Support Vector Machines, Kernel Logistic Regression, and Boosting

Trevor Hastie
Statistics Department
Stanford University

Collaborators: Brad Efron, Jerome Friedman, Saharon Rosset, Rob Tibshirani, Ji Zhu

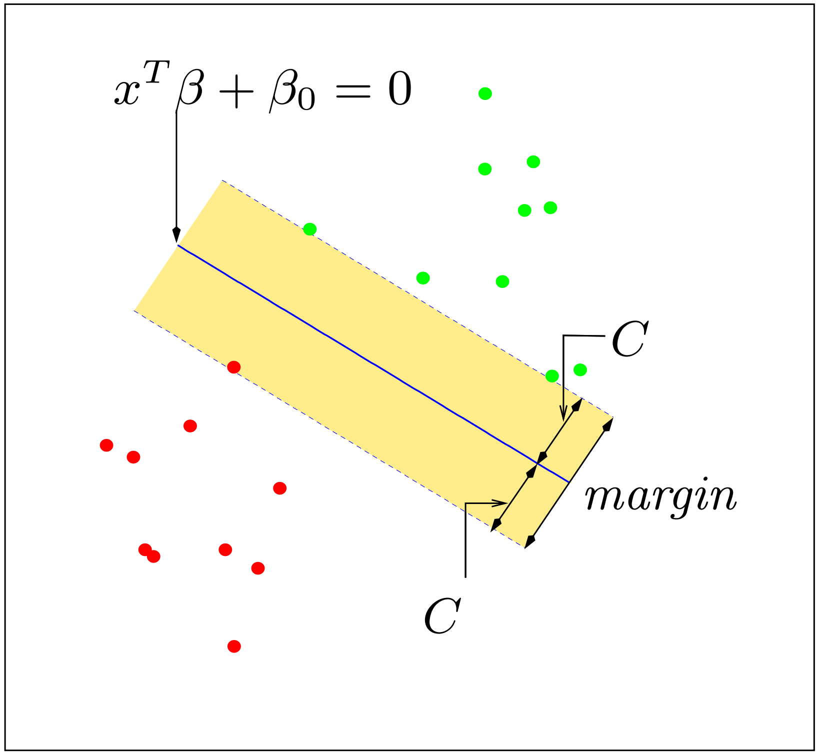
http://www-stat.stanford.edu/∼hastie/Papers/svmtalk.pdf

Outline

- ✔ Optimal separating hyperplanes and relaxations
- ✔ SVMs: nonlinear generalizations of separating hyperplanes
- ✔ SVM as a function estimation problem
- ✔ Kernel logistic regression
- ✘ Reproducing kernel Hilbert spaces
- ✔ Connections between SVM, KLR and Boosting.

First part based on work by Vapnik (1996), Wahba (1990), Evgeniou, Pontil, and Poggio (1999); described in Hastie, Tibshirani and Friedman (2001) *Elements of Statistical Learning*, Springer, NY. Gunnar Rätsch and coworkers have also made coonection between SVMs and Boosting.

Maximum Margin Classifier



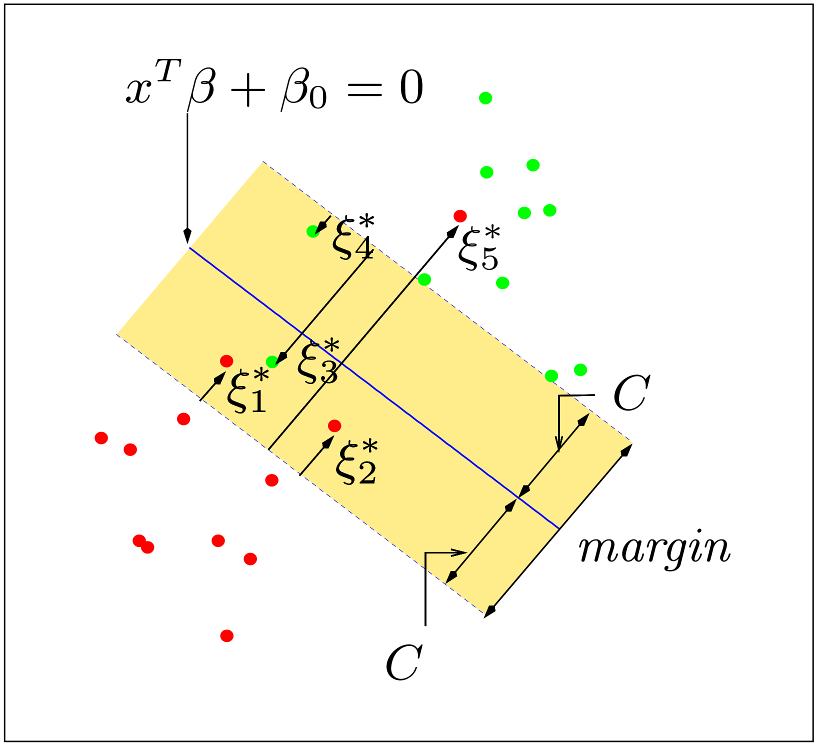
Vapnik(1995)

