SUPPORT VECTOR MACHINE ACTIVE LEARNING

CS 101.2 Caltech,
03 Feb 2009

Paper by S. Tong, D. Koller
Presented by Krzysztof Chalupka

OUTLINE

- SVM intro
  - Geometric interpretation
  - Primal and dual form
  - Convexity, quadratic programming

OUTLINE

- SVM intro
  - Geometric interpretation
  - Primal and dual form
  - Convexity, quadratic programming

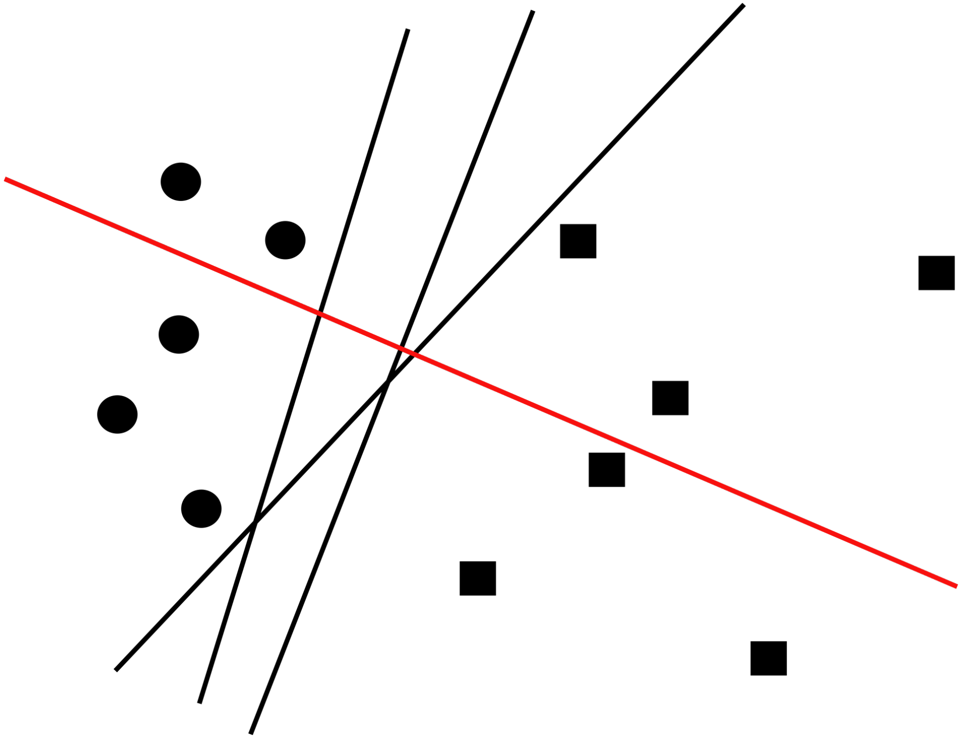
- Active learning in practice
  - Short review
  - The algorithms
  - Implementation

OUTLINE

- SVM intro
  - Geometric interpretation
  - Primal and dual form
  - Convexity, quadratic programming
- Active learning in practice
  - Short review
  - The algorithms
  - Implementation
- Practical results

SVM A SHORT INTRODUCTION

- Binary classification setting:
  - Input data $D_X=\{x_1, \ldots, x_n\}$, labels $\{y_1, \ldots, y_n\}$
  - Consistent hypotheses – Version Space *V*



SVM A SHORT INTRODUCTION

- SVM geometric derivation
  - For now, assume data linearly separable
  - Want to find the separating hyperplane that maximizes the distance between any training point and itself

