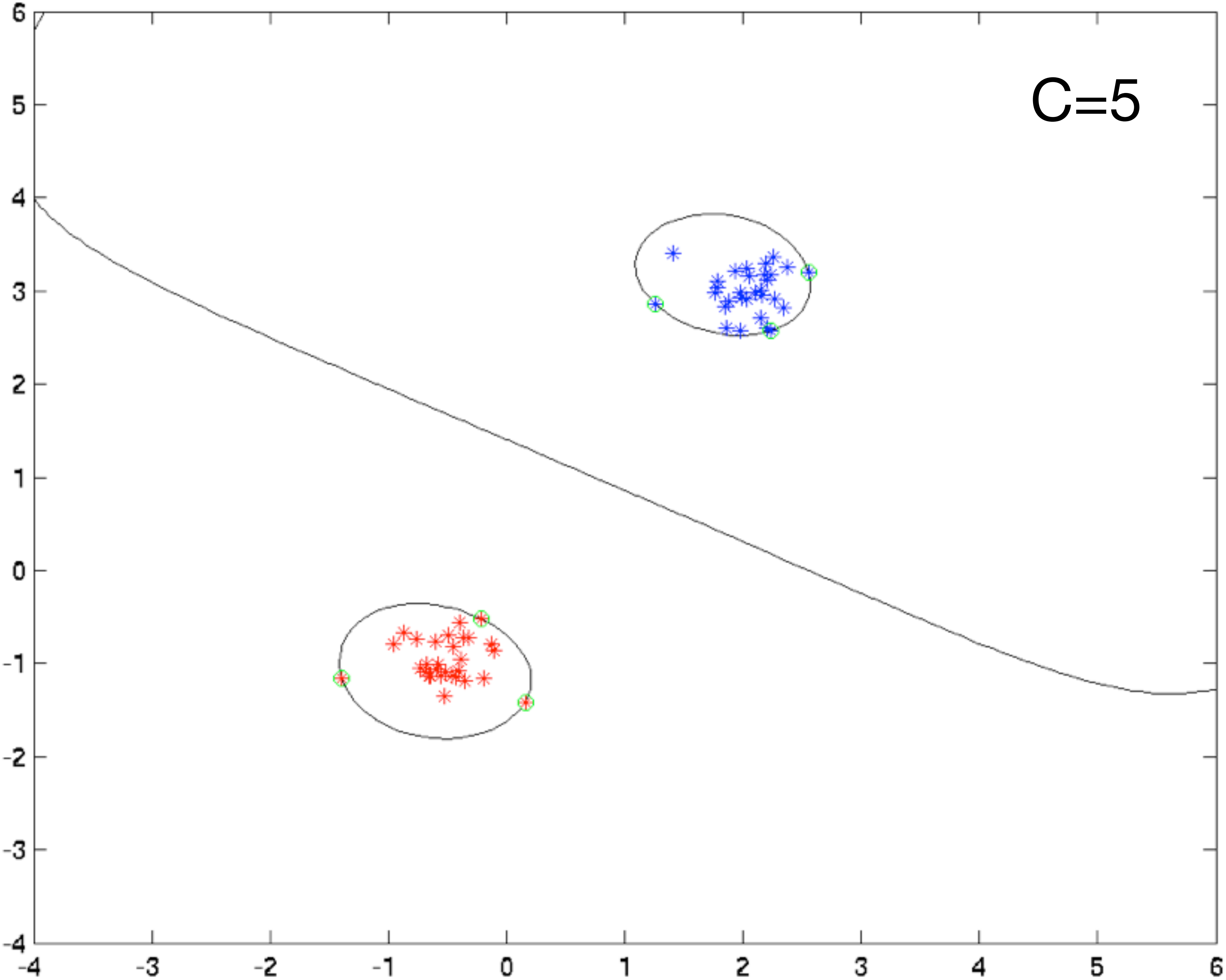


slide by Alex Smola

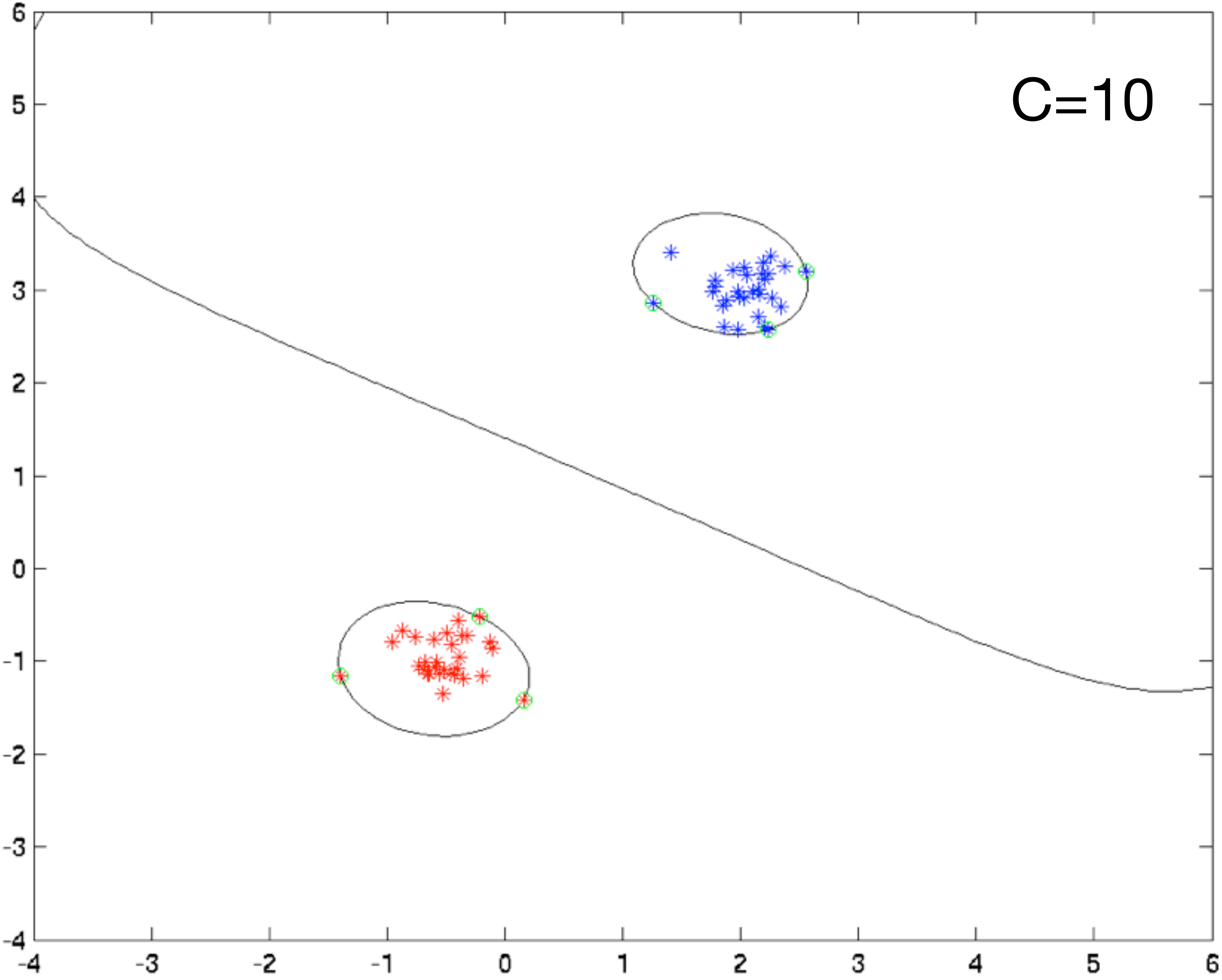


C=2

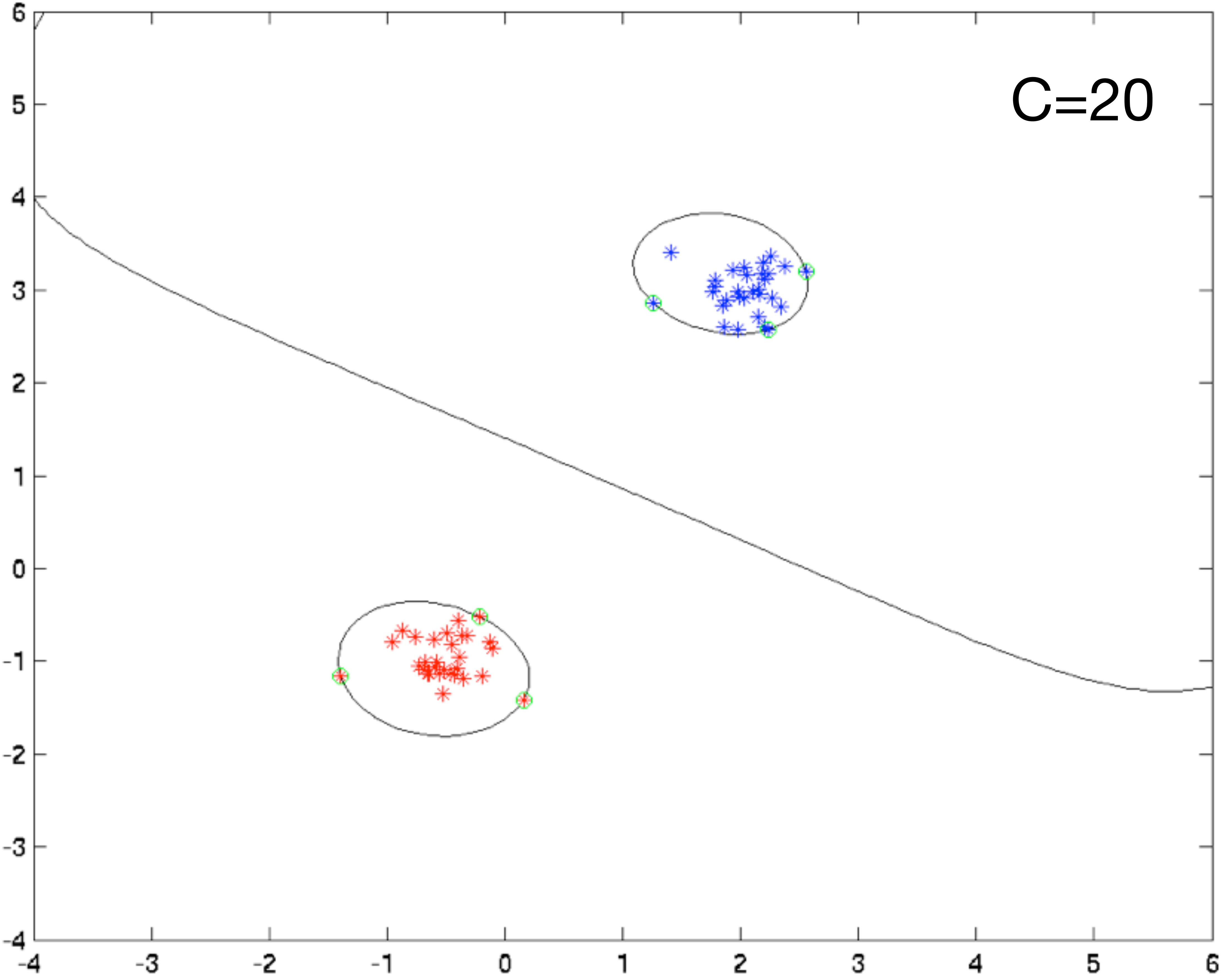
slide by Alex Smola



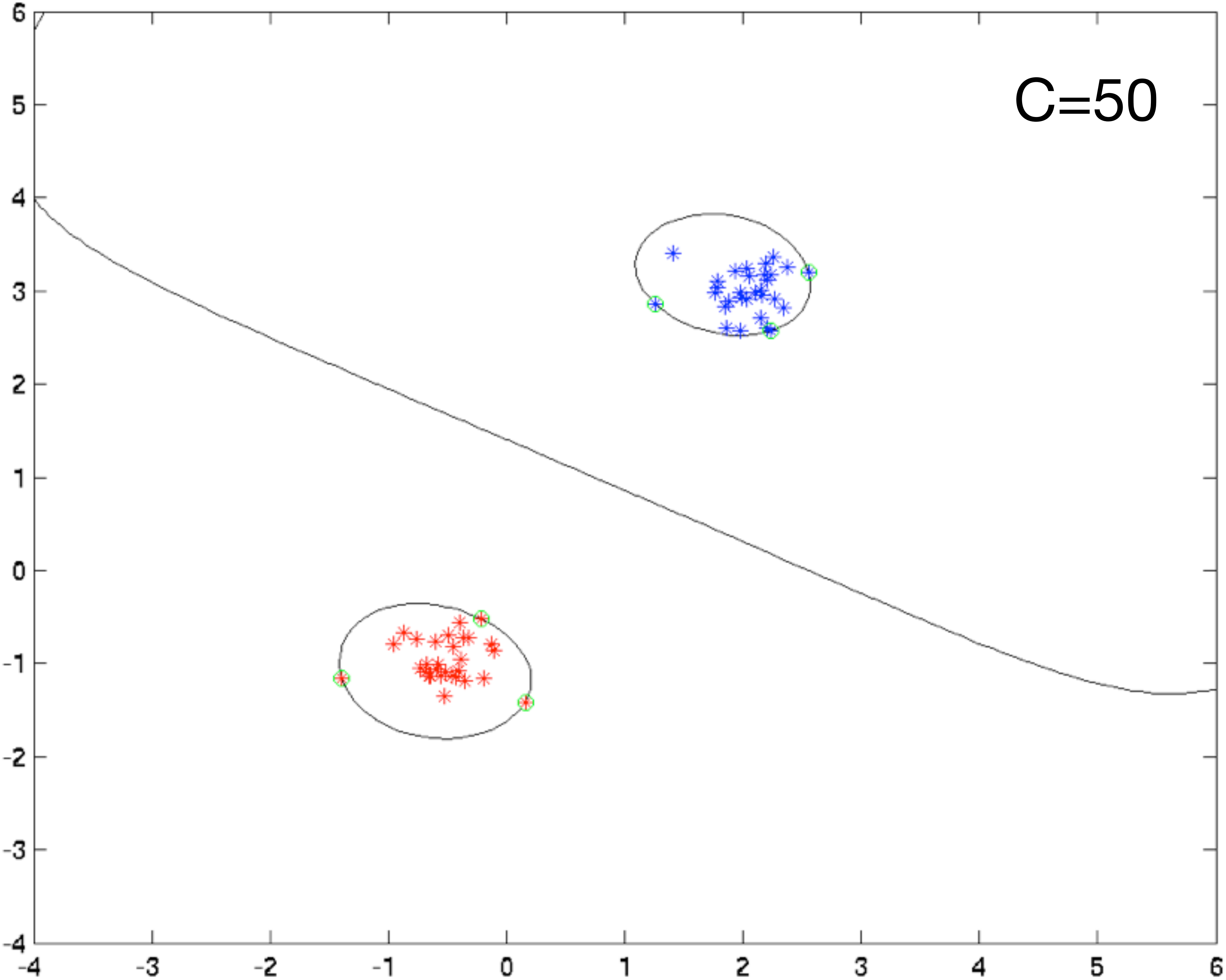
slide by Alex Smola



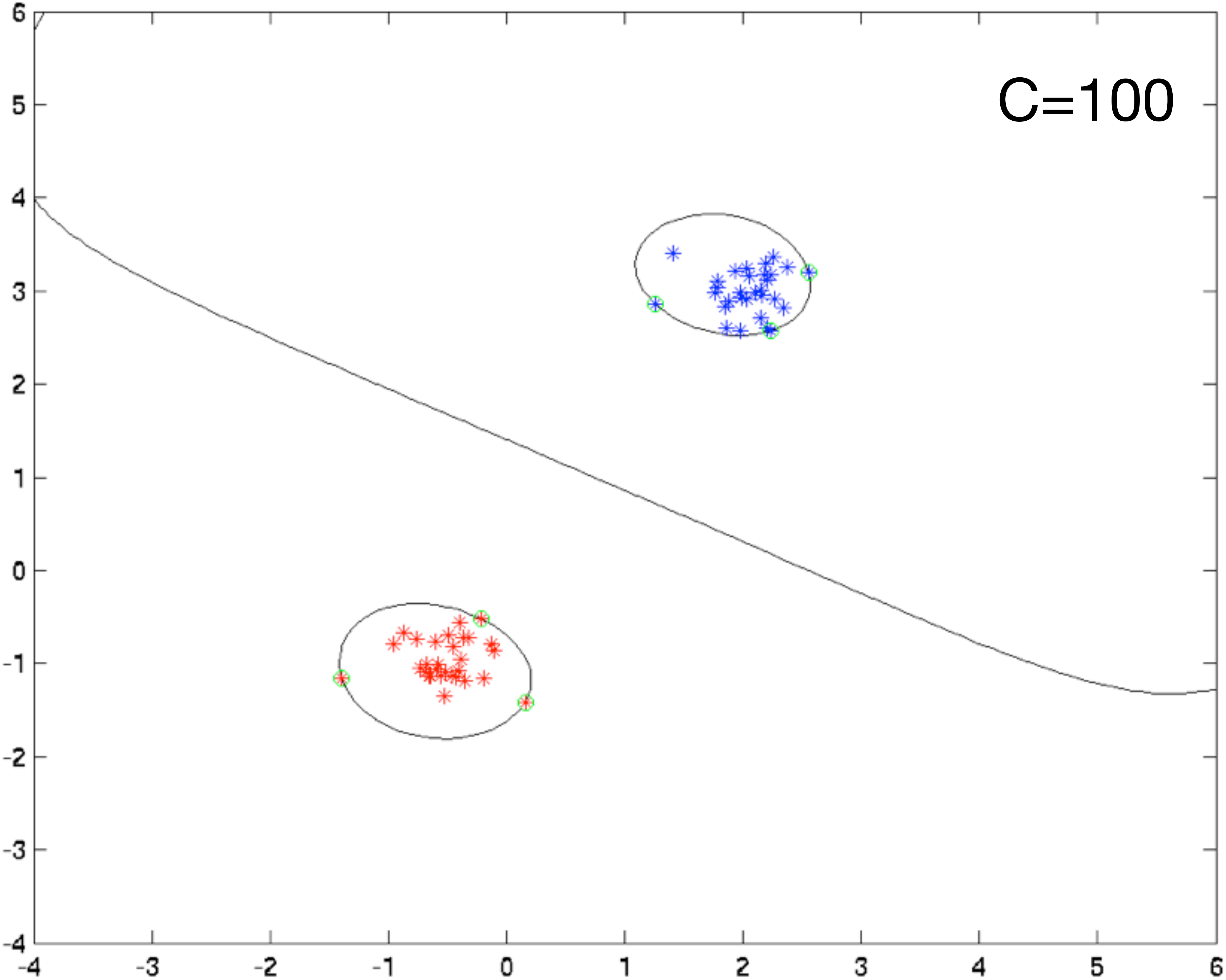
slide by Alex Smola



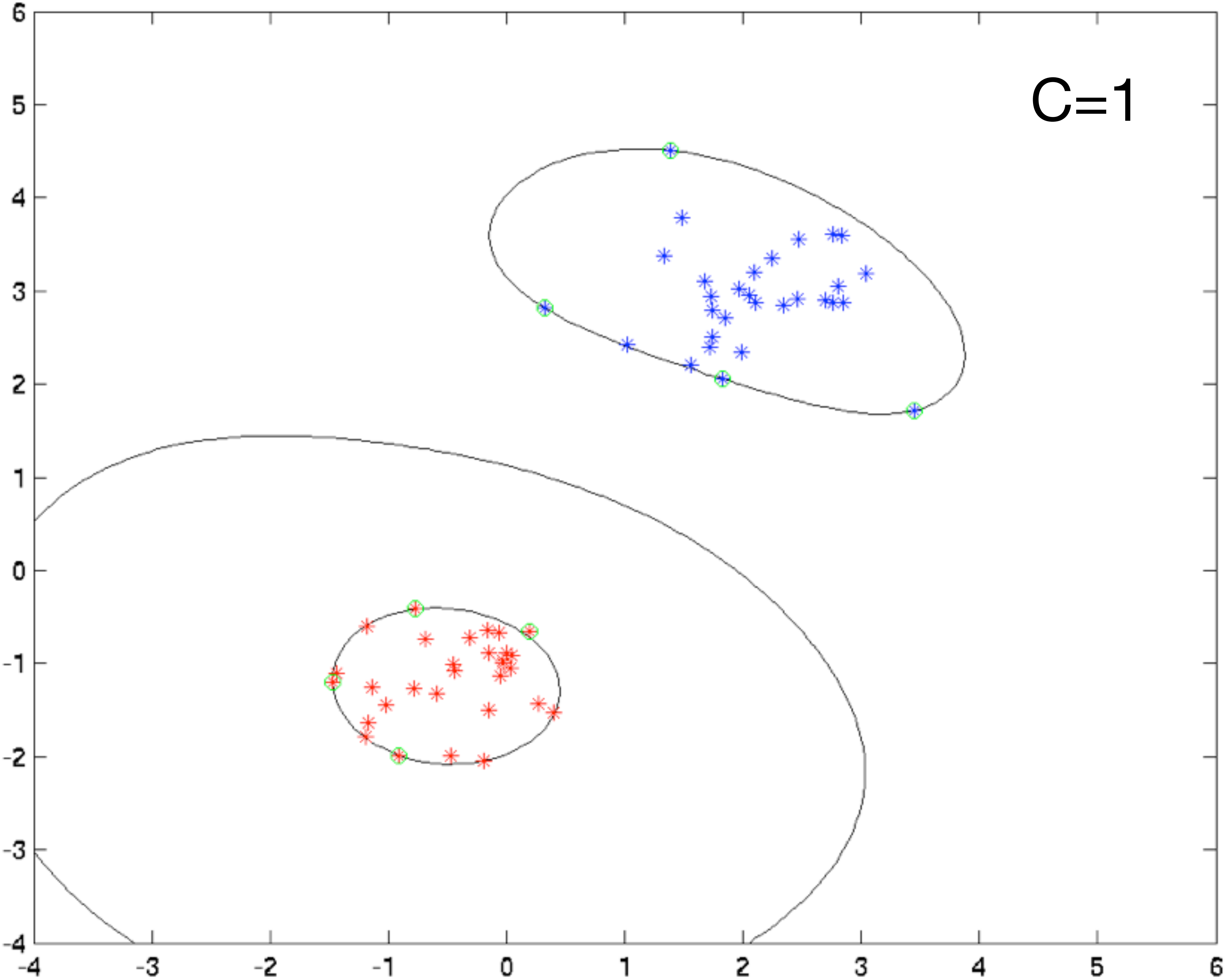
slide by Alex Smola



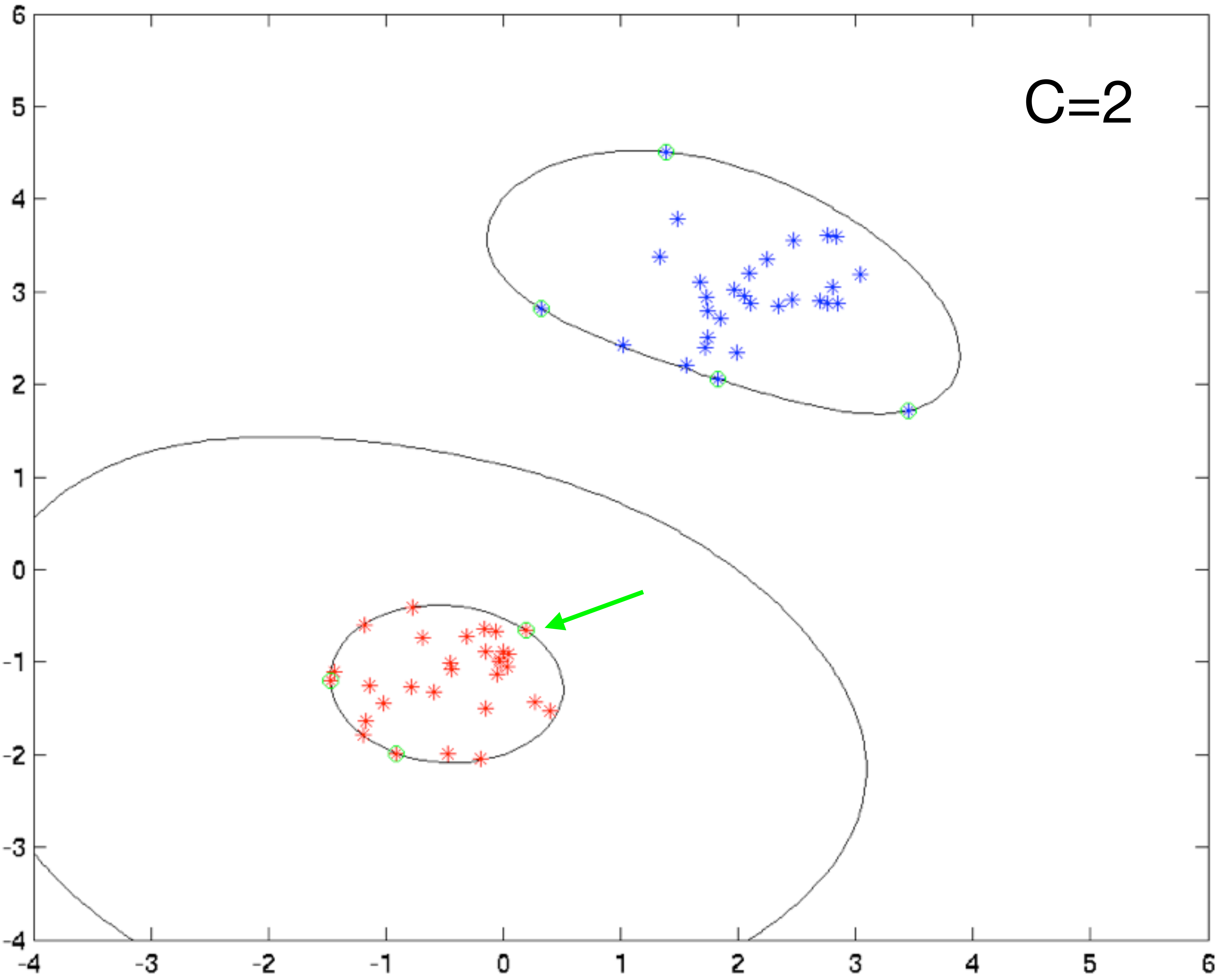
slide by Alex Smola



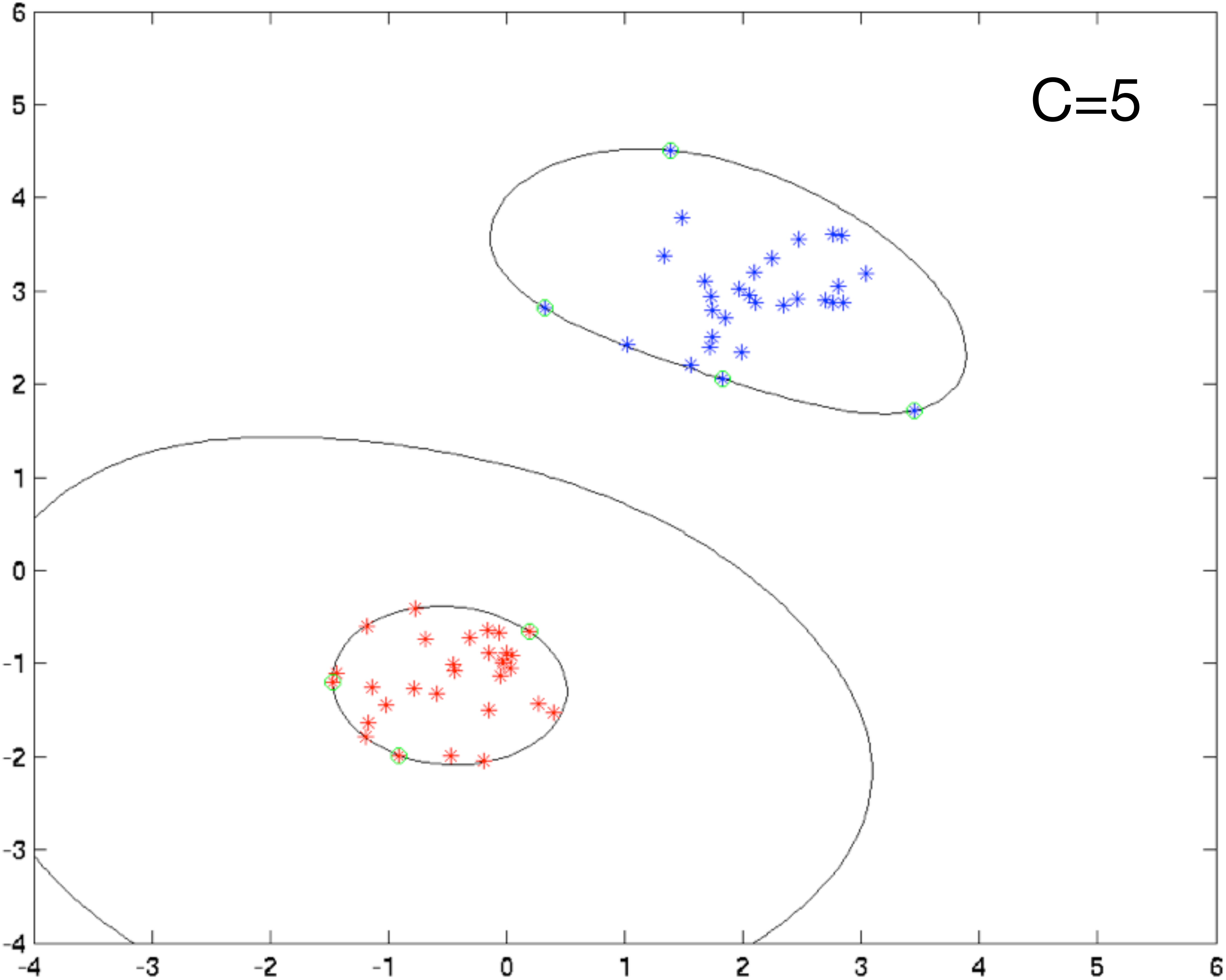
slide by Alex Smola



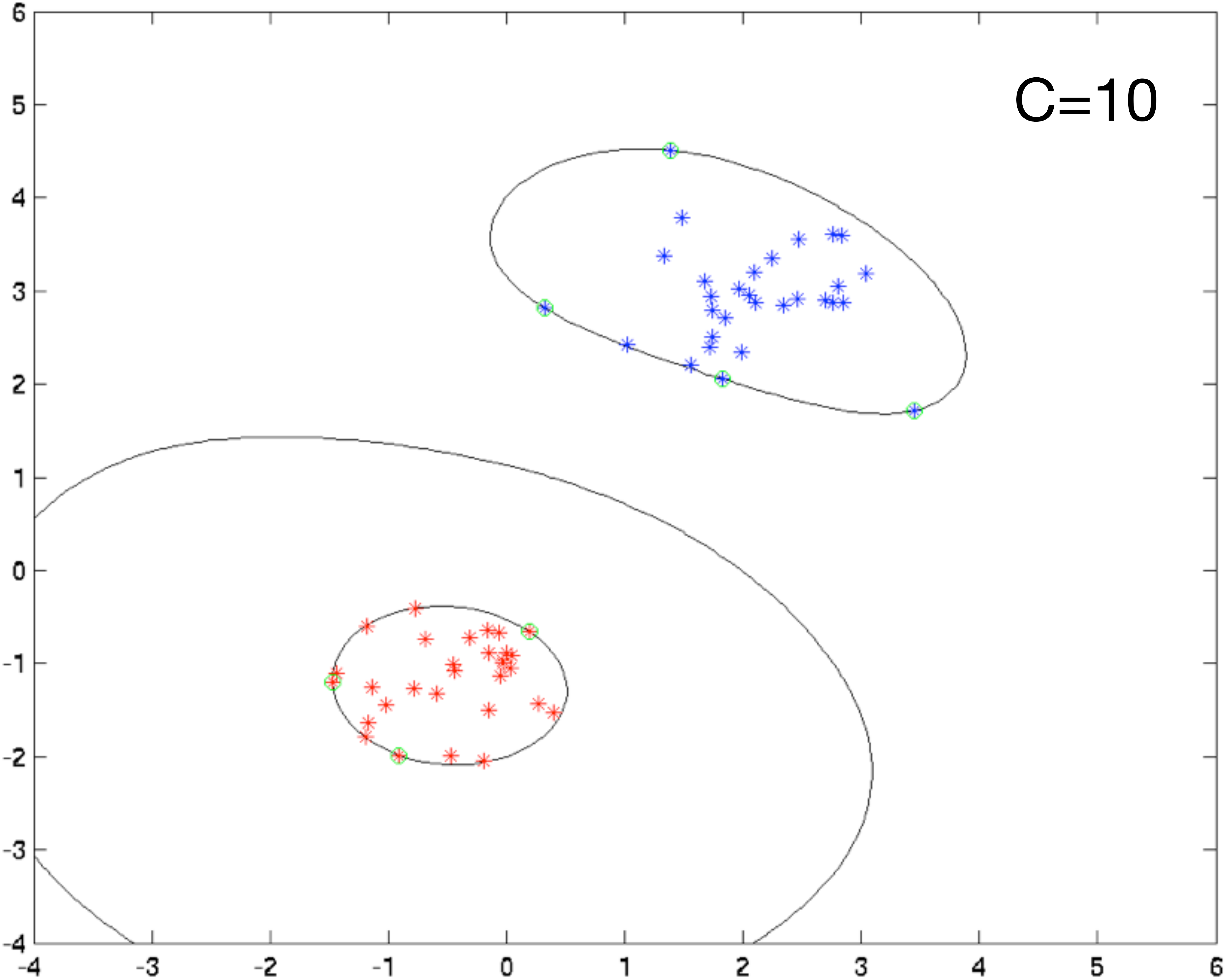
slide by Alex Smola



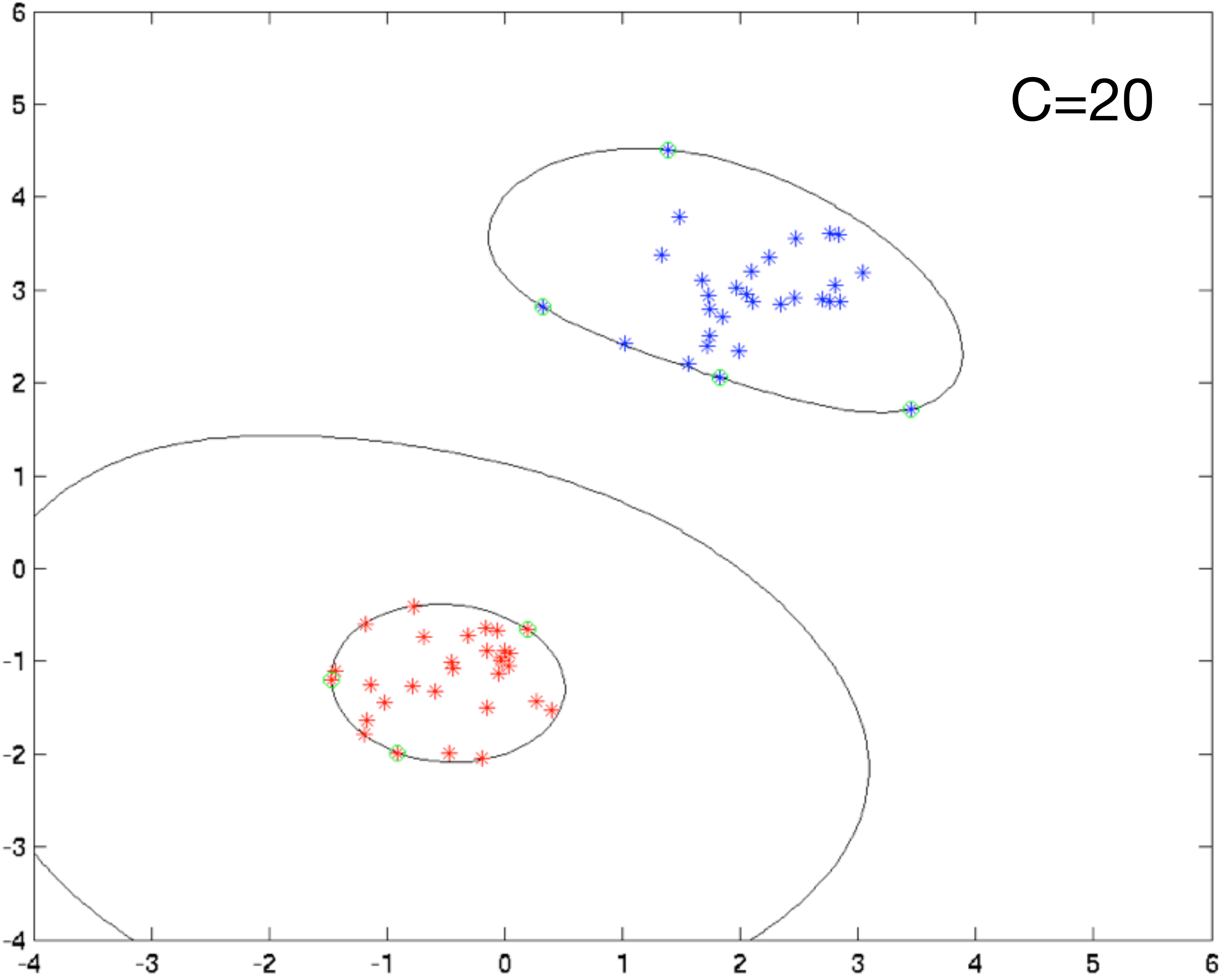
slide by Alex Smola