$x_i \in \mathbb{R}^p$, $y_i \in \{-1, 1\}$

$$\max_{\beta, \beta_0, \|\beta\|=1} C$$

$$\text{subject to} \qquad y_i(x_i^T\beta + \beta_0) \geq C, \; i = 1, \ldots, N.$$

Overlapping Classes

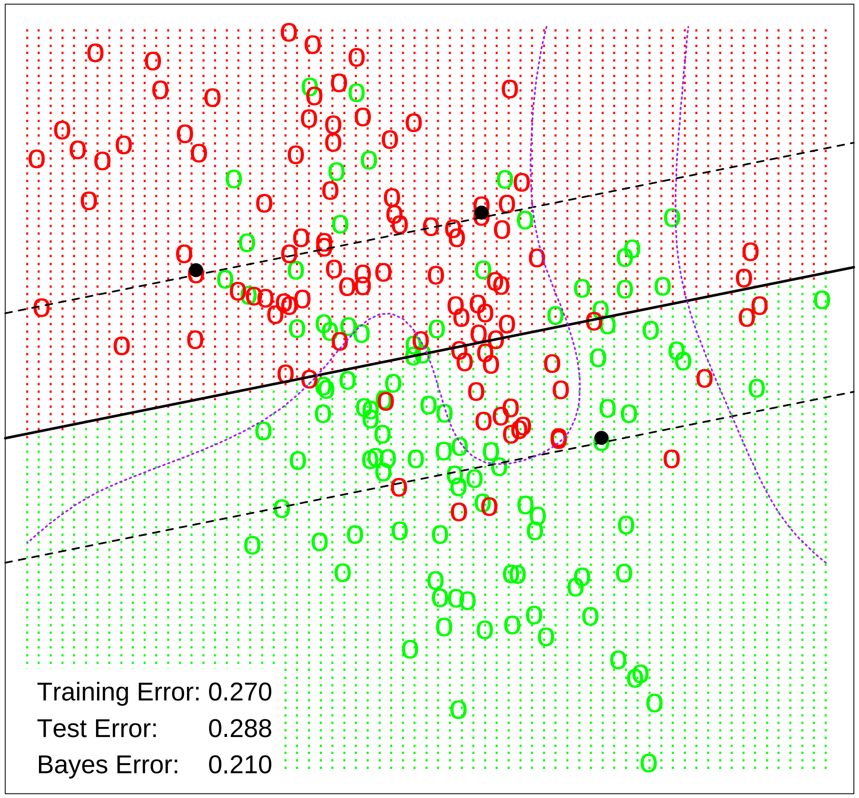


$\xi_i^* = C\xi_i$

$$\max_{\beta,\beta_0,\|\beta\|=1} C$$

subject to $y_i(x_i^T\beta + \beta_0) \geq C(1-\xi_i)$, $\xi_i \geq 0$, $\sum_i \xi_i \leq B$

Example



Fitted function is $\hat{f}(x) = x^T\hat{\beta} + \hat{\beta}_0$

Resulting classifier is $\hat{G}(x) = \text{sign}[\hat{f}(x)]$

Quadratic Programming Solution

After a lot of *stuff* we arrive at a Lagrange dual

$$L_D = \sum_{i=1}^N \alpha_i - \frac{1}{2}\sum_{i=1}^N \sum_{i'=1}^N \alpha_i \alpha_{i'} y_i y_{i'} x_i^T x_{i'}$$

which we maximize subject to constraints (involving B as well).

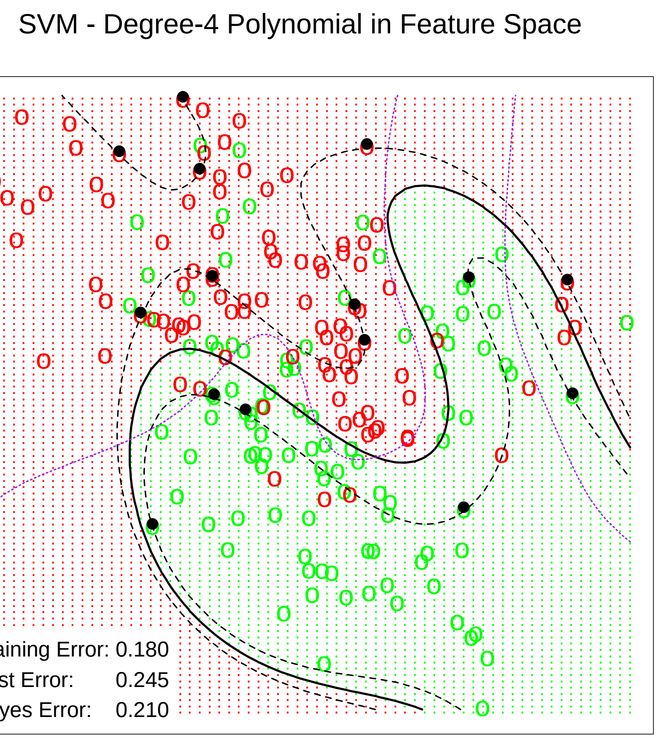
The solution is expressed in terms of fitted Lagrange multipliers $\hat{\alpha}_i$:

$$\hat{\beta} = \sum_{i=1}^N \hat{\alpha}_i y_i x_i$$

Some fraction of $\hat{\alpha}_i$ are exactly zero (from KKT conditions); the x_i for which $\hat{\alpha}_i > 0$ are called support points $\mathcal{S}$.

