

Exploring the regularization path for adaptive Gaussian kernel SVMs

Roland Memisevic, Nathan Srebro, Sam Roweis

December 9, 2005

Adaptive Gaussian kernel SVM

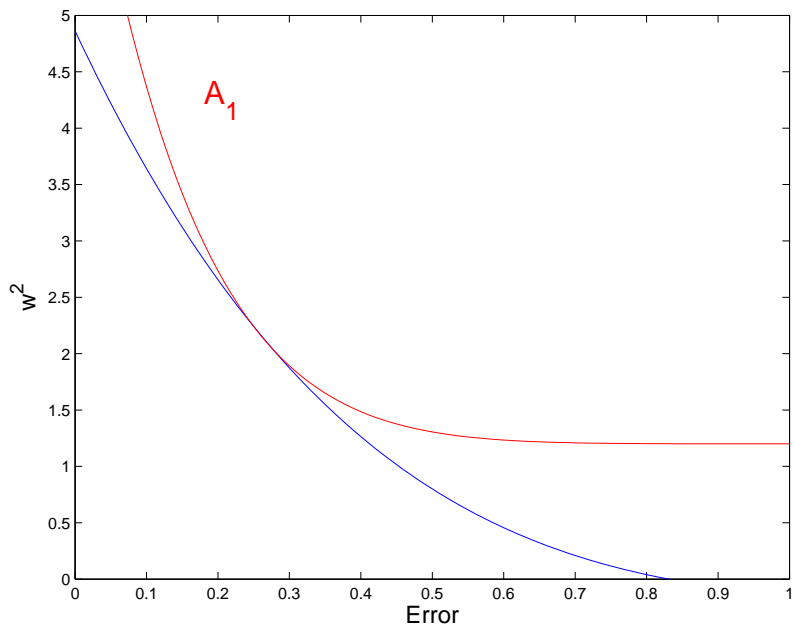
- ▶ **Goal:** Learn the 'shape' of a Gaussian kernel

$$k^A(x^i, x^j) = \exp(-(x^i - x^j)^T A^T A (x^i - x^j))$$

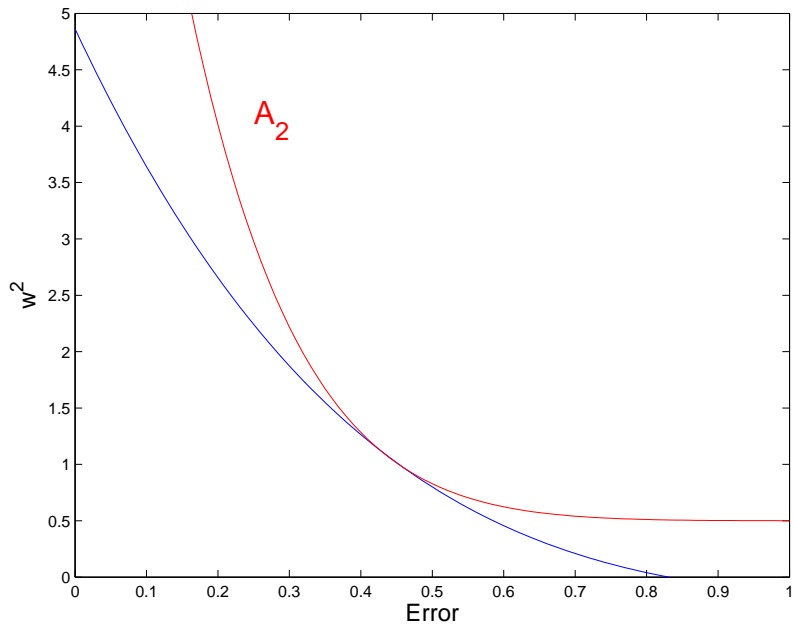
for use in an SVM.