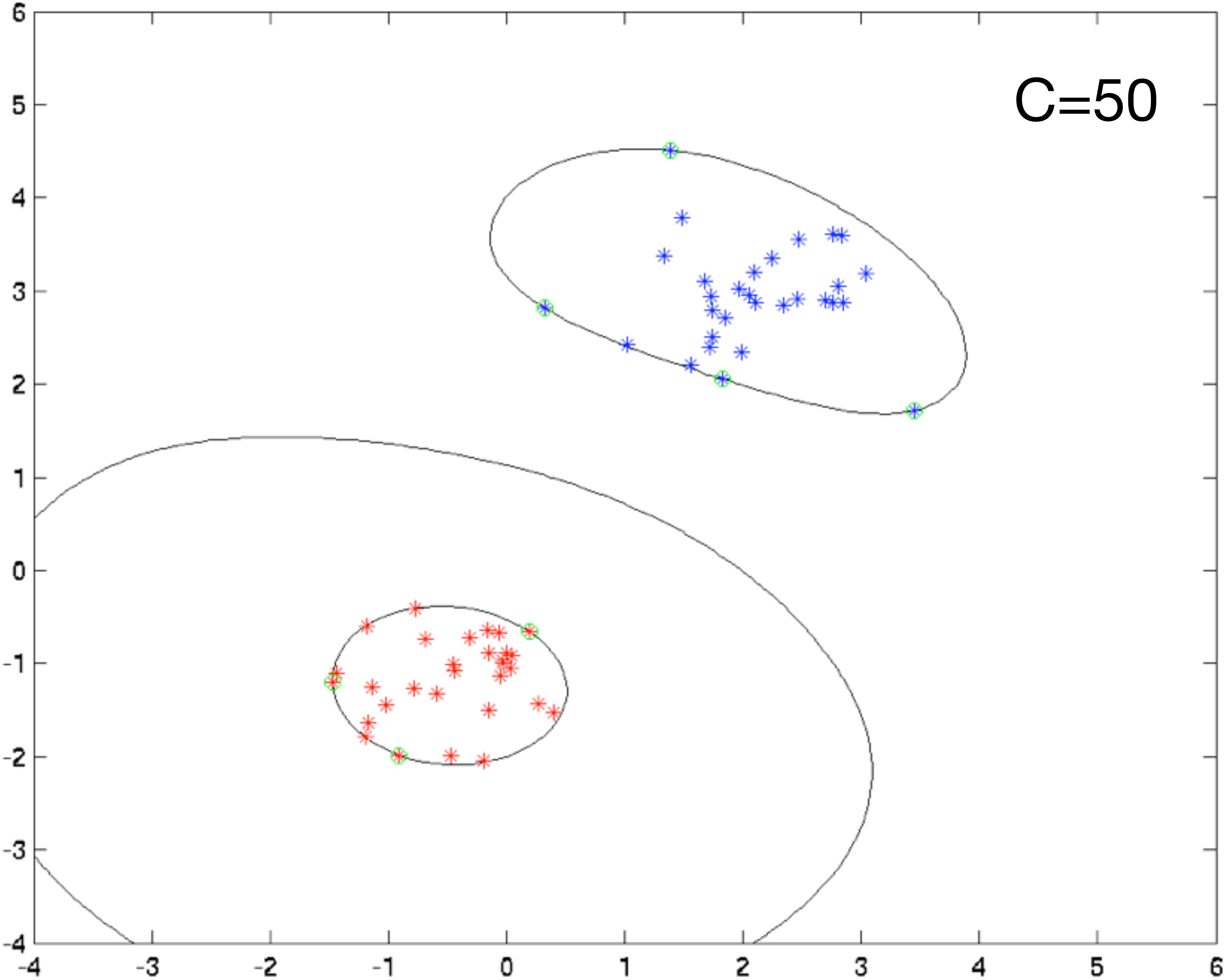
slide by Alex Smola



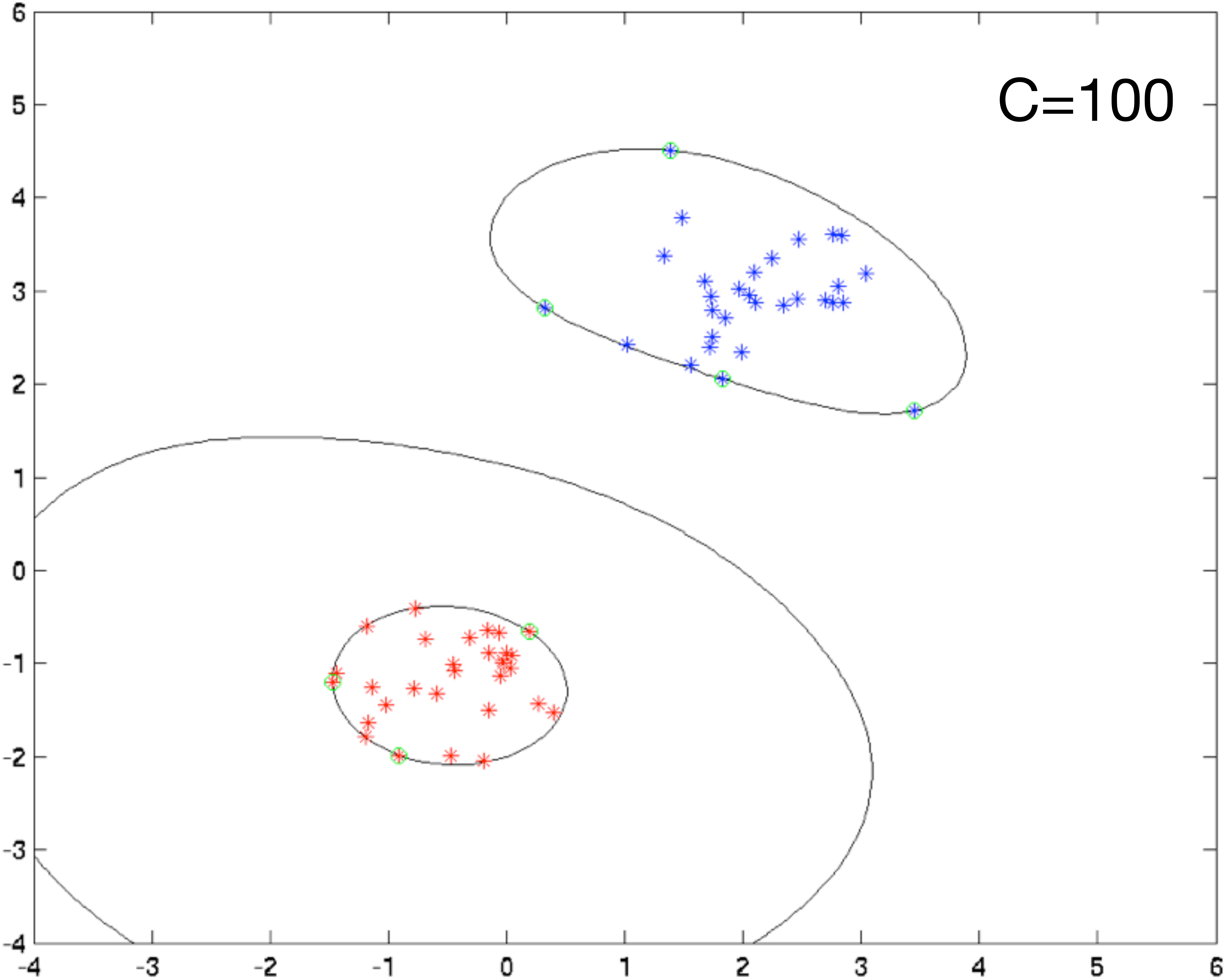
slide by Alex Smola



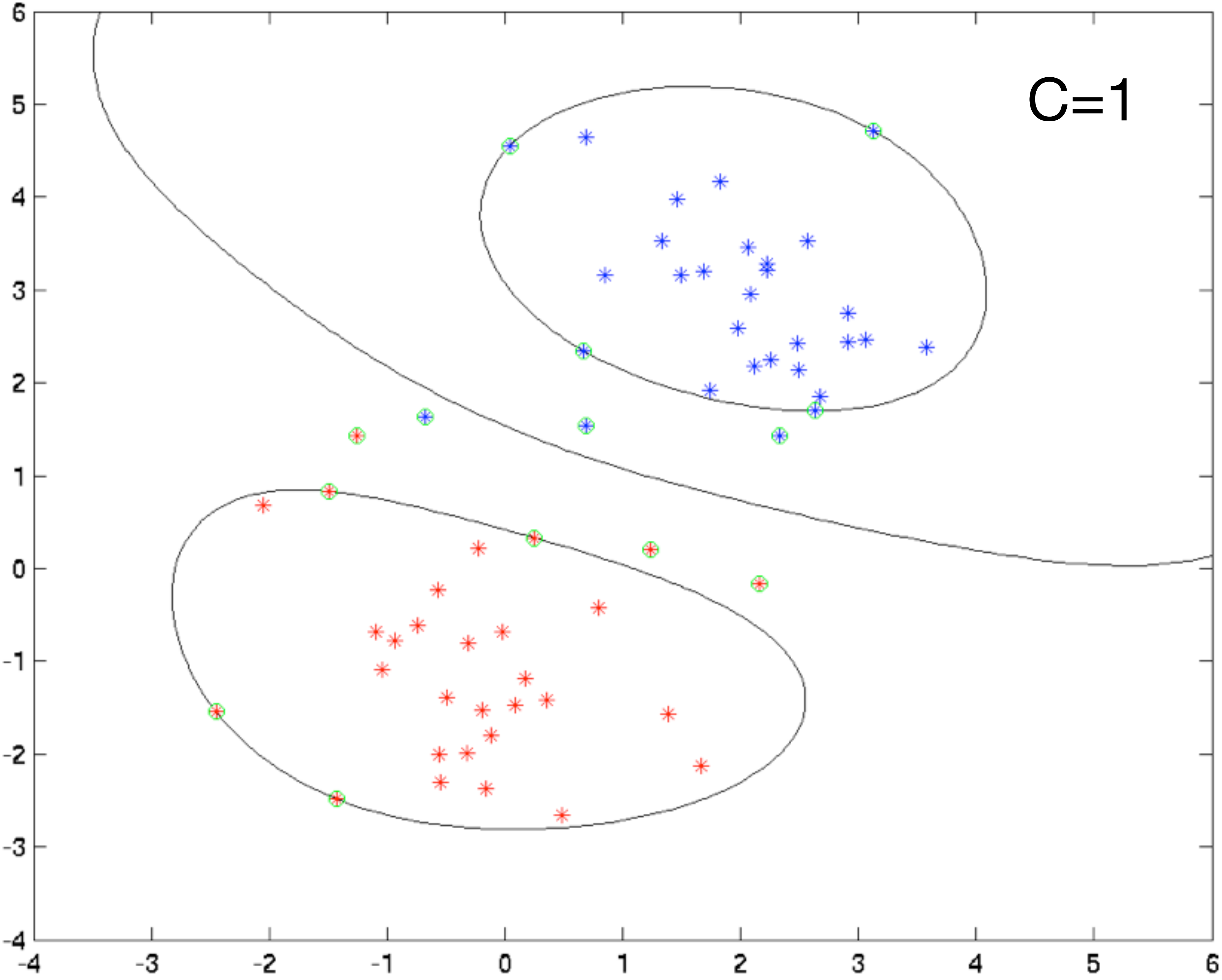
slide by Alex Smola



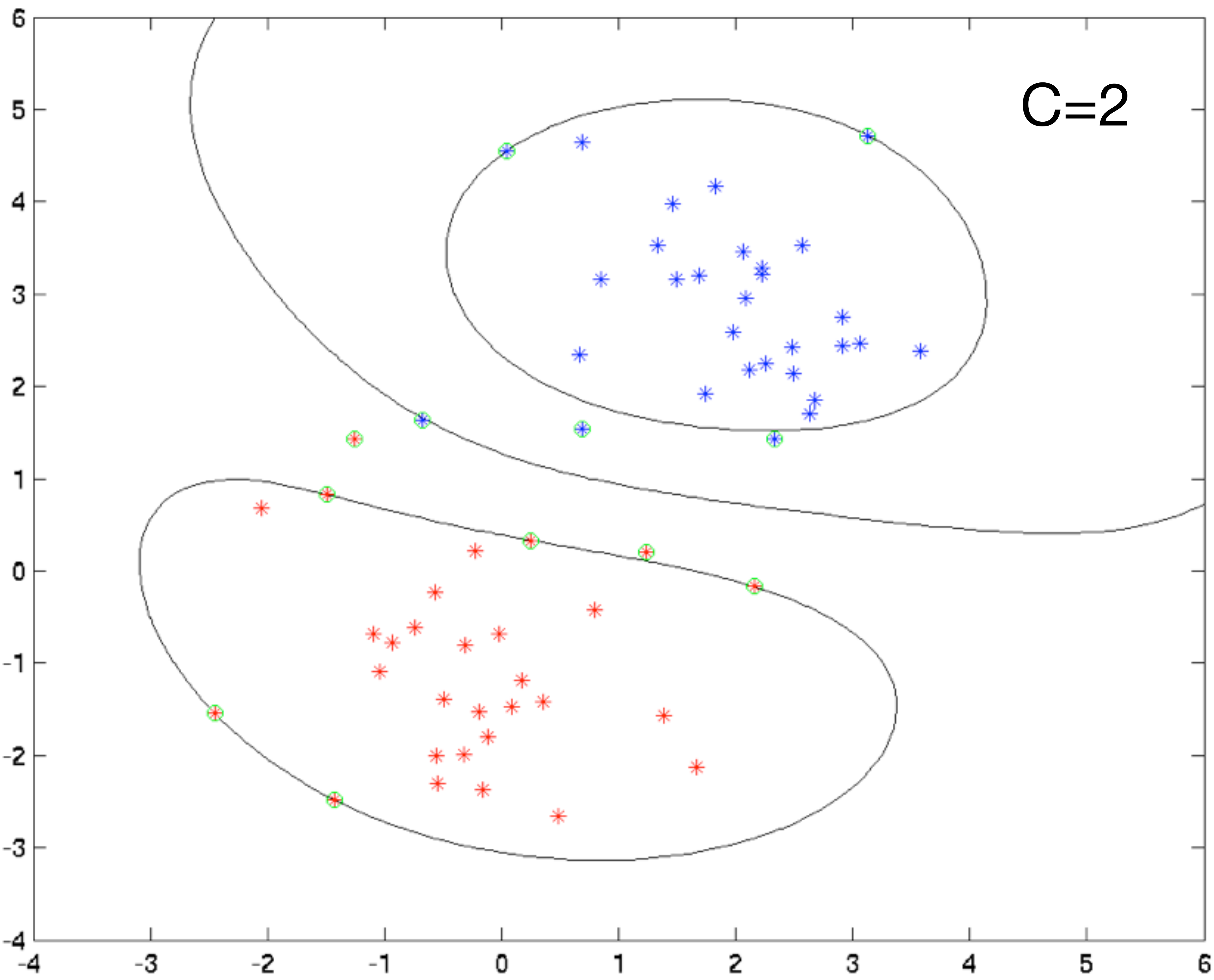
slide by Alex Smola



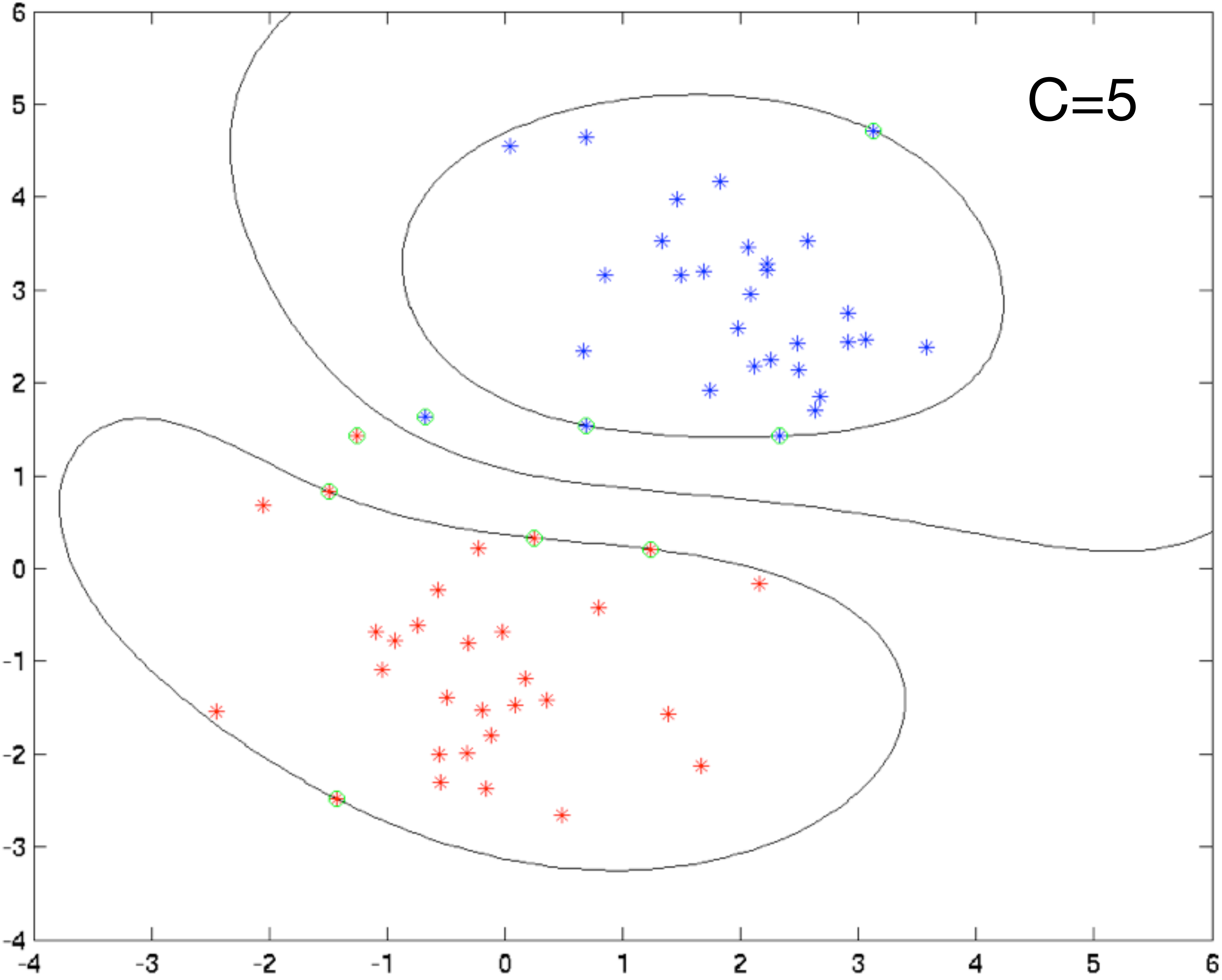
slide by Alex Smola



slide by Alex Smola

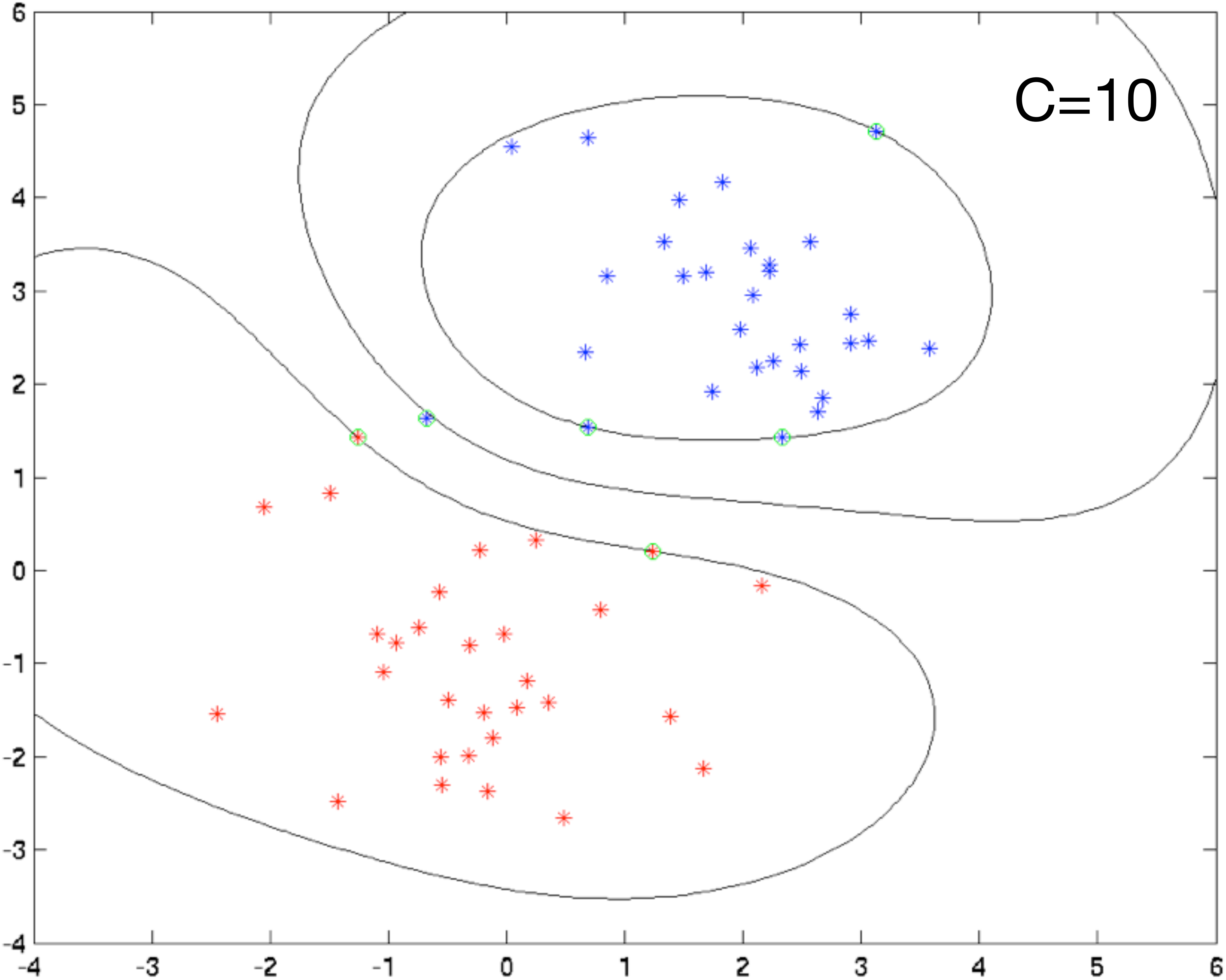


slide by Alex Smola

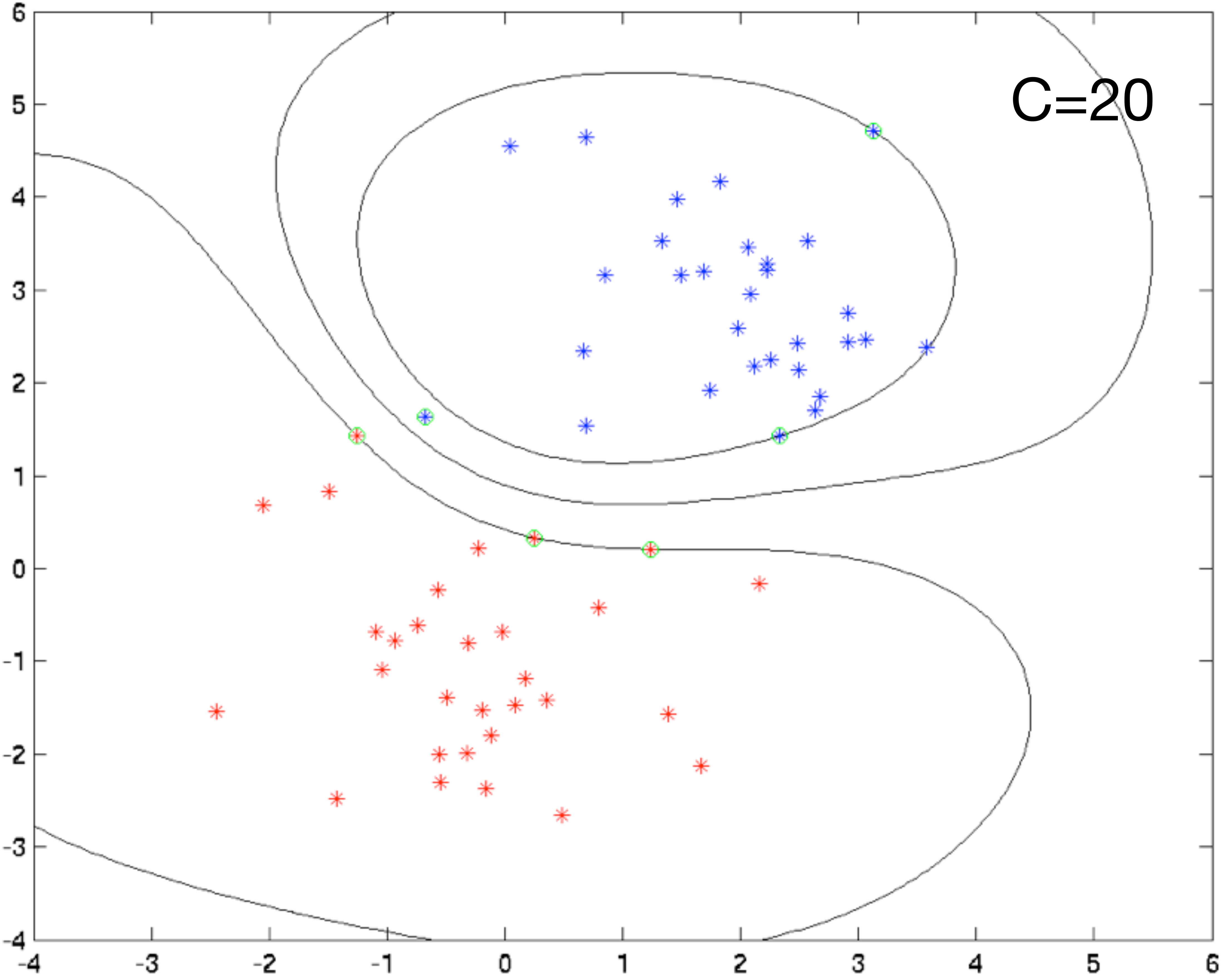


C=5

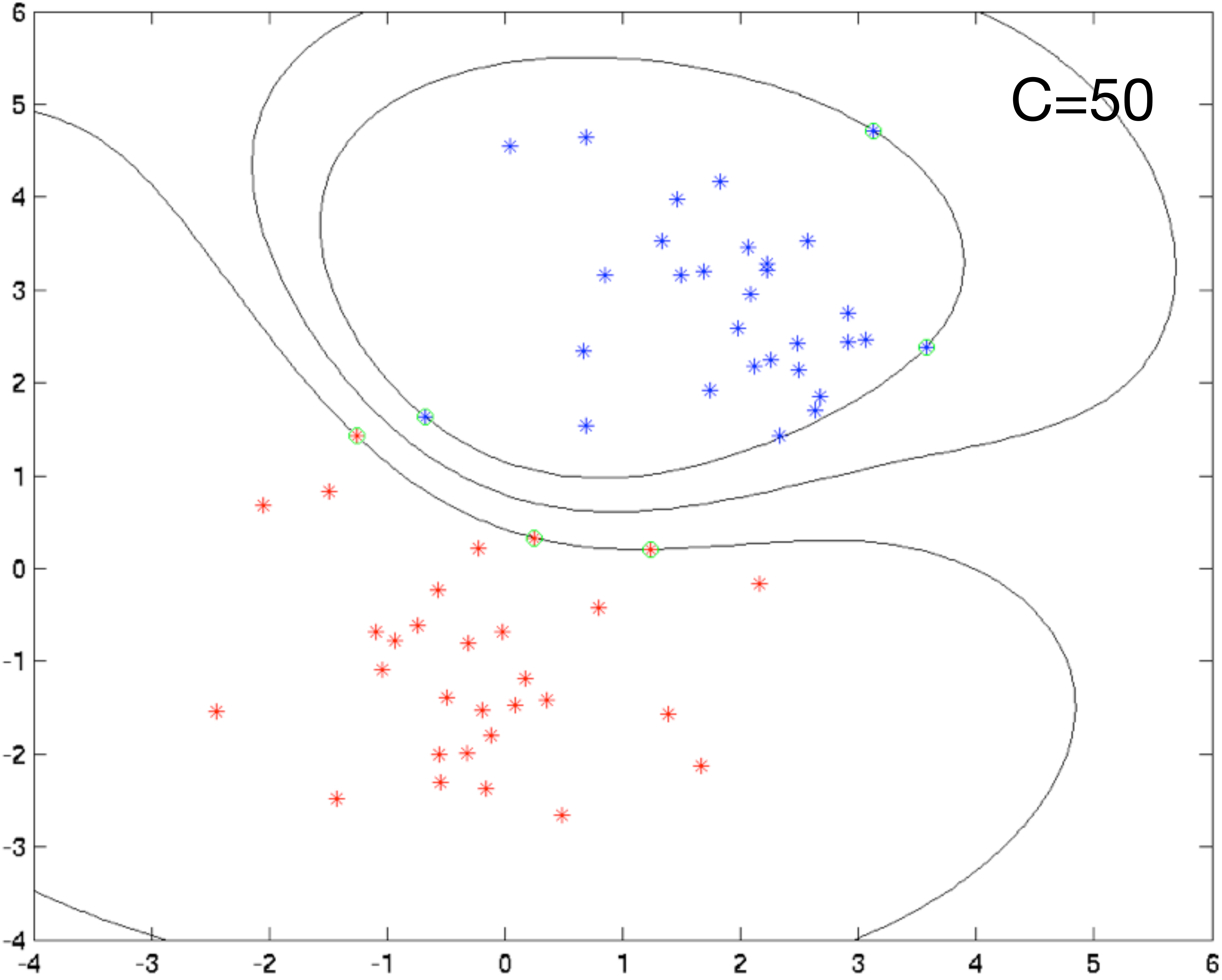
slide by Alex Smola



slide by Alex Smola

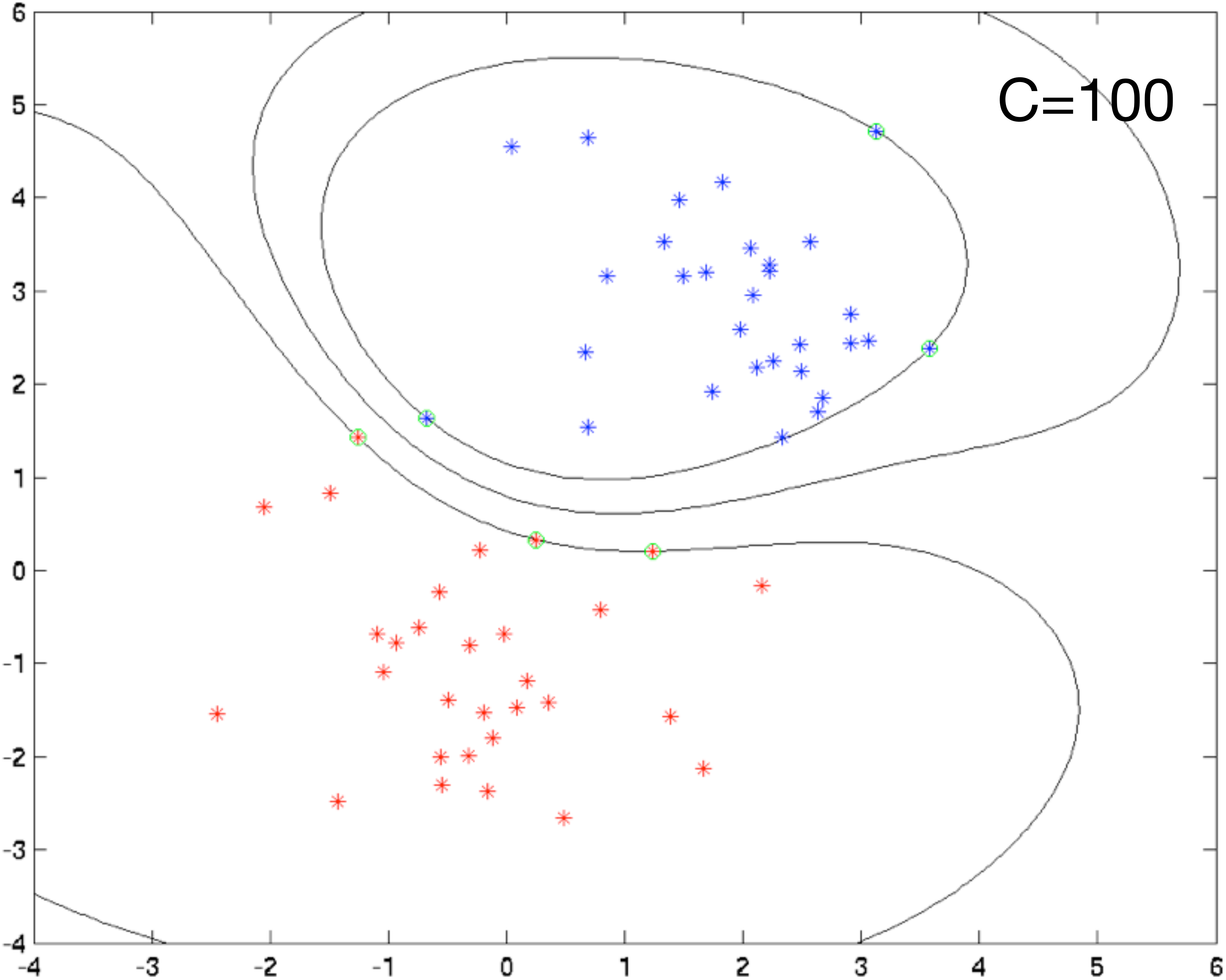


slide by Alex Smola

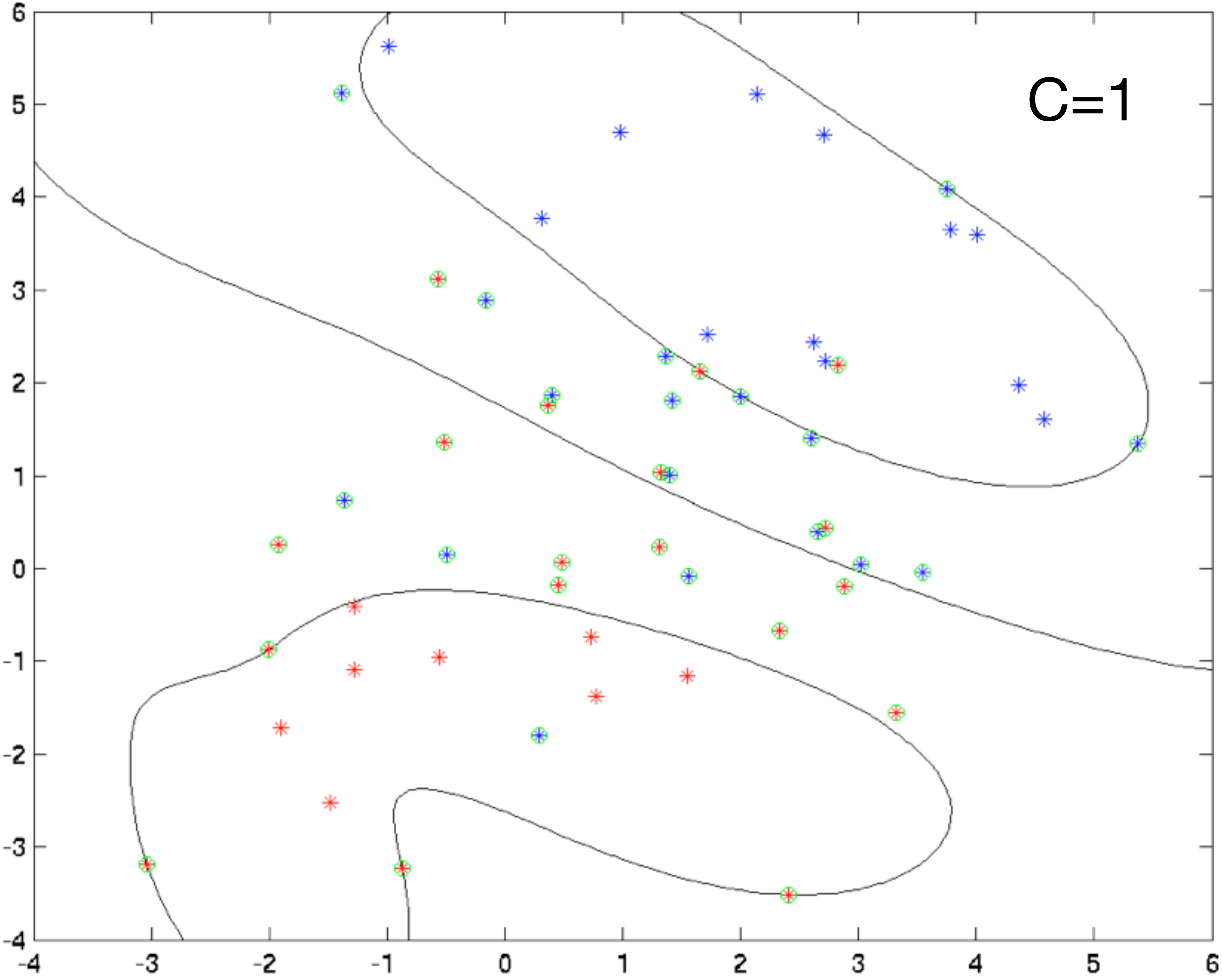


C=50

slide by Alex Smola