$$\hat{f}(x) = x^T \hat{\beta} + \hat{\beta}^0 = \sum_{i \in \mathcal{S}} \hat{\alpha}_i y_i x^T x_i + \hat{\beta}^0$$

Flexible Classifiers



Enlarge the feature space via basis expansions, e.g. polynomials of total degree 4. $h(x) = (h_1(x), h_2(x), \ldots, h_M(x))$

$$\hat{f}(x) = h(x)^T \hat{\beta} + \hat{\beta}_0$$

SVM

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i'=1}^{N} \alpha_i \alpha_{i'} y_i y_{i'} \langle h(x_i), h(x_{i'}) \rangle$$

$$\begin{aligned} f(x) &= h(x)^T \beta + \beta_0 \\ &= \sum_{i=1}^{N} \alpha_i y_i \langle h(x), h(x_i) \rangle + \beta_0. \end{aligned}$$

L_D and constraints involve $h(x)$ only through inner-products

$$K(x, x') = \langle h(x), h(x') \rangle$$

Given a suitable positive kernel $K(x, x')$, don't need $h(x)$ at all!

$$\hat{f}(x) = \sum_{i \in \mathcal{S}} \hat{\alpha}_i y_i K(x, x_i) + \hat{\beta}_0$$

Popular Kernels

$K(x, x')$ is a symmetric, positive (semi-)definite function.

$$d\text{th deg. poly.: } K(x, x') = (1 + \langle x, x' \rangle)^d$$

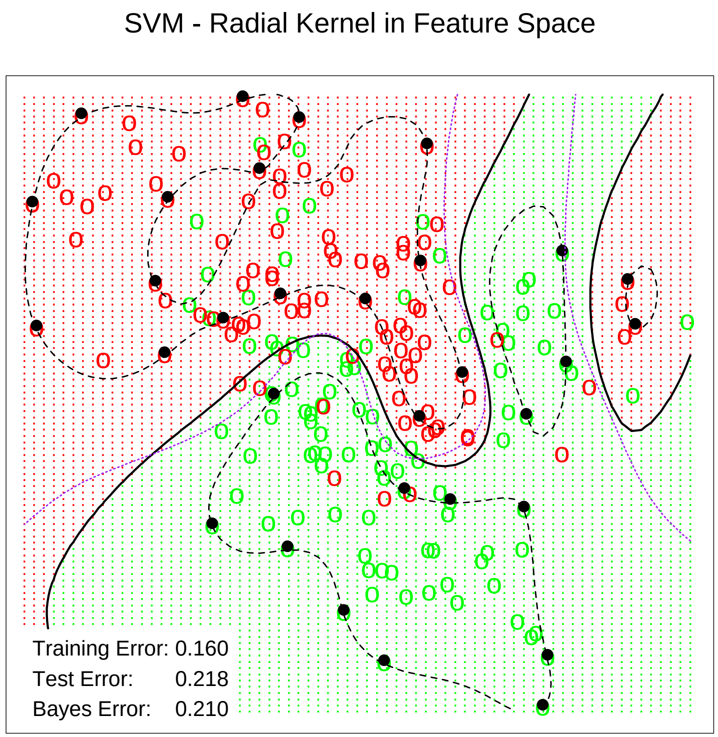
$$\text{radial basis: } K(x, x') = \exp(-\|x - x'\|^2/c)$$

Example: 2nd degree polynomial in $\mathbb{R}^2$.

$$\begin{aligned} K(x, x') &= (1 + \langle x, x' \rangle)^2 \\ &= (1 + x_1 x_1' + x_2 x_2')^2 \\ &= 1 + 2x_1 x_1' + 2x_2 x_2' + (x_1 x_1')^2 + (x_2 x_2')^2 + 2x_1 x_1' x_2 x_2' \end{aligned}$$

Then $M = 6$, and if we choose
$h_1(x) = 1$, $h_2(x) = \sqrt{2}x_1$, $h_3(x) = \sqrt{2}x_2$, $h_4(x) = x_1^2$, $h_5(x) = x_2^2$,
and $h_6(x) = \sqrt{2}x_1 x_2$,
then $K(x, x') = \langle h(x), h(x') \rangle$.

Dim $h(x)$ infinite



- Fraction of support points depends on overlap; here 45%.
- The smaller B, the smaller the overlap, and more wiggly the function.
- B controls generalization error.