- ▶ Jointly learn the feature transformation A *and* the SVM parameters.
- ▶ Simple feature weighting if A diagonal.
- ▶ (Chapelle, et. al., 2002), (Ong, et. al., 2003)

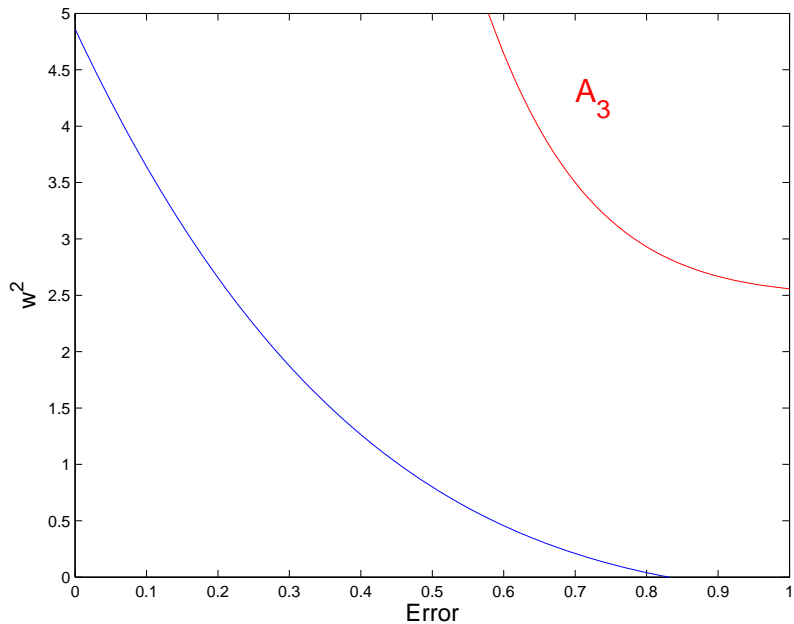
Regularization path



Regularization path



Regularization path



Approach

- ▶ Use a smooth approximation to the hinge loss.
- ▶ → Unconstrained, gradient based optimization in the primal (**not convex**):

$$\min_{\alpha, A} \frac{\lambda}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K_{ij}^A + \sum_i h(y_i (\sum_j \alpha_j y_j \exp(-\|A(x_i - x_j)\|^2) + b))$$

- ▶ To recover the regularization path:
 1. 'Get to' the frontier: Fix λ (large) and optimize.
 2. Iterate:
 - ▶ Decrease lambda.
 - ▶ **Predict** new (α, A) based on the previous iterations.
 - ▶ **Correct** the prediction (Conjugate gradients).

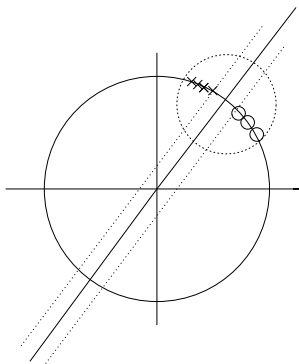
- ▶ Several approaches for the predictor step:
 - ▶ (1) Take the solution from the previous iteration.
 - ▶ (2) Solution from previous iteration + previous step.
 - ▶ (3) Function approximation based on solutions of previous iterations.
 - ▶ (4) Analytical ('implicit function theorem')

Details

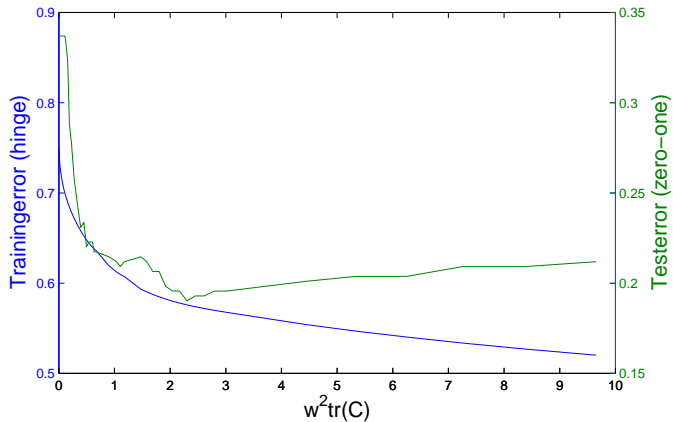
- ▶ Smooth hinge: $h(z) = \frac{1}{\gamma} \log(1 + \exp(\gamma(1 - z)))$
- ▶ Regularizing A: We want to maximize the margin over the 'spread' of points in feature space. One way to achieve this is by replacing:

$$\|w\|^2 \rightarrow \|w\|^2_{\text{tr}(C)},$$

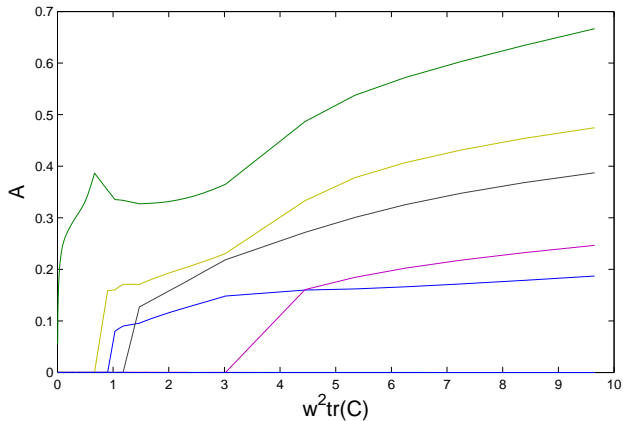
where C is the covariance in feature space ('Power-penalty').



UCI diabetes, train 400 points, test 368 points, 8 dimensions. Training/Test-errors:



UCI diabetes, train 400 points, test 368 points, 8 dimensions. A:



Run times toydata, 100 points

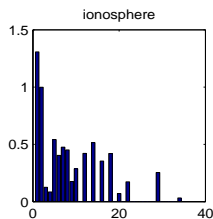
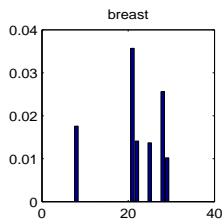
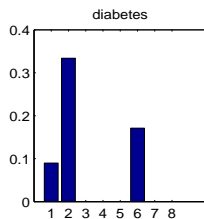
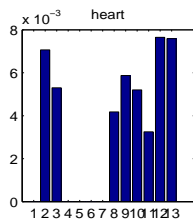
	Average time spent (secs)	Average prediction error
Naive:	11.27	1.30
Simple first order	5.61	2.22
Quadratic in λ	6.95	3.28
Quadratic in $\frac{1}{\lambda}$	8.58	3.02
Analytical (*)	10.60	0.06

Some results

Error rates

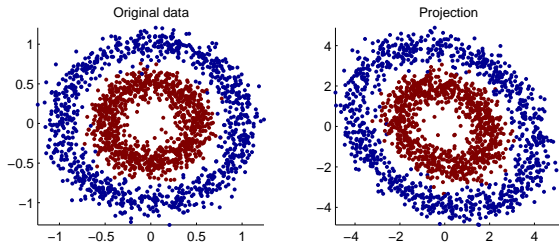
	diagonal	diagonal (power-penalty)	gridsearch
heart (200 pts, dim=13)	0.1949	0.1462	0.1897
diabetes (400 pts, dim=8)	0.2210	0.2054	0.2312
breast (400 pts, dim=30)	0.0529	0.0647	0.0250
ionosphere (250 pts, dim=34)	0.0741	0.0617	0.0815

Optimal transformations (diagonal)



Two rings in 5d

- ▶ We embed 2d-data ('two rings') in 5d and add noise orthogonal to the embedding dimensions.
- ▶ Learn rank-2-covariance.
- ▶ Result:



- ▶ Error rates:

rank 2	diagonal	gridsearch
0.01	0.06	0.0458