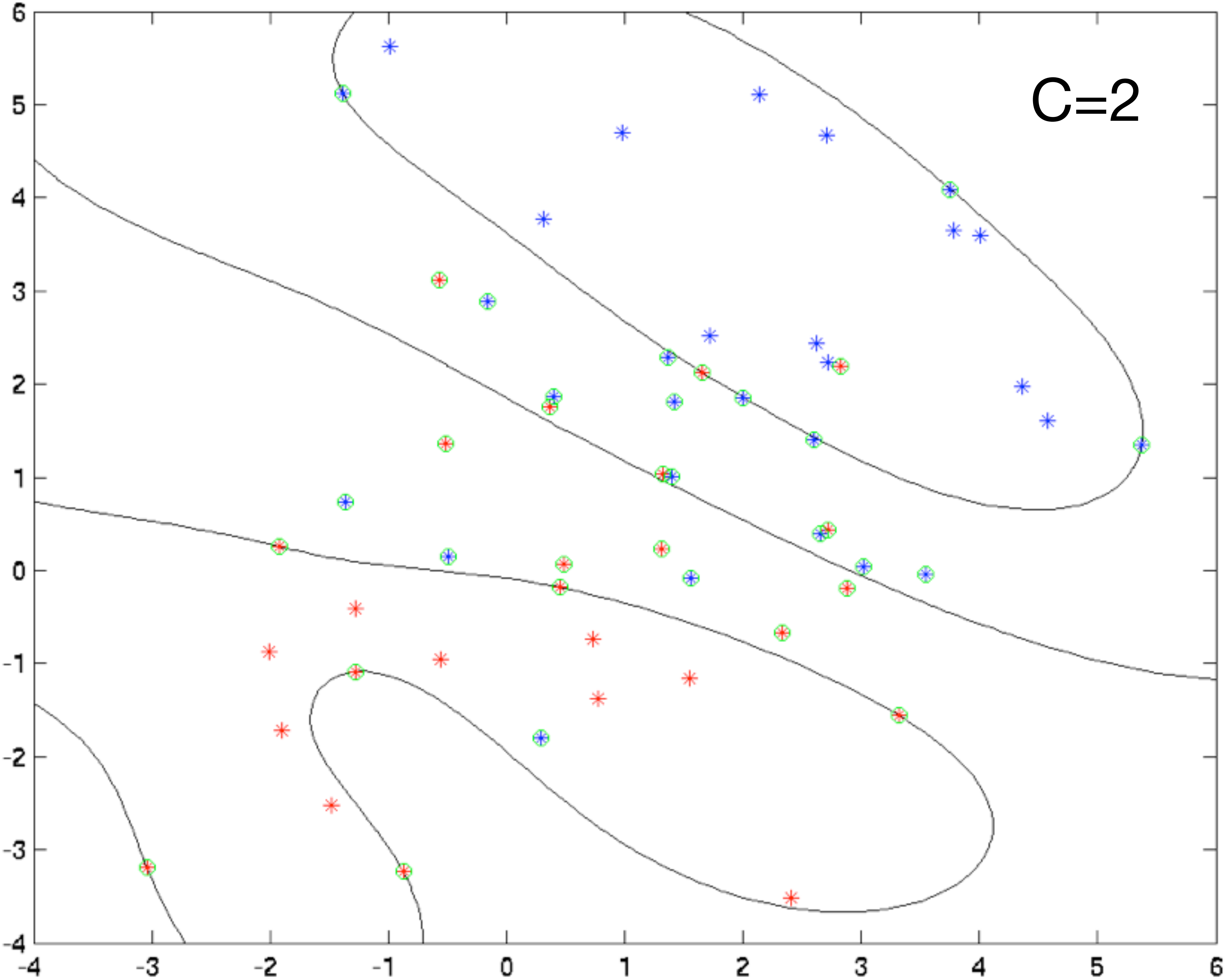


slide by Alex Smola

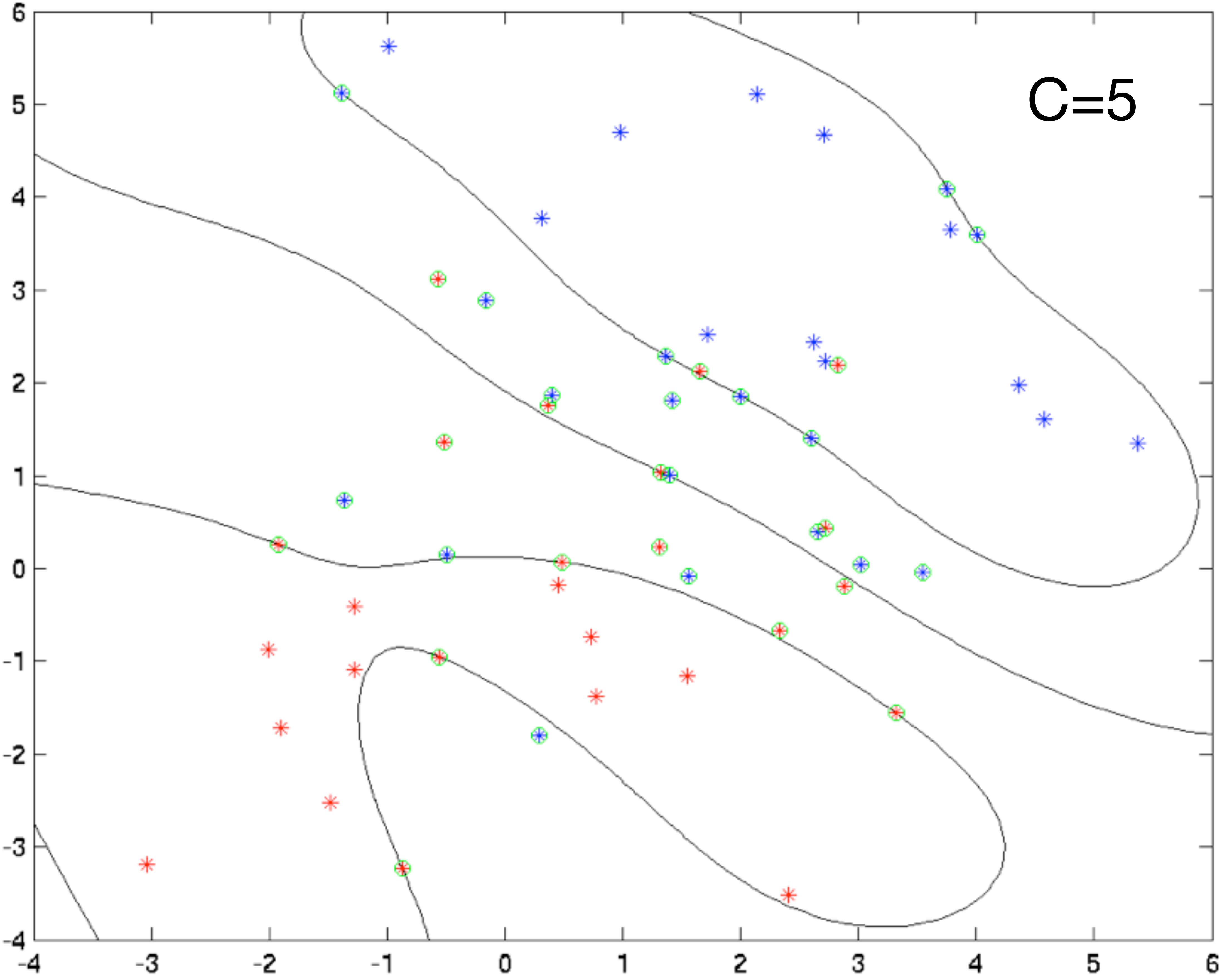


C=1

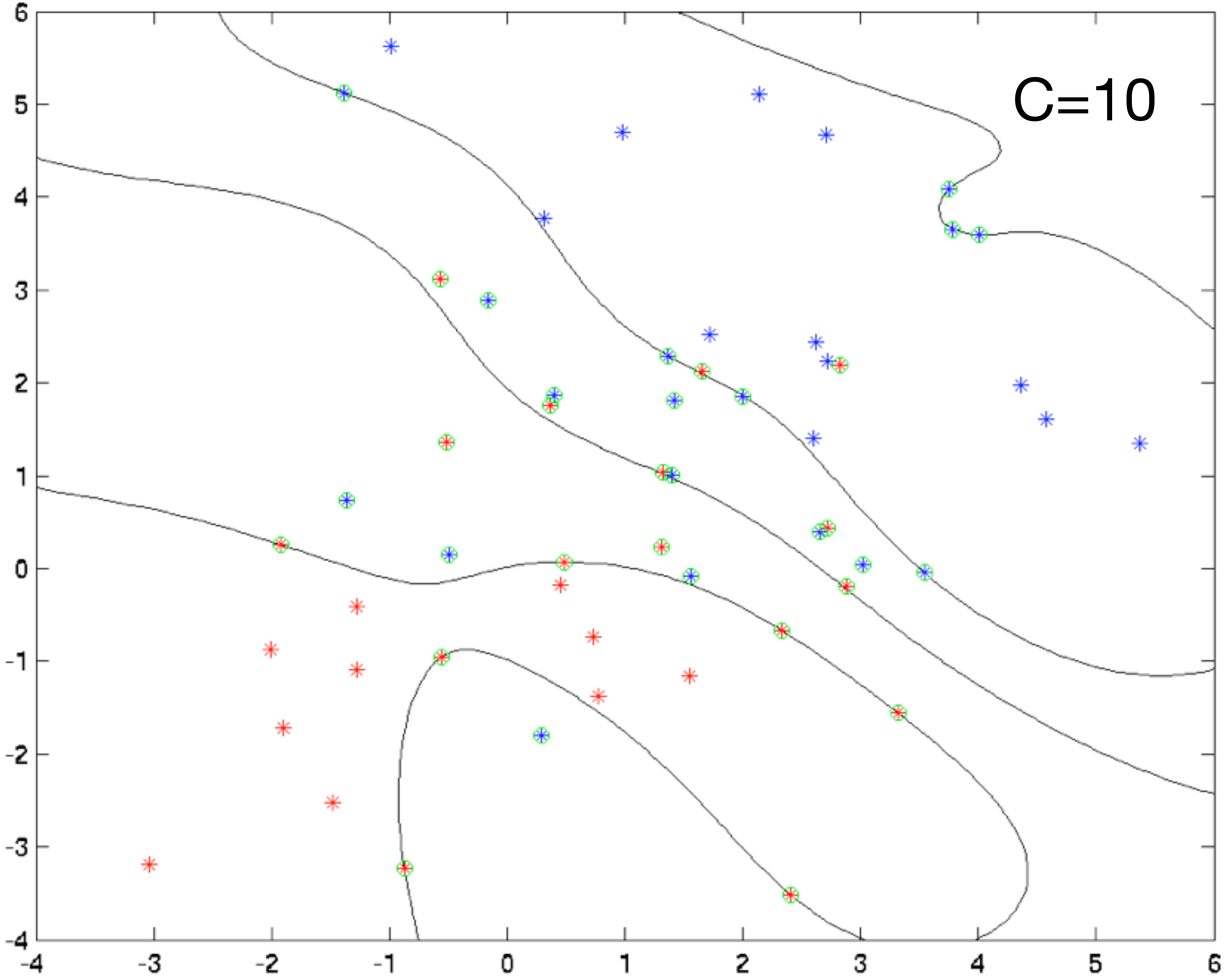
slide by Alex Smola



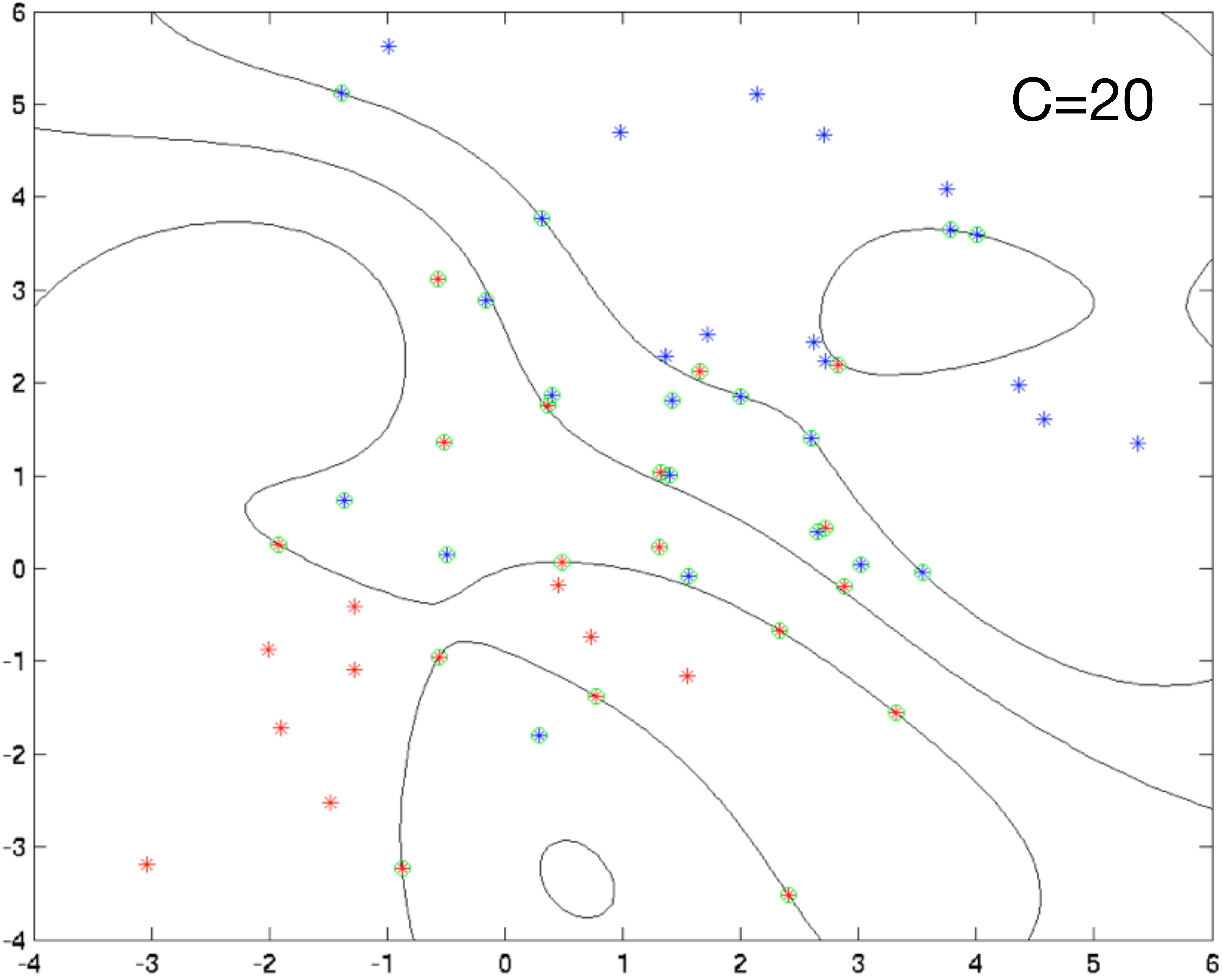
slide by Alex Smola



slide by Alex Smola

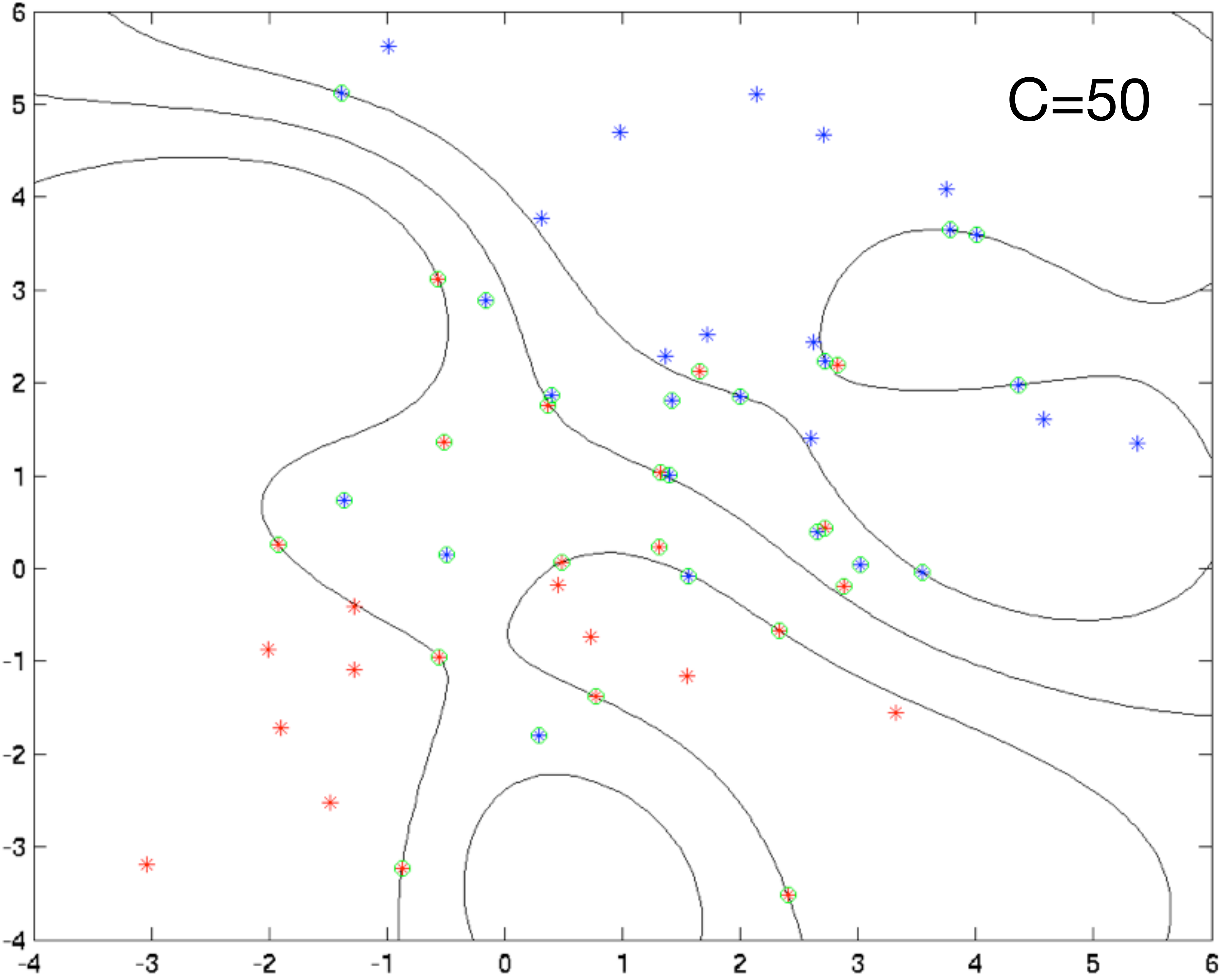


slide by Alex Smola



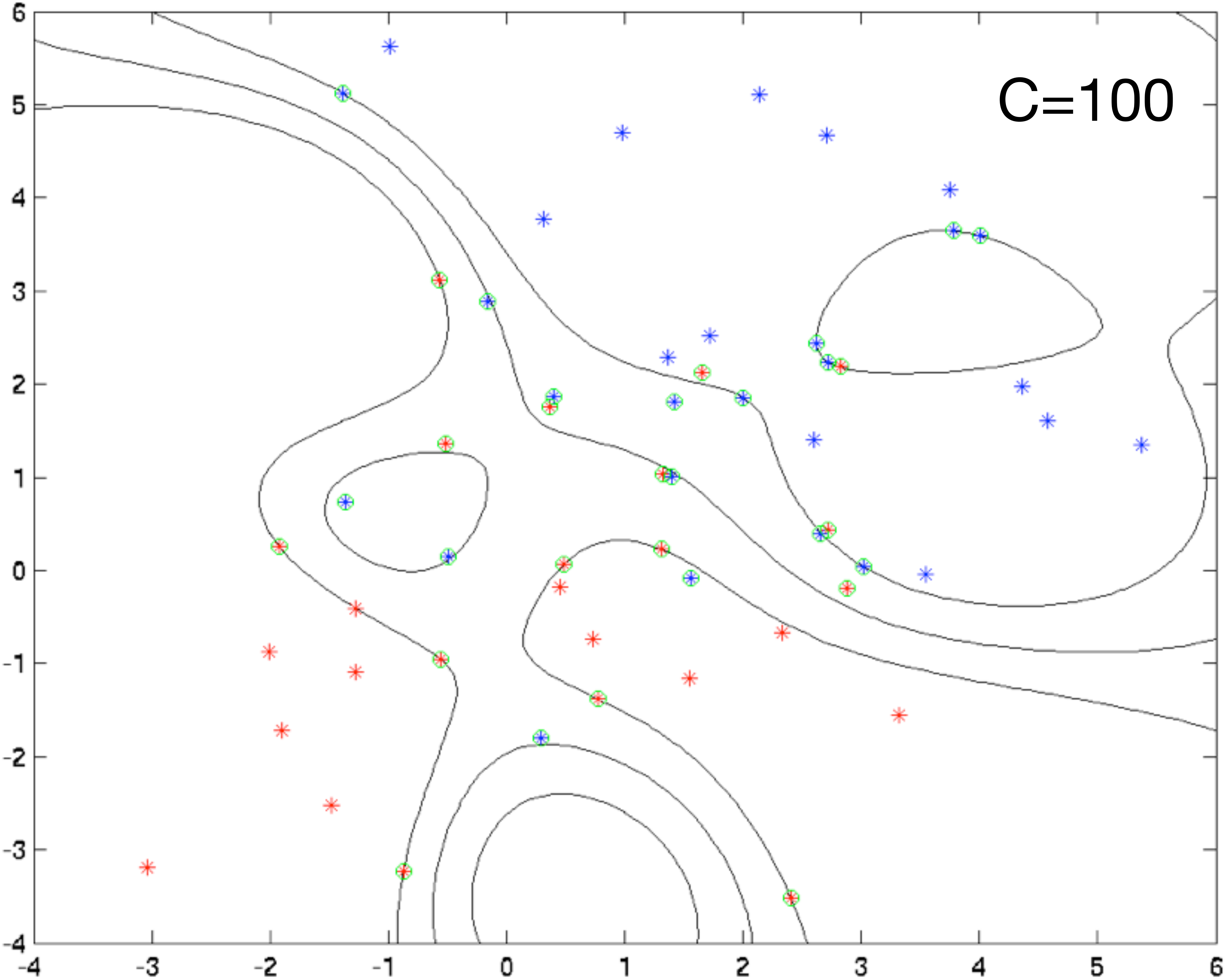
C=20

slide by Alex Smola



C=50

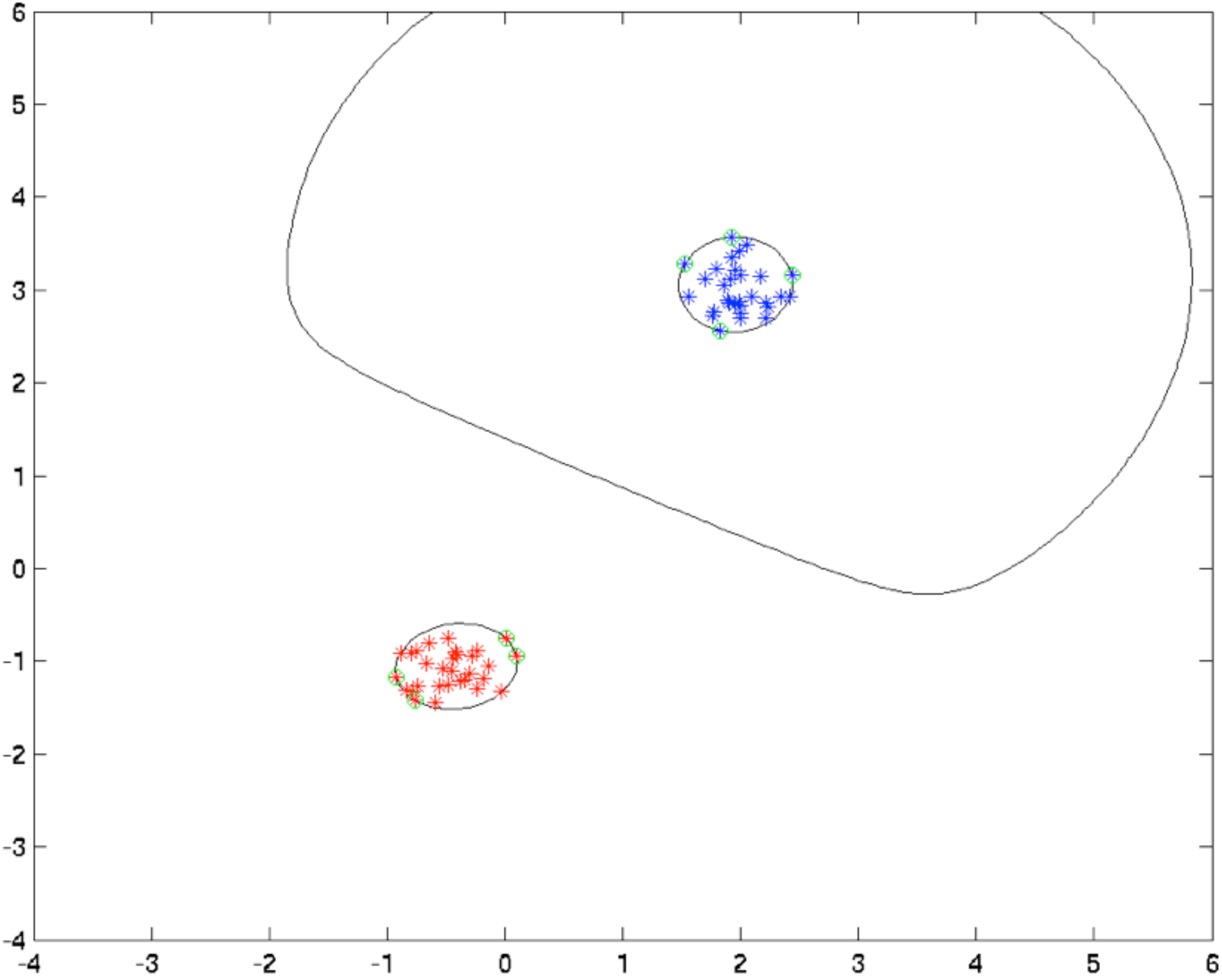
slide by Alex Smola



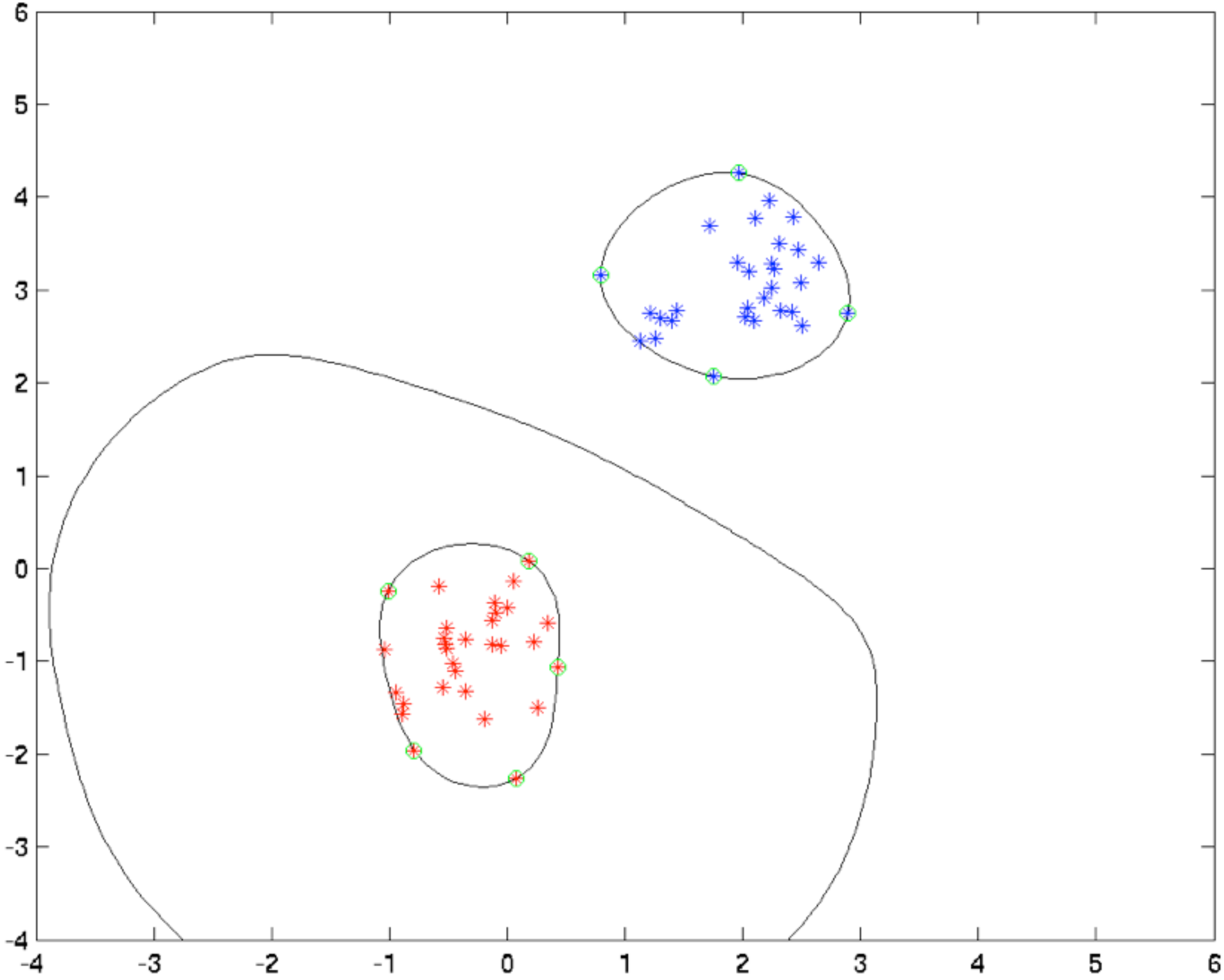
slide by Alex Smola

And now with a narrower kernel

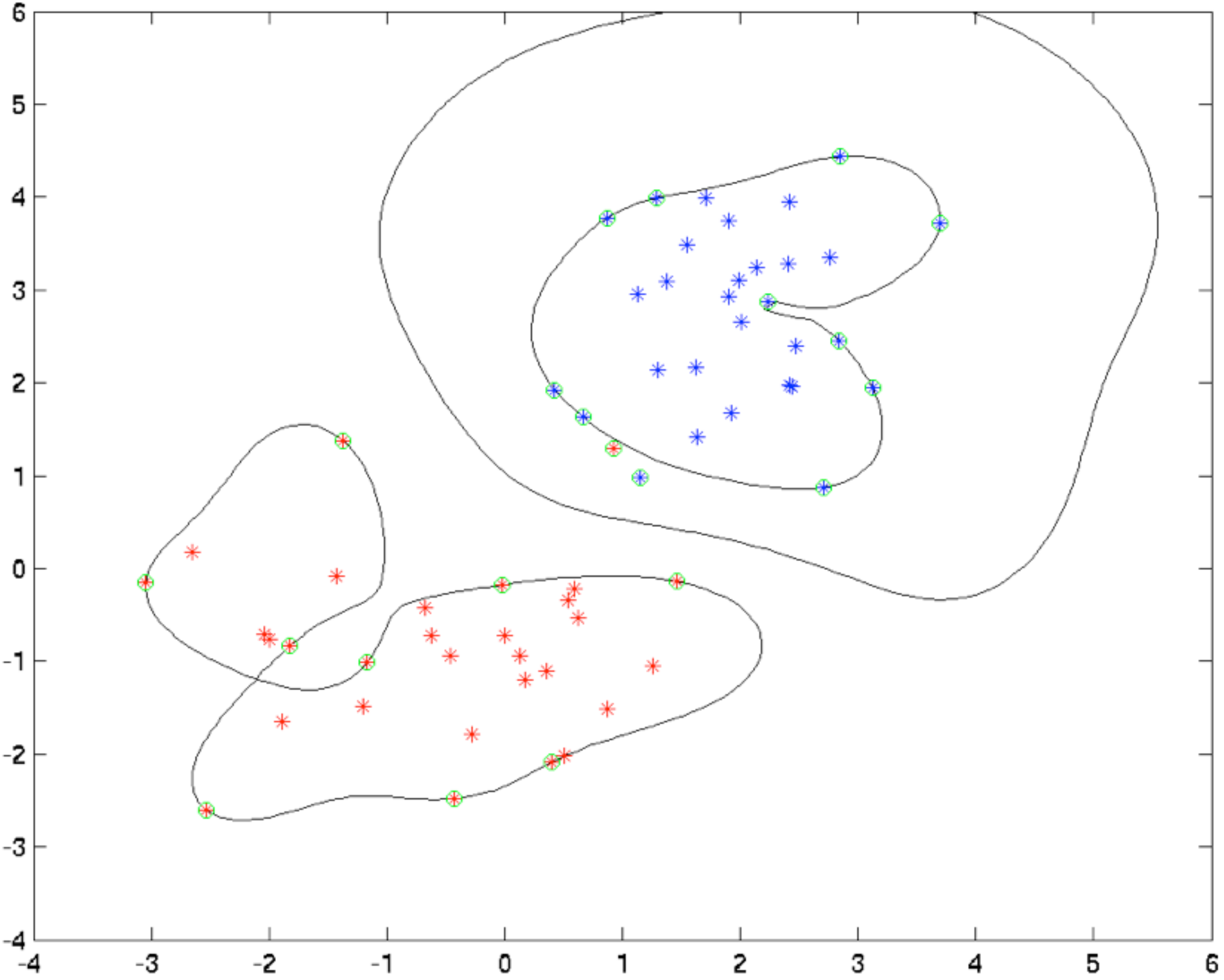
slide by Alex Smola