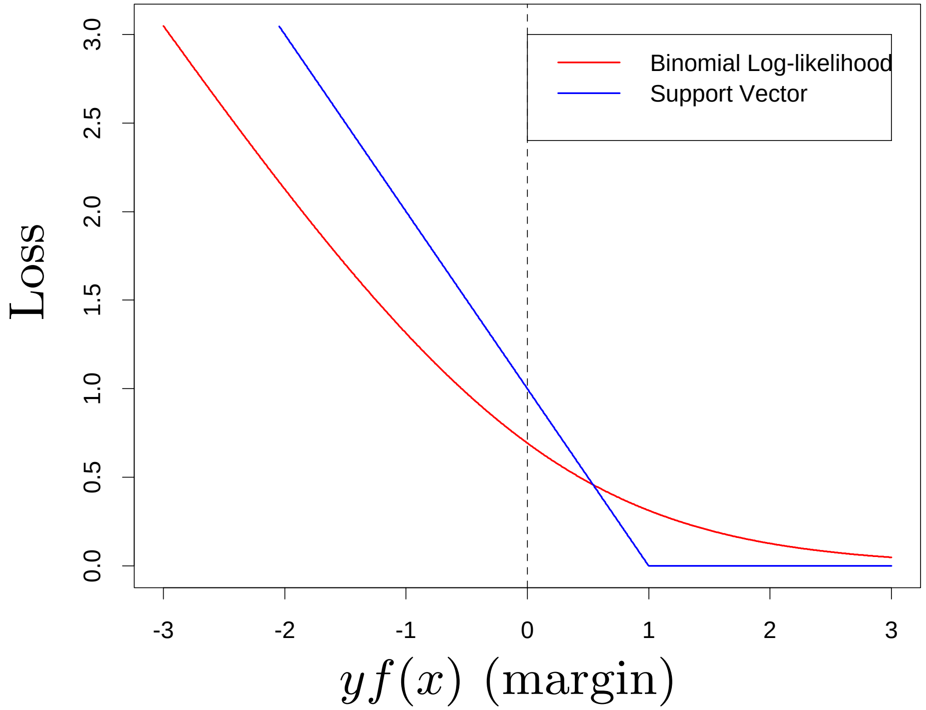
Curse of Dimensionality

Support Vector Machines can suffer in high dimensions.

		Test Error (SE)	
	Method	No Noise Features	Six Noise Features
1	SV Classifier	0.450 (0.003)	0.472 (0.003)
2	SVM/poly 2	0.078 (0.003)	0.152 (0.004)
3	SVM/poly 5	0.180 (0.004)	0.370 (0.004)
4	SVM/poly 10	0.230 (0.003)	0.434 (0.002)
5	BRUTO	0.084 (0.003)	0.090 (0.003)
6	MARS	0.156 (0.004)	0.173 (0.005)
	Bayes	0.029	0.029

The addition of 6 noise features to the 4-dimensional feature space causes the performance of the SVM to degrade. The true decision boundary is the surface of a sphere, hence a quadratic monomial (additive) function is sufficient.

SVM via Loss + Penalty



With $f(x) = h(x)^T\beta + \beta_0$ and $y_i \in \{-1, 1\}$, consider

$$\min_{\beta_0,\,\beta} \sum_{i=1}^{N} [1 - y_i f(x_i)]_+ + \lambda\|\beta\|^2$$

Solution identical to SVM solution, with $\lambda = \lambda(B)$.

In general $\min_{\beta_0,\,\beta} \sum_{i=1}^{N} L[y_i, f(x_i)] + \lambda\|\beta\|^2$

Loss Functions

For $Y \in \{-1, 1\}$

Log-likelihood: $L[Y, f(X)] = \log\left(1 + e^{-Yf(X)}\right)$

- (negative) binomial log-likelihood or deviance.
- estimates the logit

$$f(X) = \log \frac{\Pr(Y = 1|X)}{\Pr(Y = -1|X)}$$

SVM: $L[Y, f(X)] = (1 - Yf(X))_+$.

- Called "hinge loss"
- Estimates the classifier (threshold)

$$C(x) = \text{sign}\left(\Pr(Y = 1|X) - \frac{1}{2}\right)$$

SVM and Function Estimation

SVM with general kernel K minimizes:

$$\sum_{i=1}^{N}(1 - y_i f(x_i))_+ + \lambda \|f\|^2_{\mathcal{H}_K}$$

with $f = b + h$, $h \in \mathcal{H}_K$, $b \in \mathcal{R}$. $\mathcal{H}_K$ is the reproducing kernel Hilbert space (RKHS) of functions generated by the kernel K. The norm $\|f\|_{\mathcal{H}_K}$ is generally interpreted as a roughness penalty.

More generally we can optimize

$$\sum_{i=1}^{N} L(y_i, f(x_i)) + \lambda \|f\|^2_{\mathcal{H}_K}$$

The solutions have the form

$$\hat{f}(x) = \hat{b} + \sum_{i=1}^{N} \hat{\alpha}_i K(x, x_i),$$

a finite expansion in the representers $K(x, x_i)$.