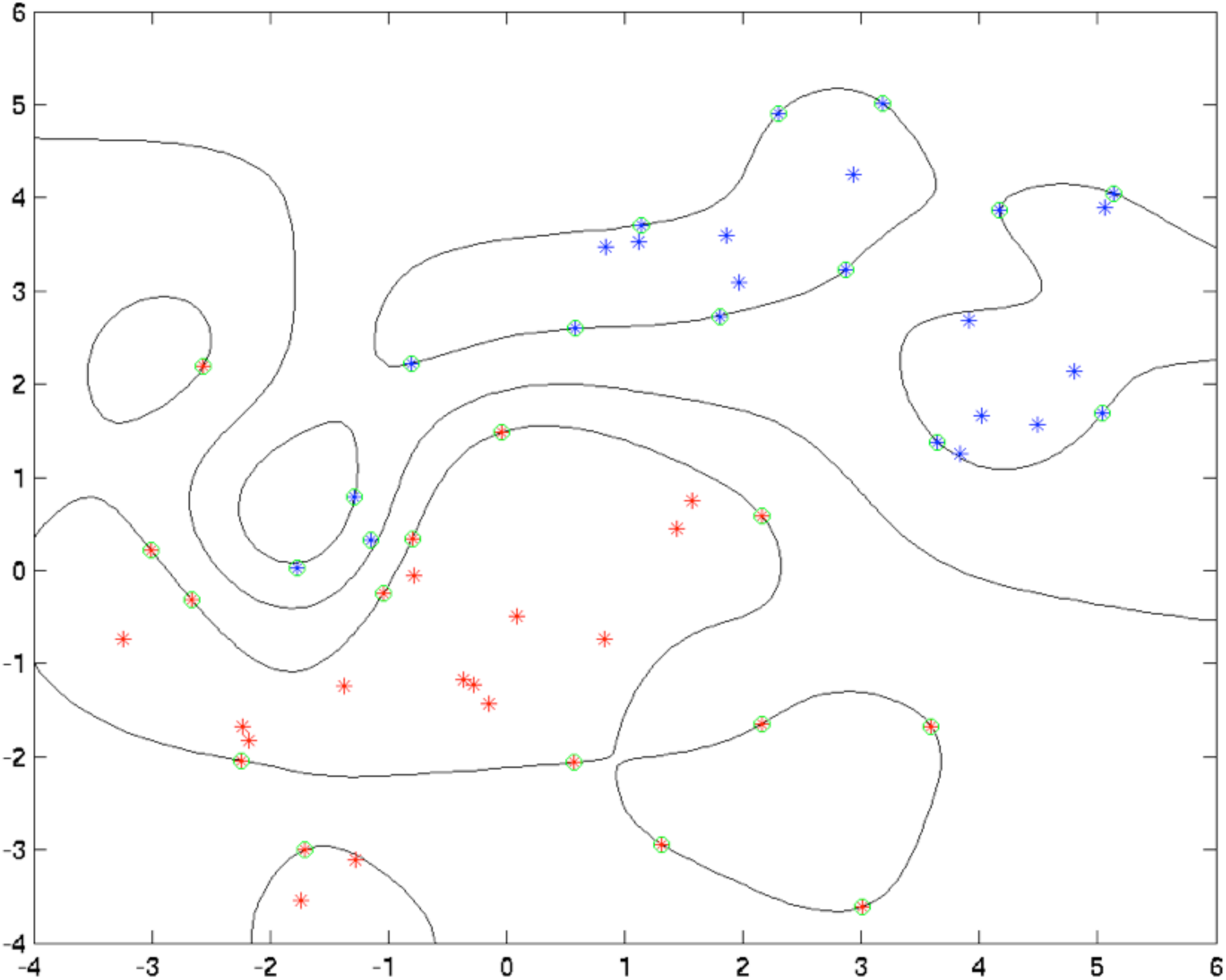
slide by Alex Smola



slide by Alex Smola



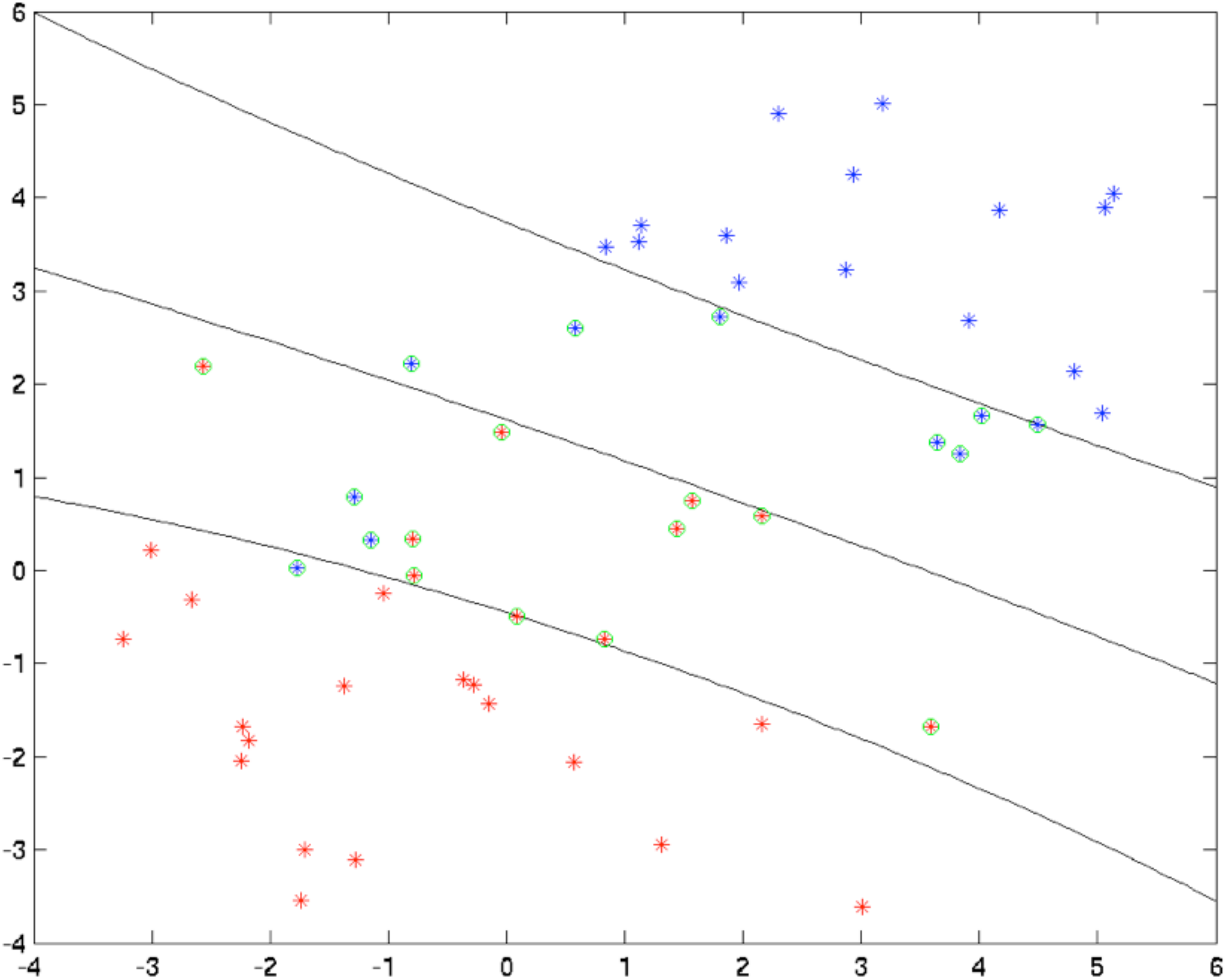
slide by Alex Smola



slide by Alex Smola

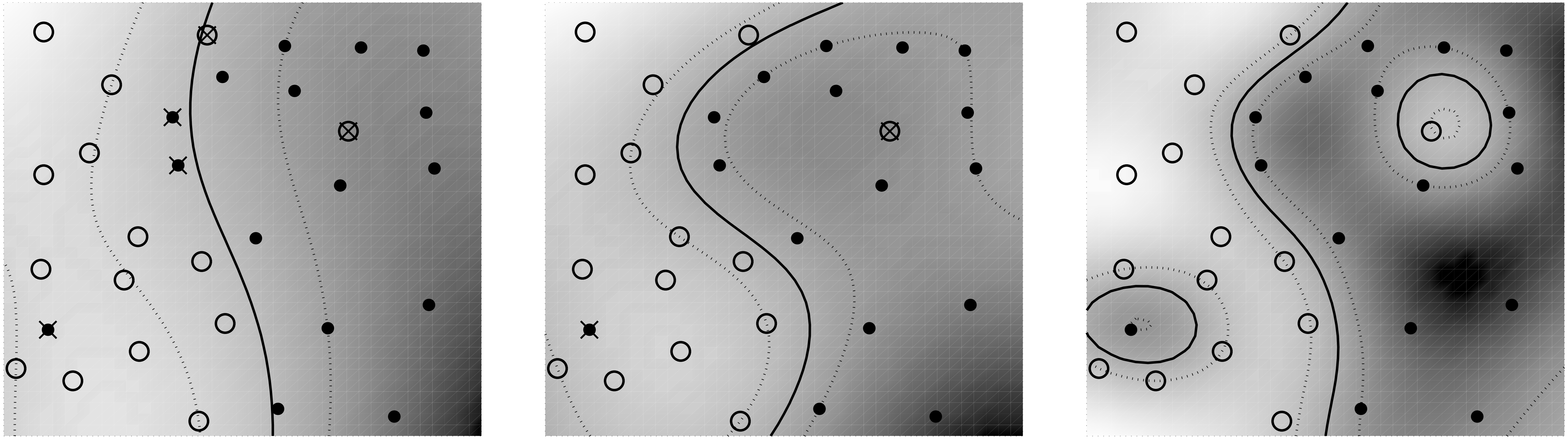
And now with a very wide kernel

slide by Alex Smola



slide by Alex Smola

Nonlinear Separation



- Increasing C allows for more nonlinearities
- Decreases number of errors
- SV boundary need not be contiguous
- Kernel width adjusts function class

slide by Alex Smola

Overfitting?

- **Huge feature space with kernels: should we worry about overfitting?**
- SVM objective seeks a solution with large margin
  - Theory says that large margin leads to good generalization (we will see this in a couple of lectures)
- **But everything overfits sometimes!!!**
- Can control by:
  - Setting C
  - Choosing a better Kernel
  - Varying parameters of the Kernel (width of Gaussian, etc.)

slide by Alex Smola

Risk and Loss

slide by Alex Smola

Loss function point of view

- Constrained quadratic program

$$\underset{w,b}{\text{minimize}} \;\; \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

$$\text{subject to } y_i\left[\langle w, x_i\rangle + b\right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

- Risk minimization setting

$$\underset{w,b}{\text{minimize}} \;\; \frac{1}{2}\|w\|^2 + C\sum_i \max\left[0, 1 - y_i\left[\langle w, x_i\rangle + b\right]\right]$$

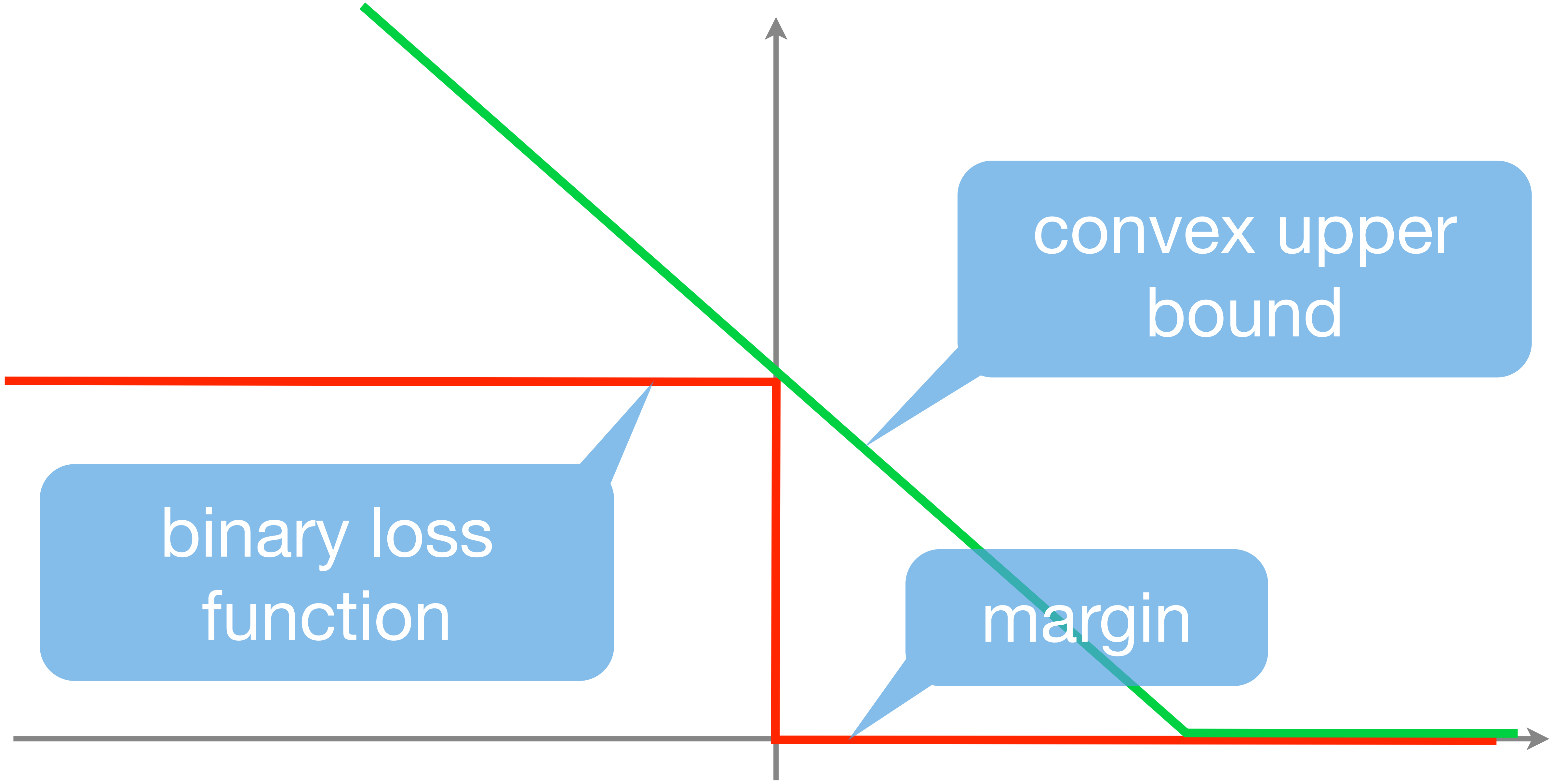
empirical risk

Follows from finding minimal slack variable for given (w,b) pair.

slide by Alex Smola

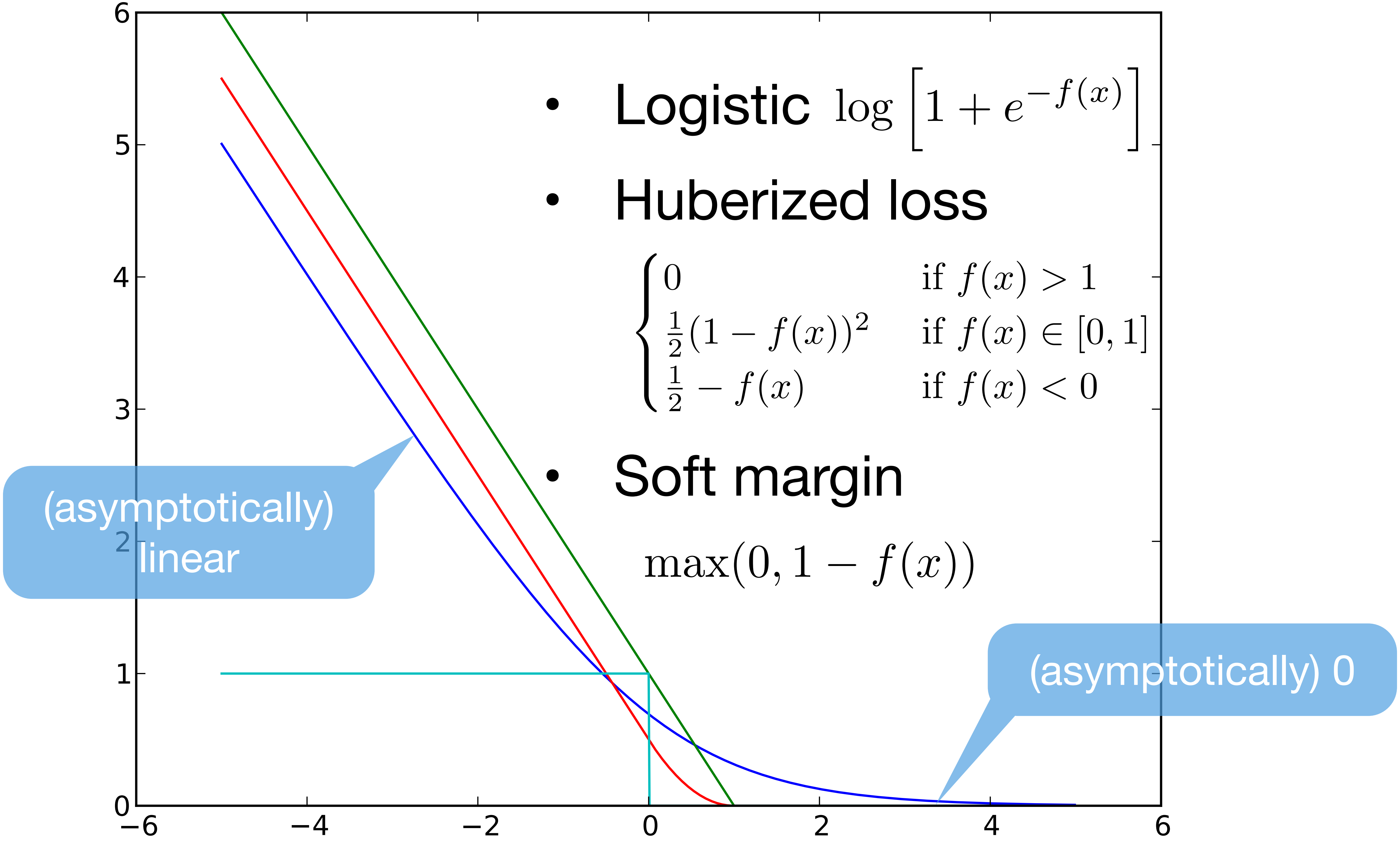
Soft margin as proxy for binary

- Soft margin loss $\max(0, 1 - yf(x))$
- Binary loss $\{yf(x) < 0\}$



slide by Alex Smola

More loss functions



- Logistic $\log\left[1+e^{-f(x)}\right]$
- Huberized loss

$$\begin{cases} 0 & \text{if } f(x) > 1 \\ \frac{1}{2}(1-f(x))^2 & \text{if } f(x) \in [0,1] \\ \frac{1}{2} - f(x) & \text{if } f(x) < 0 \end{cases}$$