Aside: RKHS

Function space $\mathcal{H}_K$ generated by a positive (semi-) definite function $K(x, x')$.

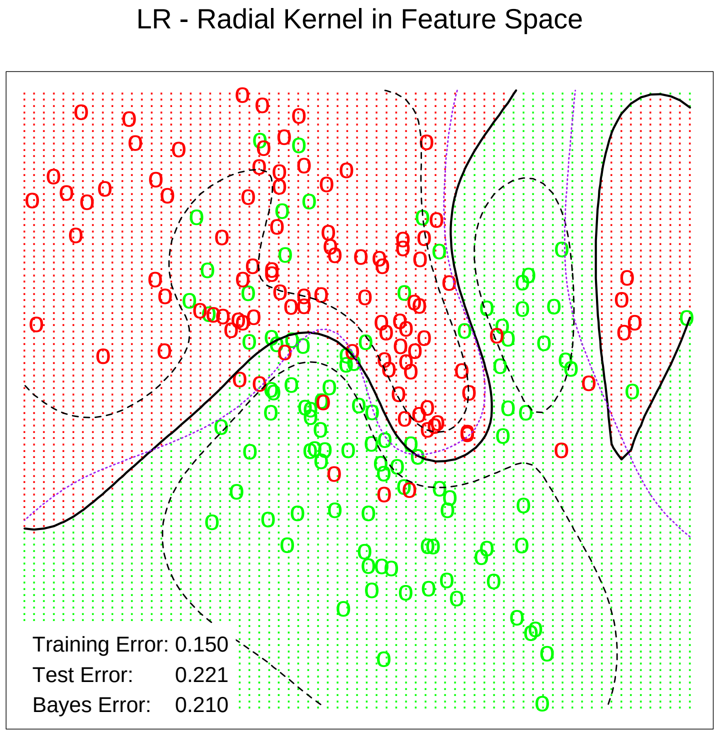
$$\text{Eigen expansion: } K(x, y) = \sum_{i=1}^{\infty} \gamma_i \phi_i(x) \phi_i(y)$$

with $\gamma_i \geq 0$, $\sum_{i=1}^{\infty} \gamma_i^2 < \infty$. $f \in \mathcal{H}_K$ if

$$\begin{aligned} f(x) &= \sum_{i=1}^{\infty} c_i \phi_i(x) \\ c_i &= \int \phi_i(t) f(t) dt \\ ||f||_{\mathcal{H}_K}^2 &\stackrel{\text{def}}{=} \sum_{i=1}^{\infty} c_i^2 / \gamma_i < \infty \end{aligned}$$

The squared norm $J(f) = ||f||_{\mathcal{H}_K}^2$ is viewed as a roughness penalty.

Kernel Logistic Regression



- Replace $(1 - yf)_+$ with $\ln(1 + e^{-yf})$, the binomial deviance.
- $\hat{\Pr}(Y = 1|x) = e^{\hat{f}(x)}/(1 + e^{\hat{f}(x)})$, so class probabilities directly available.
- We have graphed the 0.5 (solid), 0.25, and 0.75 (broken) contours of $\hat{\Pr}(Y = 1|x)$.

Comparison: KLR vs SVM

- The classification performance is very similar.
- Has limiting optimal margin properties (next slide).
- Provides estimates of the class probabilities. Often these are more useful than the classifications (e.g. credit risk scoring).
- Generalizes naturally to M-class classification through kernel multi-logit regression:

$$\Pr(Y = j|x) = \frac{e^{f_j(x)}}{e^{f_1(x)} + \cdots e^{f_M(x)}}$$

with $\sum_m f_m(x) = 0$. Fit using multinomial log-likelihood and penalty $\sum_{m=1}^{M} \|f_m\|_{\mathcal{H}_K}$.

KLR and Optimal Margins

Suppose $h(x)$ is rich enough so that $f(x) = h(x)^T\beta + \beta_0$ can separate the training data.

Consider $\hat{\beta}(\lambda)$, the solution to

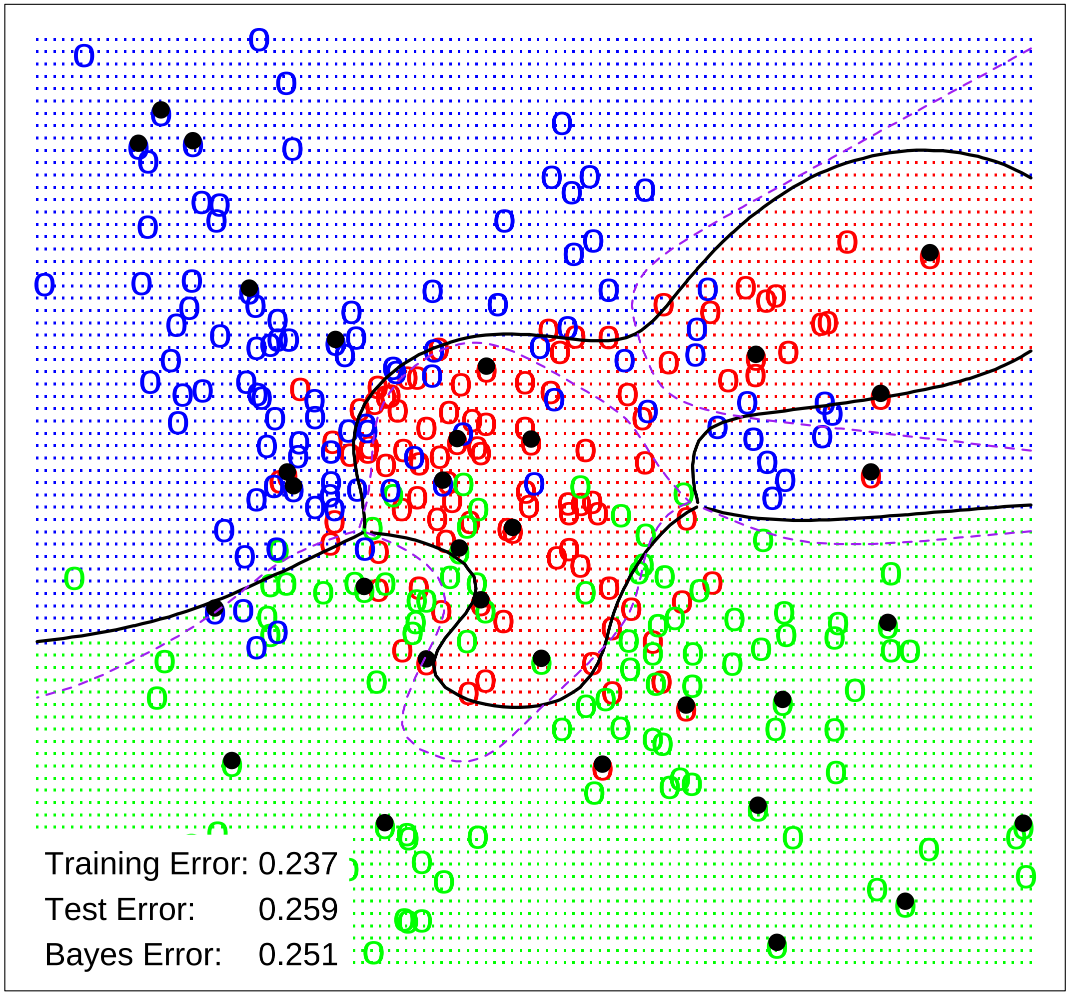
$$\min_{\beta_0,\, \beta} \sum_{i=1}^{N} L[y_i, f(x_i)] + \lambda\|\beta\|^2,$$

where L is the binomial deviance (negative log-likelihood).

Theorem (Rosset, Zhu & Hastie 2002)

$\lim_{\lambda\to 0} \hat{\beta}(\lambda) = \beta^*$, the maximum margin solution.