- Soft margin

$$\max(0, 1 - f(x))$$

(asymptotically) linear

(asymptotically) 0

slide by Alex Smola

Risk minimization view

- Find function *f* minimizing classification error

$$R[f] := \mathbf{E}_{x,y\sim p(x,y)}\left[\{yf(x) > 0\}\right]$$

- Compute empirical average

$$R_{\mathrm{emp}}[f] := \frac{1}{m}\sum_{i=1}^{m}\{y_i f(x_i) > 0\}$$

  - Minimization is nonconvex
  - Overfitting as we minimize empirical error
- Compute convex upper bound on the loss
- Add regularization for capacity control

regularization

$$R_{\mathrm{reg}}[f] := \frac{1}{m}\sum_{i=1}^{m}\max(0, 1 - y_i f(x_i)) + \lambda\Omega[f]$$

how to control λ

slide by Alex Smola

Support Vector Regression

Regression Estimation

- Find function f minimizing regression error

$$R[f] := \mathbf{E}_{x,y \sim p(x,y)} \left[l(y, f(x)) \right]$$

- Compute empirical average

$$R_{\mathrm{emp}}[f] := \frac{1}{m} \sum_{i=1}^{m} l(y_i, f(x_i))$$

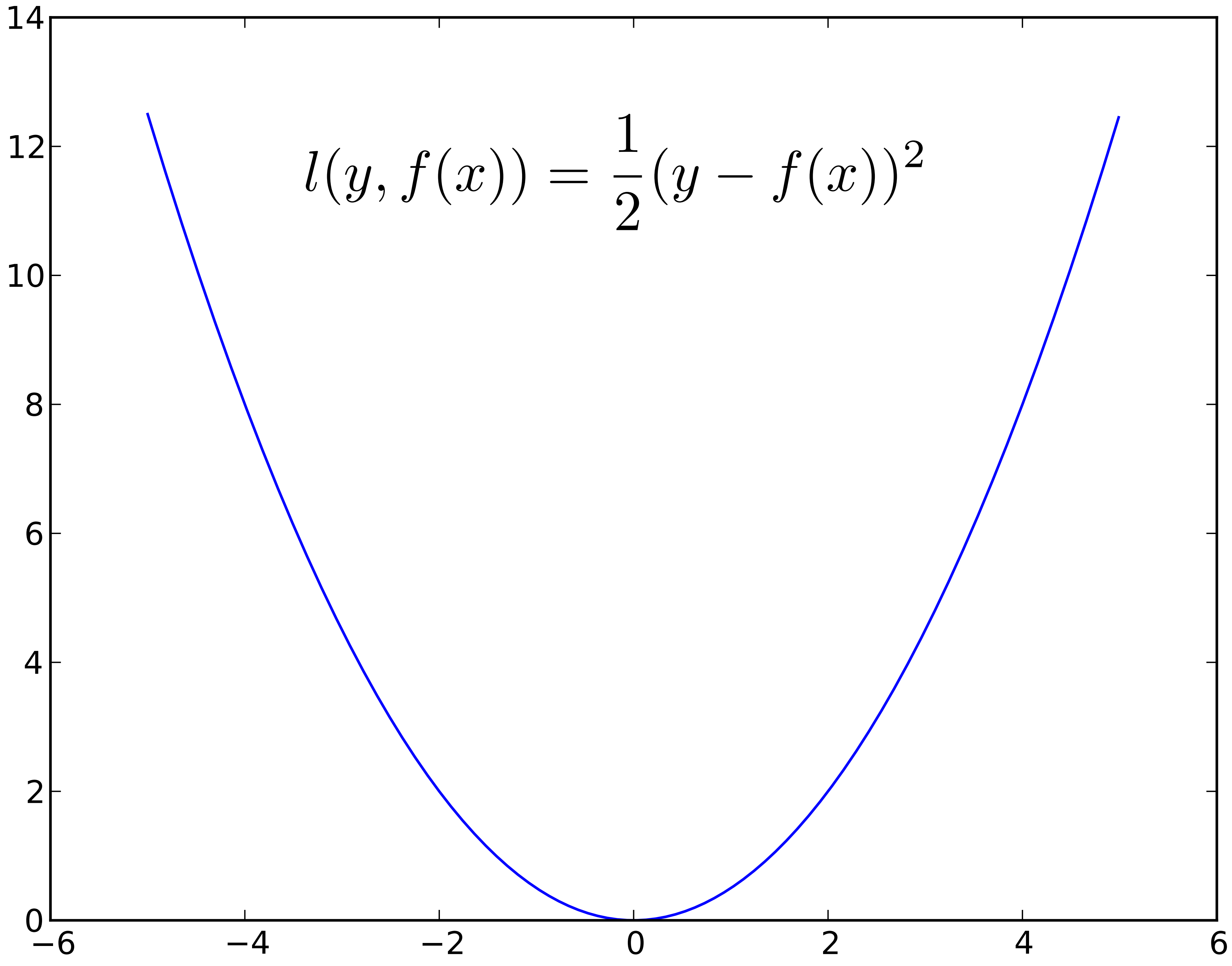
Overfitting as we minimize empirical error

- Add regularization for capacity control

$$R_{\mathrm{reg}}[f] := \frac{1}{m} \sum_{i=1}^{m} l(y_i, f(x_i)) + \lambda \Omega[f]$$

slide by Alex Smola

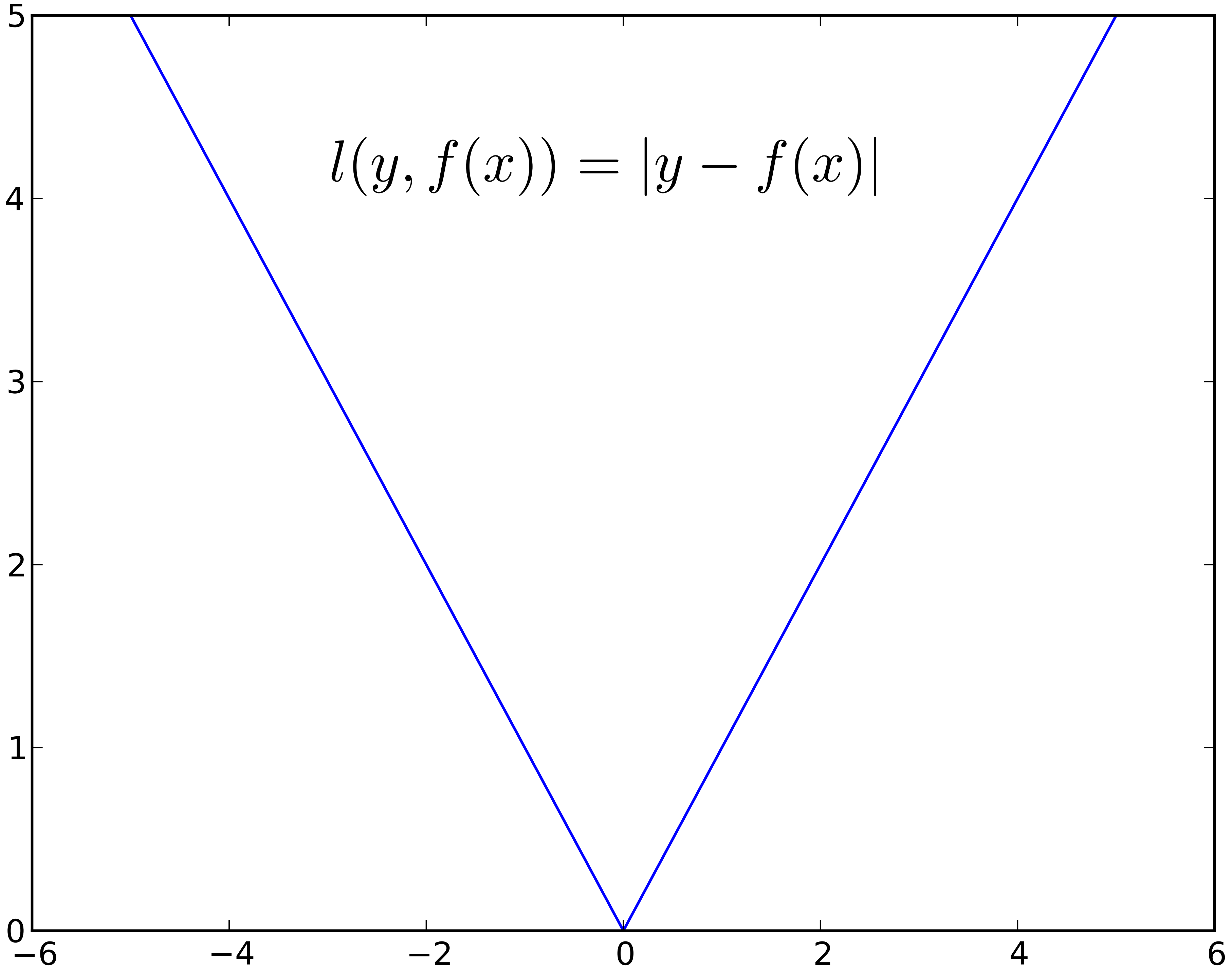
Squared loss



$$l(y, f(x)) = \frac{1}{2}(y - f(x))^2$$

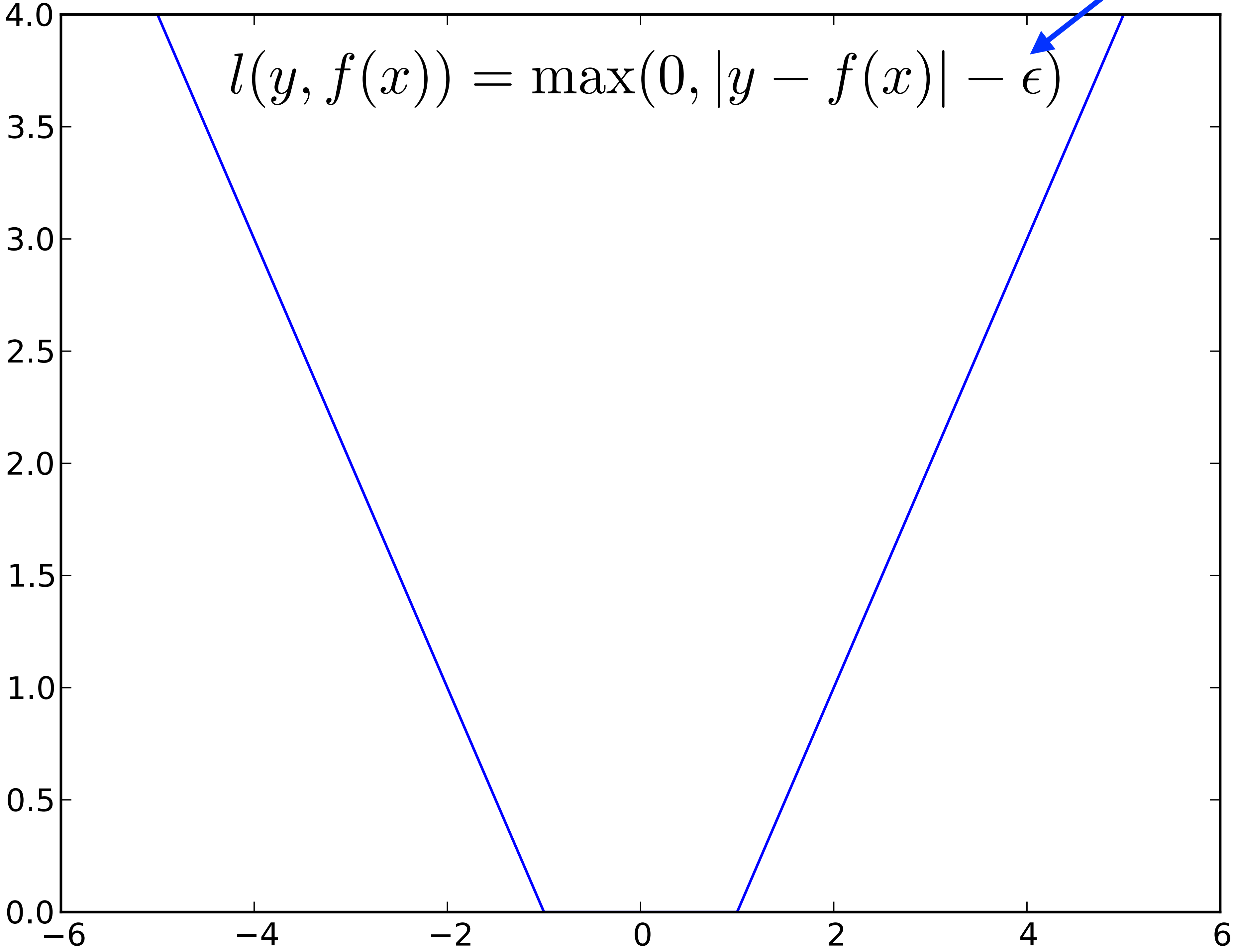
slide by Alex Smola

l1 loss



slide by Alex Smola

ε-insensitive Loss



allow some deviation without a penalty

slide by Alex Smola

Penalized least mean squares

- Optimization problem

$$\underset{w}{\text{minimize}}\ \frac{1}{2m}\sum_{i=1}^{m}(y_i - \langle x_i, w\rangle)^2 + \frac{\lambda}{2}\left\|w\right\|^2$$

- Solution

$$\partial_w\left[\ldots\right] = \frac{1}{m}\sum_{i=1}^{m}\left[x_i x_i^\top w - x_i y_i\right] + \lambda w$$

$$= \left[\frac{1}{m}XX^\top + \lambda\mathbf{1}\right] w - \frac{1}{m}Xy = 0$$

$$\text{hence } w = \left[XX^\top + \lambda m\mathbf{1}\right]^{-1} Xy$$

Outer product matrix in X

Conjugate Gradient
Sherman Morrison Woodbury

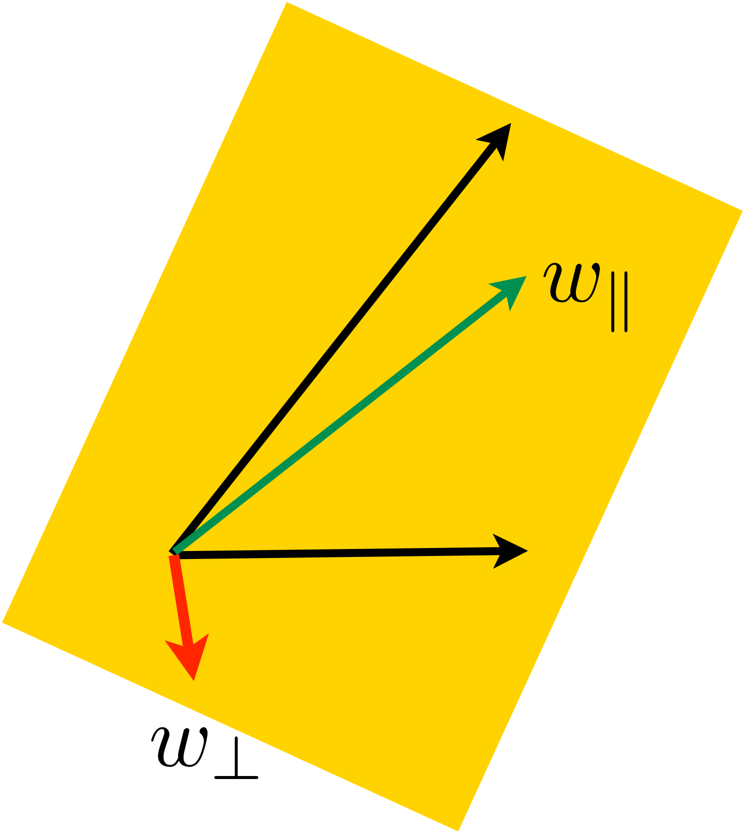
slide by Alex Smola

Penalized least mean squares ... now with kernels

- Optimization problem

$$\underset{w}{\text{minimize}}\ \frac{1}{2m}\sum_{i=1}^{m}(y_i - \langle\phi(x_i), w\rangle)^2 + \frac{\lambda}{2}\|w\|^2$$

- Representer Theorem (Kimeldorf & Wahba, 1971)



$$\|w\|^2 = \underbrace{\|w_{\|}\|^2}_{\text{empirical risk dependent}} + \|w_{\perp}\|^2$$

slide by Alex Smola

Penalized least mean squares ... now with kernels

- Optimization problem

$$\underset{w}{\text{minimize}} \frac{1}{2m} \sum_{i=1}^{m} (y_i - \langle \phi(x_i), w \rangle)^2 + \frac{\lambda}{2} \|w\|^2$$

- Representer Theorem (Kimeldorf & Wahba, 1971)
  - Optimal solution is in span of data $w = \sum_i \alpha_i \phi(x_i)$
  - Proof - risk term only depends on data via $\phi(x_i)$
  - Regularization ensures that orthogonal part is 0
- Optimization problem in terms of w

$$\underset{\alpha}{\text{minimize}} \frac{1}{2m} \sum_{i=1}^{m} \Big(y_i - \sum_j K_{ij} \alpha_j \Big)^2 + \frac{\lambda}{2} \sum_{i,j} \alpha_i \alpha_j K_{ij}$$

solve for $\alpha = (K + m\lambda \mathbf{1})^{-1} y$ as linear system

Penalized least mean squares ... now with kernels

- Optimization problem

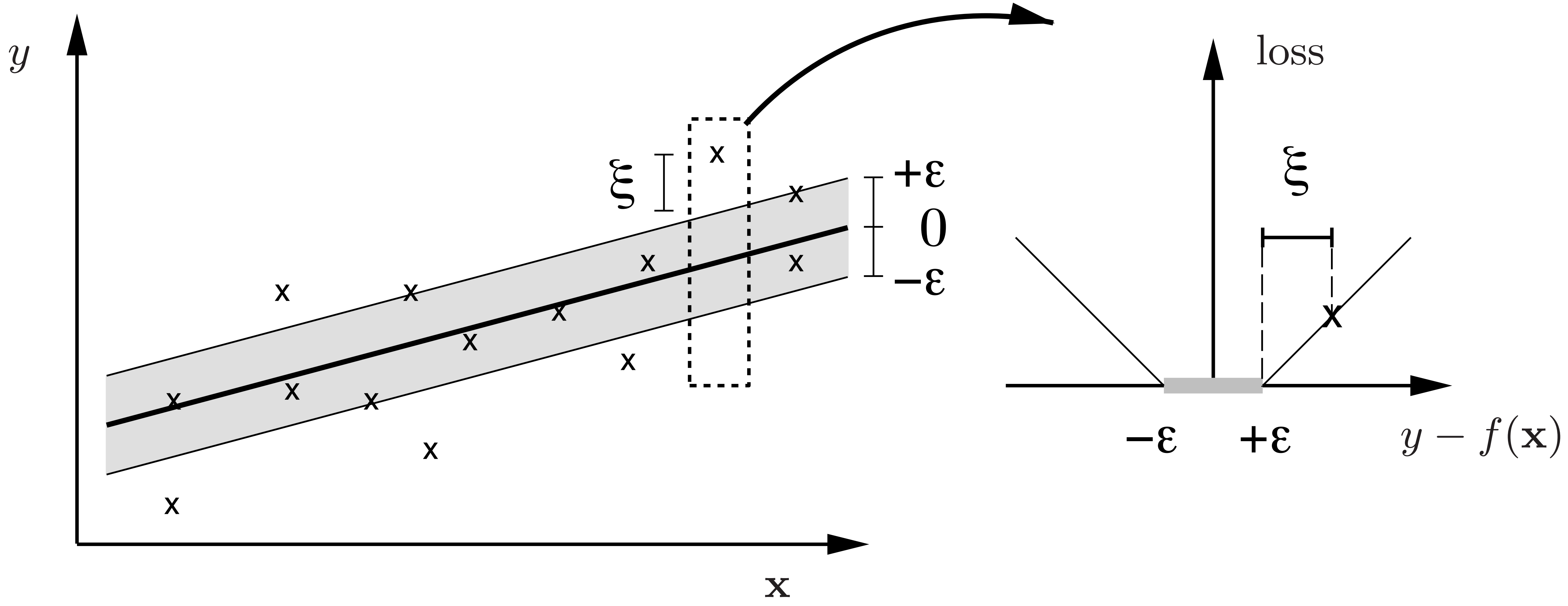
$$\underset{w}{\text{minimize}} \frac{1}{2m} \sum_{i=1}^{m} (y_i - \langle \phi(x_i), w \rangle)^2 + \frac{\lambda}{2} \|w\|^2$$

- Representer Theorem (Kimeldorf & Wahba, 1971)
  - Optimal solution is in span of data $\boxed{w = \sum_i \alpha_i \phi(x_i)}$
  - Proof - risk term only depends on data via $\phi(x_i)$
  - Regularization ensures that orthogonal part is 0
- Optimization problem in terms of w

$$\underset{\alpha}{\text{minimize}} \frac{1}{2m} \sum_{i=1}^{m} \left(y_i - \sum_j K_{ij} \alpha_j \right)^2 + \frac{\lambda}{2} \sum_{i,j} \alpha_i \alpha_j K_{ij}$$

solve for $\alpha = (K + m\lambda\mathbf{1})^{-1} y$ as linear system

SVM Regression (ϵ-insensitive loss)



don’t care about deviations within the tube

slide by Alex Smola

SVM Regression (є-insensitive loss)