Multi-class IVM - with 32 import points

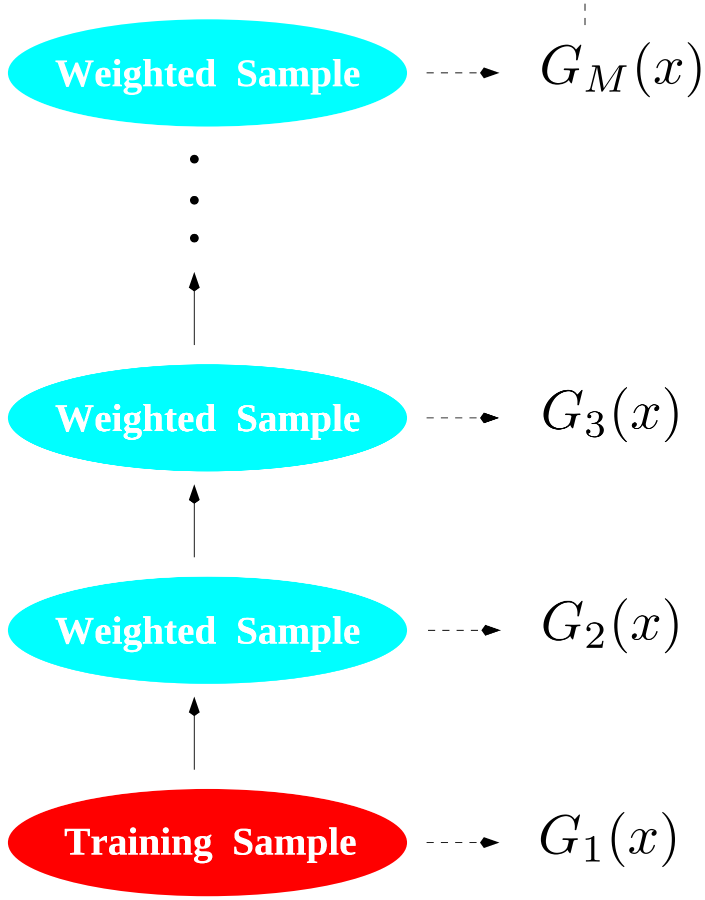


Disadvantages: KLR vs SVM

- Computationally more expensive $O(N^3)$ versus $O(N^2 m)$, where m is the number of support points. In noisy problems, m can be large, approx $N/2$.
- With KLR fit $\hat{f}(x) = \hat{b} + \sum_{i=1}^{N} \hat{\alpha}_i K(x, x_i)$, all the $\hat{\alpha}_i$ are typically nonzero. For the SVM, only the support points have nonzero $\hat{\alpha}_i$. This allows for a useful data compression and quicker lookup.
- SVMs are **hot** right now, while logistic regression is a **traditional statistical tool**.

FINAL CLASSIFIER

$$G(x) = \text{sign}\left[\sum_{m=1}^{M} \alpha_m G_m(x)\right]$$



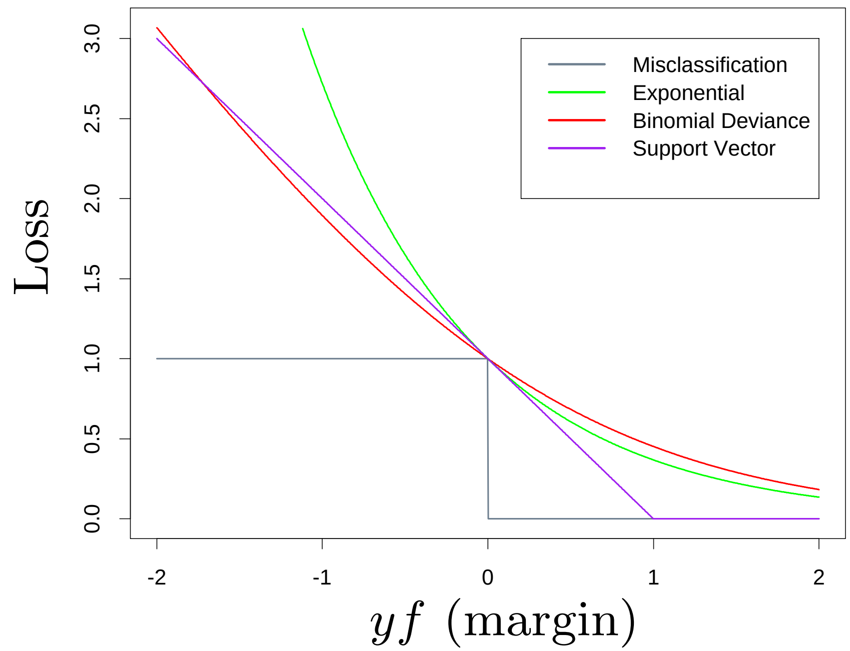
Boosting

Classifiers are trained on weighted versions of the dataset, and then combined to produce a final prediction.

AdaBoost (Freund & Schapire, 1996)

- Start with weights $w_i = 1/N \; \forall i = 1, \dots, N$. $y_i \in \{-1, 1\}$.
- Repeat for $m = 1, 2, \dots, M$:
  - Estimate the weak learner $f_m(x) \in \{-1, 1\}$ from the training data with weights w_i.
  - Compute $e_m = E_w[1(y \neq f_m(x)]$, $c_m = \log((1 - e_m)/e_m)$.
  - Set $w_i \leftarrow w_i \exp[c_m \cdot 1(y_i \neq f_m(x_i)]$, $i = 1, 2, \dots N$, and renormalize so that $\sum_i w_i = 1$.
- Output the majority weight classifier $C(x) = \text{sign}[\sum_{m=1}^{M} c_m f_m(x)]$.

SVM, KLR and Boosting?



- Boosting builds a sequence of models $f_J(x) = \sum_{j=1}^{J} g_j(x)$, where each $g_j(x)$ is a "weak" classifier fit to weighted training data.
- Even though at stage J, $f_J(x)$ may have zero training errors, boosting increases the "margin".
- Actually, boosting is fitting the model $f(x) = \log \Pr(Y = 1|x)/\Pr(Y = -1|x)$ by stagewise optimization of the loss function $L[Y, f(X)] = \exp[-Y f(X)]$ (FHT, 2000), *Ann. Stat.*

Boosting and L_1 Penalized Fitting

In a restricted setting where

- the base learners are chosen from a fixed set of basis functions;
- the increments at each boosting step are shrunk towards zero;
- + a few mild assumptions (yeah, right!),

the boosting sequence corresponds to a sequence (as λ varies) of solutions to the L_1 penalized optimization problem

$$\min_{\beta} \sum_{i=1}^{N} L[y_i, f(x_i)] + \lambda \|\beta\|_1$$

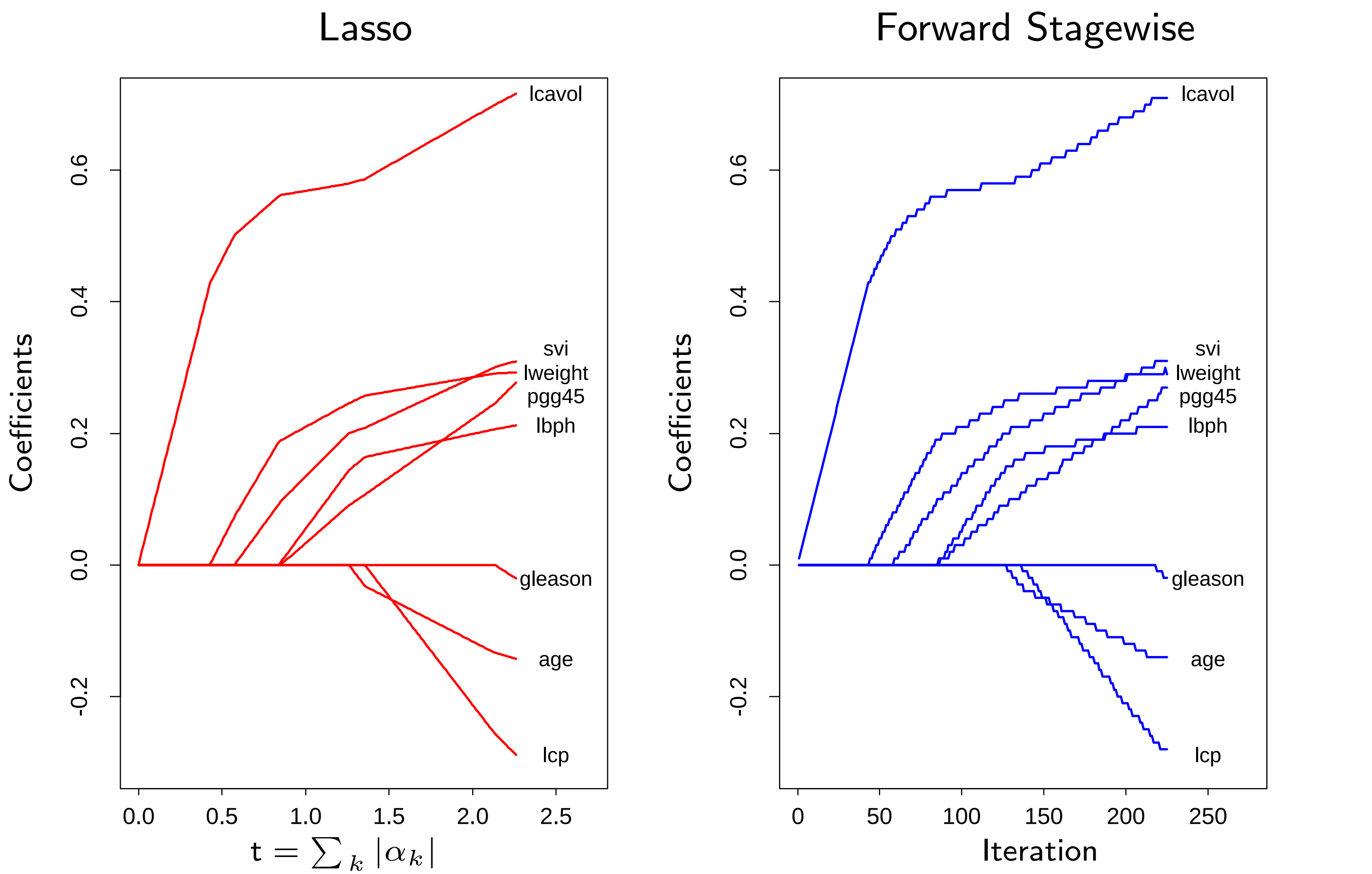
where $L[Y, f(X)] = \exp[-Y f(X)]$.

- As $\lambda \downarrow 0$, $\hat{\beta} \to \beta^*$, the L_1 optimal margin separator.

Details

- ϵ forward stagewise:(idealized boosting with shrinkage). Given a family of basis functions $h_1(x), \ldots h_M(x)$, and loss function L.
- Model at kth step is $F_k(x) = \sum_m \beta_m^k h_m(x)$.
- At step $k+1$, identify coordinate m with largest $|\partial L/\partial \beta_m|$, and update $\beta_m^{k+1} \leftarrow \beta_m^k + \epsilon$.
- Equivalent to the lasso: $\min L(\beta) + \lambda_k ||\beta||_1$
- As $\lambda_k \downarrow 0$, $\beta^k \rightarrow \beta^*$, the L_1 optimal margin separator.

Example and Illustration



Summary

- SVM can be viewed as regularized fitting with a particular loss function: hinge loss.
- Regularized logistic regression gives very similar fit, with added benefits. Also approaches a separating hyperplane. Uses binomial deviance as loss.
- Boosting can be viewed as L_1 regularized fitting (exponential or binomial loss); has optimal margin limiting behavior.