- Optimization Problem (as constrained QP)

$$\underset{w,b}{\text{minimize}} \ \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}[\xi_i + \xi_i^*]$$

$$\text{subject to} \ \langle w, x_i\rangle + b \le y_i + \epsilon + \xi_i \ \text{and} \ \xi_i \ge 0$$

$$\langle w, x_i\rangle + b \ge y_i - \epsilon - \xi_i^* \ \text{and} \ \xi_i^* \ge 0$$

- Lagrange Function

$$L = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}[\xi_i + \xi_i^*] - \sum_{i=1}^{m}[\eta_i\xi_i + \eta_i^*\xi_i^*] +$$

$$\sum_{i=1}^{m}\alpha_i\left[\langle w, x_i\rangle + b - y_i - \epsilon - \xi_i\right] + \sum_{i=1}^{m}\alpha_i^*\left[y_i - \epsilon - \xi_i^* - \langle w, x_i\rangle - b\right]$$

slide by Alex Smola

SVM Regression (ϵ-insensitive loss)

- First order conditions

$$\partial_w L = 0 = w + \sum_i [\alpha_i - \alpha_i^*] \, x_i$$

$$\partial_b L = 0 = \sum_i [\alpha_i - \alpha_i^*]$$

$$\partial_{\xi_i} L = 0 = C - \eta_i - \alpha_i$$

$$\partial_{\xi_i^*} L = 0 = C - \eta_i^* - \alpha_i^*$$

- Dual problem

$$\underset{\alpha,\alpha^*}{\text{minimize}} \;\; \frac{1}{2}(\alpha - \alpha^*)^\top K(\alpha - \alpha^*) + \epsilon 1^\top (\alpha + \alpha^*) + y^\top (\alpha - \alpha^*)$$

$$\text{subject to } 1^\top (\alpha - \alpha^*) = 0 \text{ and } \alpha_i, \alpha_i^* \in [0, C]$$

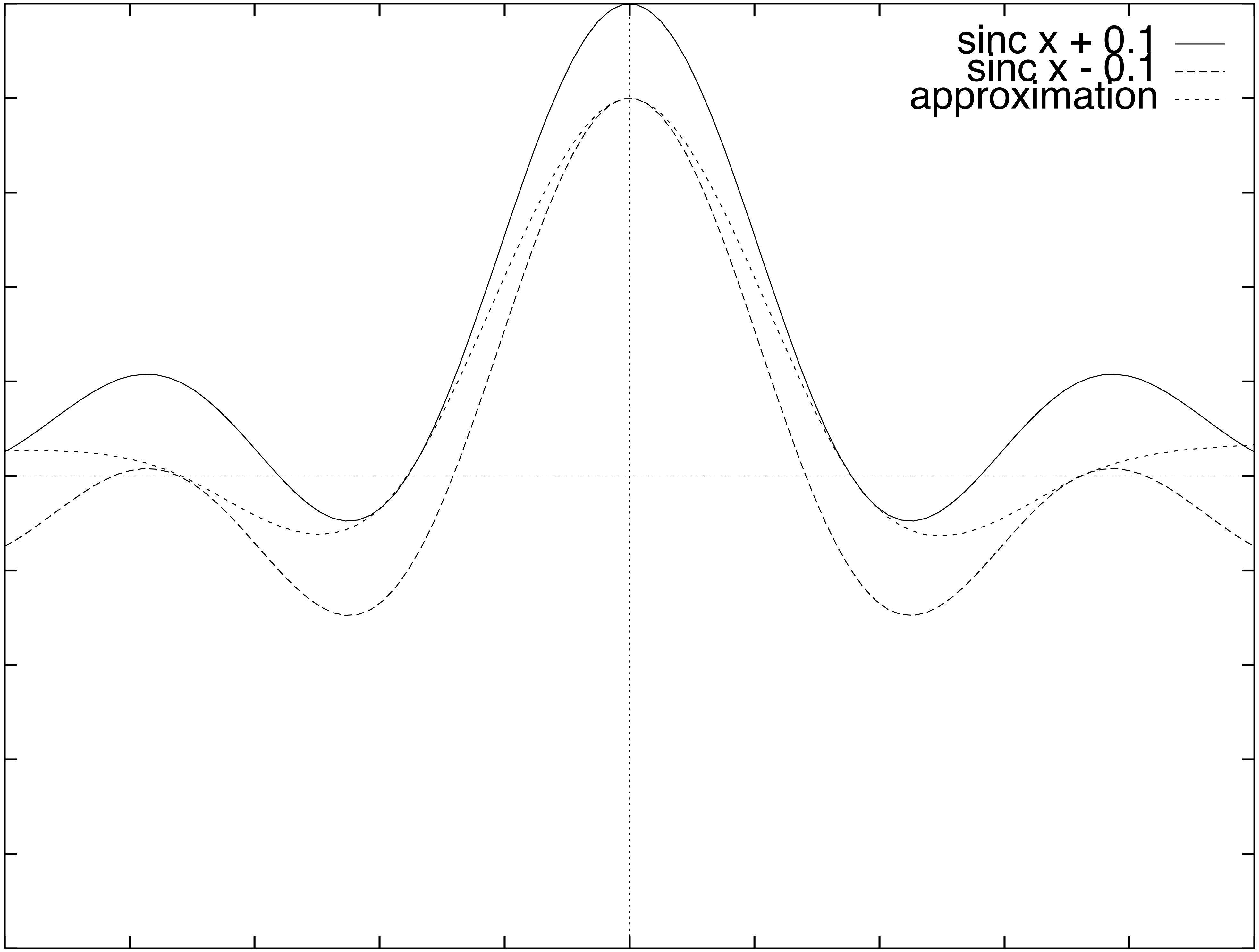
slide by Alex Smola

Properties

- Ignores ‘typical’ instances with small error
- Only upper or lower bound active at any time
- QP in 2n variables as cheap as SVM problem
- Robustness with respect to outliers
  - l_1 loss yields same problem without epsilon
  - Huber’s robust loss yields similar problem but with added quadratic penalty on coefficients

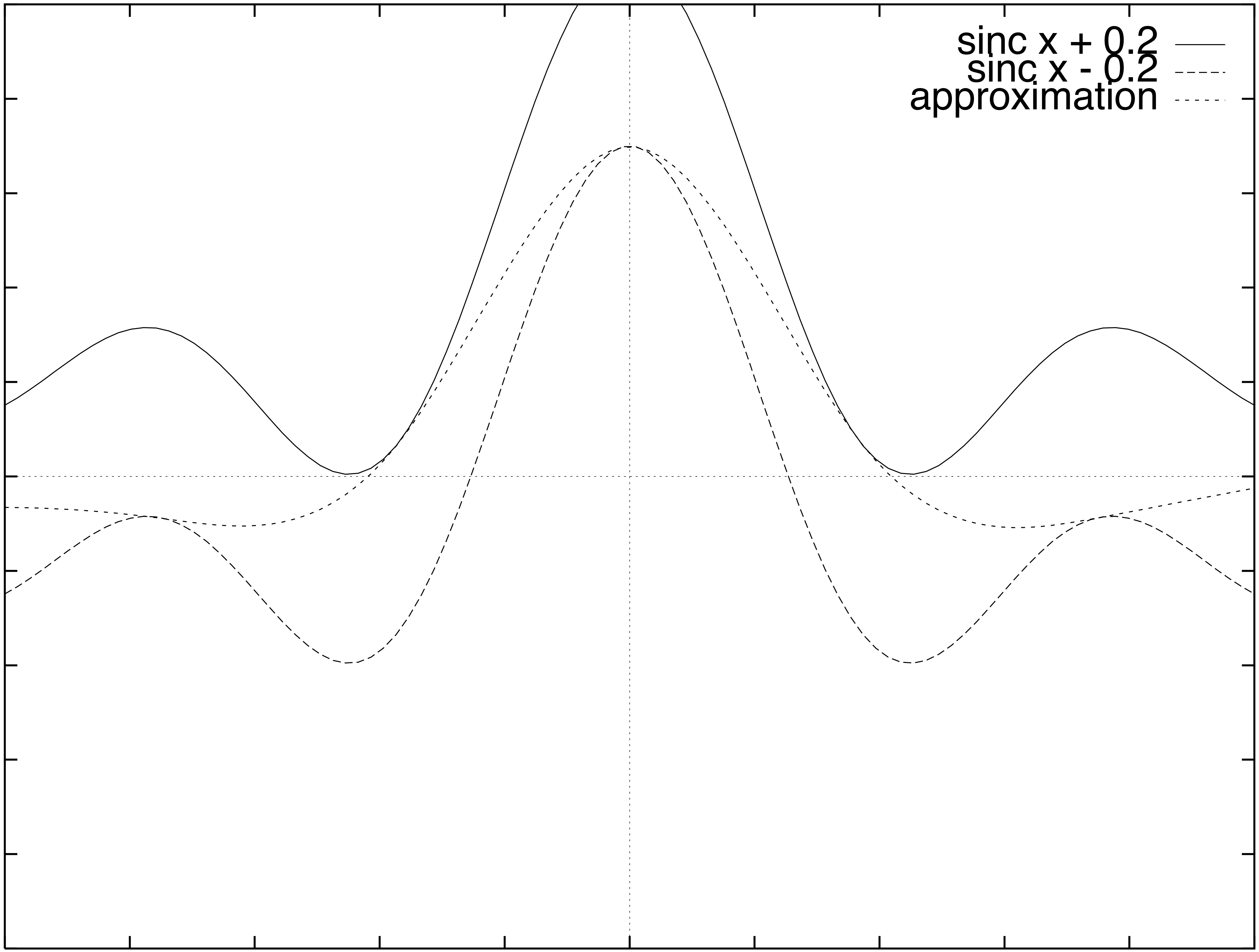
slide by Alex Smola

Regression example



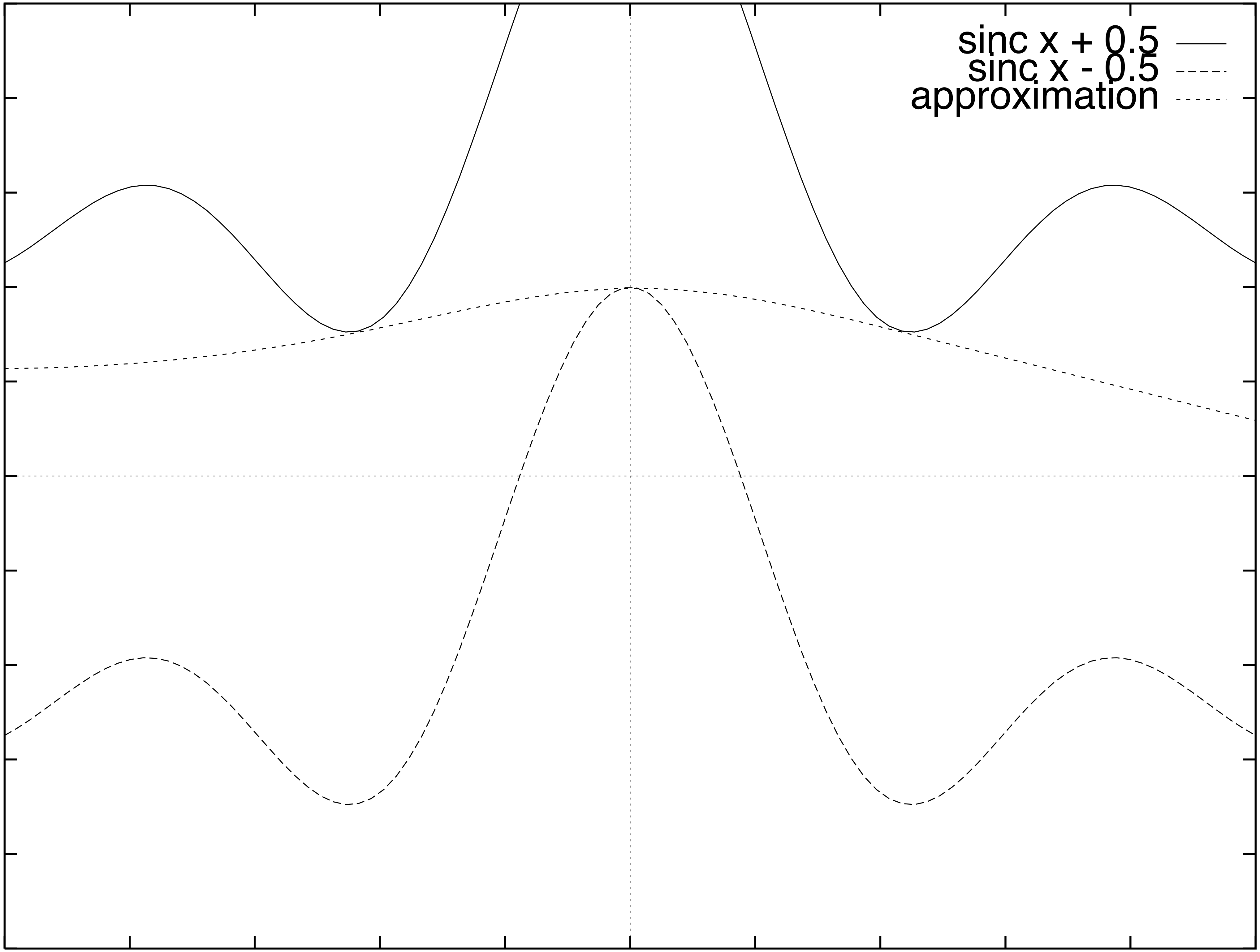
slide by Alex Smola

Regression example



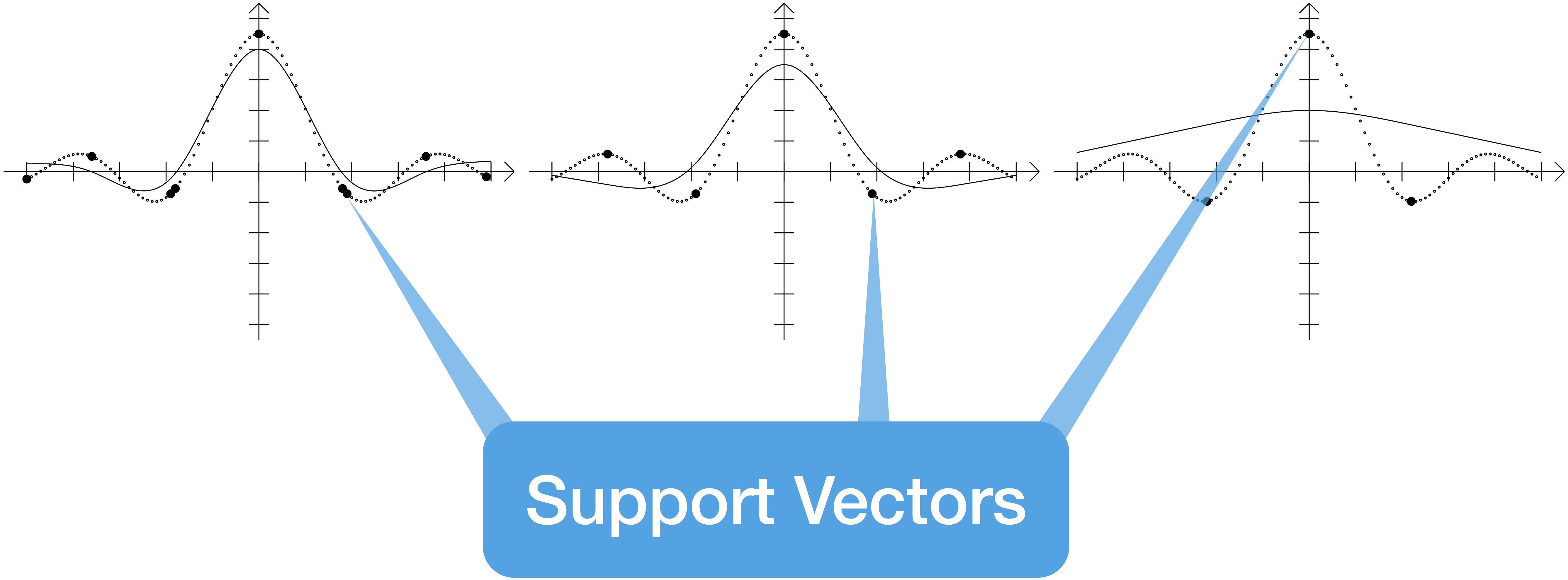
slide by Alex Smola

Regression example



slide by Alex Smola

Regression example

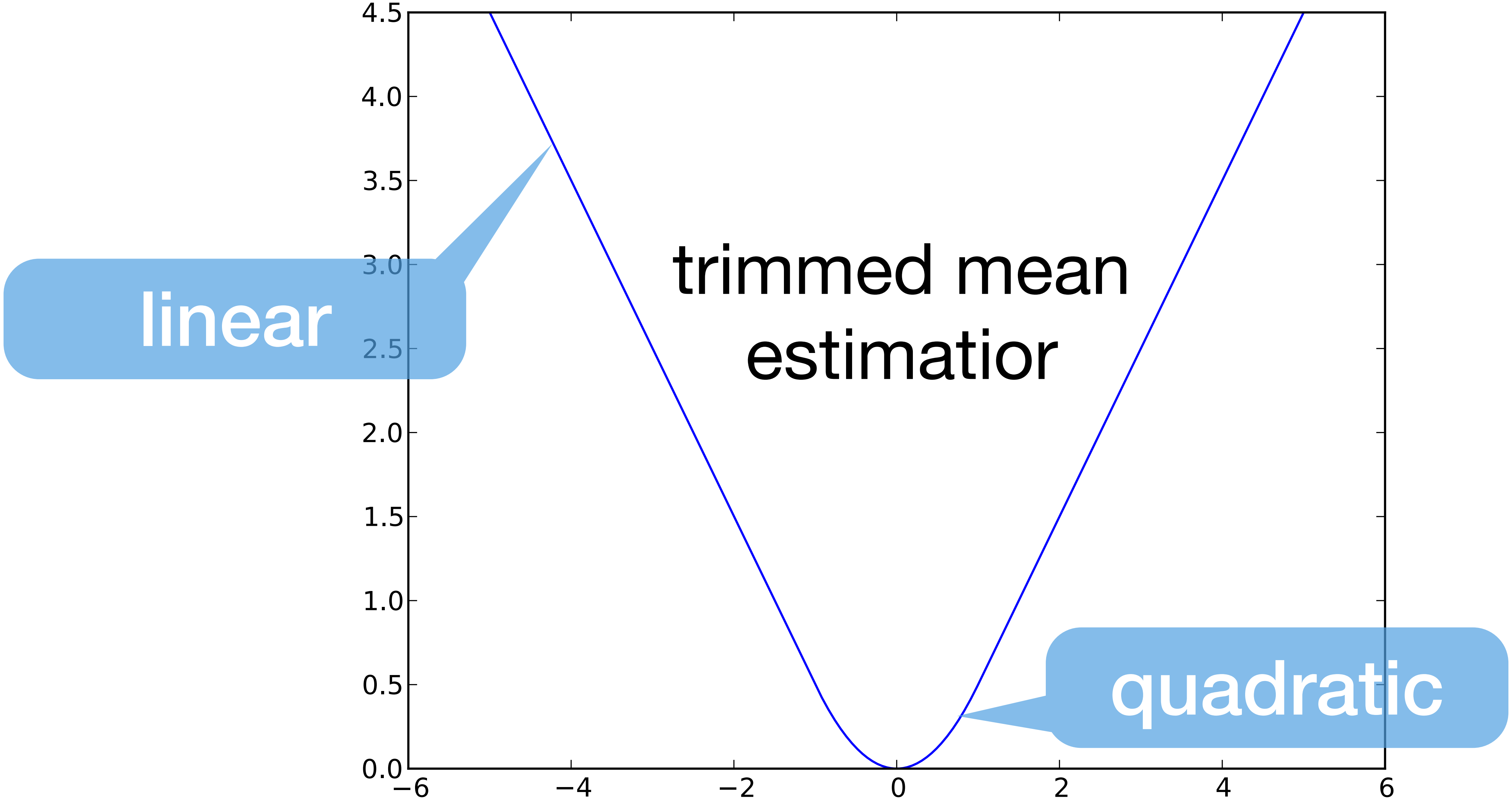


Support Vectors

slide by Alex Smola

Huber’s robust loss

$$l(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2 & \text{if } |y - f(x)| < 1 \\ |y - f(x)| - \frac{1}{2} & \text{otherwise} \end{cases}$$



linear

trimmed mean estimatior

quadratic

slide by Alex Smola

Summary

- **Advantages:**
  - Kernels allow very flexible hypotheses
  - Poly-time exact optimization methods rather than approximate methods
  - Soft-margin extension permits mis-classified examples
  - Variable-sized hypothesis space
  - Excellent results (1.1% error rate on handwritten digits vs. LeNet's 0.9%)
- **Disadvantages:**
  - Must choose kernel parameters
  - Very large problems computationally intractable
  - Batch algorithm

slide by Sanja Fidler

Software

- SVM^{light}: one of the most widely used SVM packages. Fast optimization, can handle very large datasets, C++ code.
- LIBSVM
- Both of these handle multi-class, weighted SVM for unbalanced data, etc.
- There are several new approaches to solving the SVM objective that can be much faster:
  - Stochastic subgradient method (discussed in a few lectures)
  - Distributed computation (also to be discussed)
- See http://mloss.org, “machine learning open source software”

slide by Alex Smola

Next Lecture:

Decision Trees