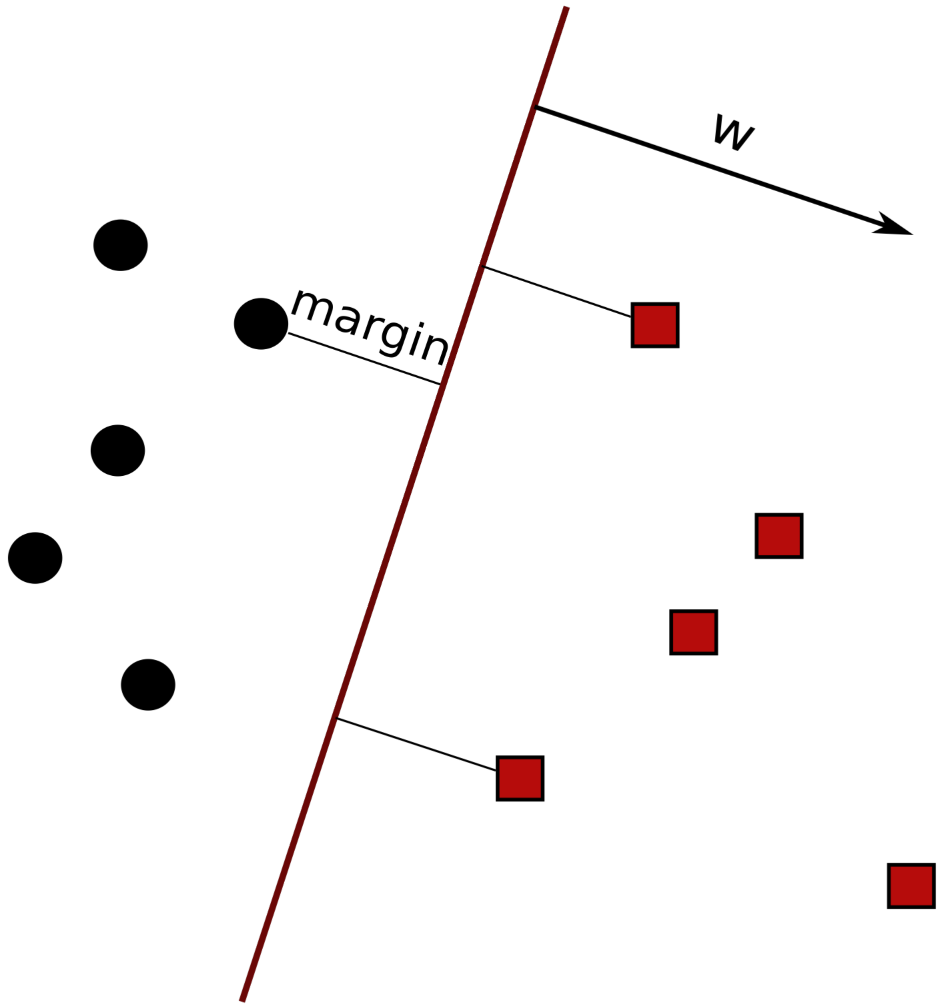
SVM A SHORT INTRODUCTION

- SVM geometric derivation
  - For now, assume data linearly separable
  - Want to find the separating hyperplane that maximizes the distance between any training point and itself
    - Good generalization

SVM A SHORT INTRODUCTION

- SVM geometric derivation
  - For now, assume data linearly separable
  - Want to find the separating hyperplane that maximizes the distance between any training point and itself
    - Good generalization
    - Computationally attractive (later)

SVM A SHORT INTRODUCTION



SVM A SHORT INTRODUCTION

- Primal form

$$minimize_{w,b} \ \frac{1}{2}||w||^2$$

$$subj \ to \ \forall_i \ y_i(w.x_i + b) \geq 1$$

SVM A SHORT INTRODUCTION

- Primal form

$$minimize_{w,b}\ \frac{1}{2}||w||^2$$

$$subj\ to\ \forall_i\ y_i(w.x_i + b) \geq 1$$

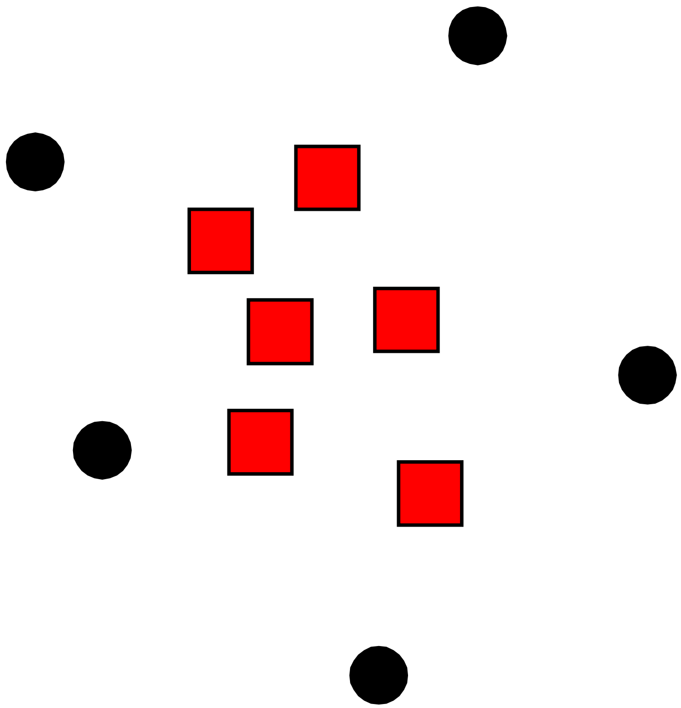
- Dual form (Lagrangian multipliers)

$$minimize_{\lambda}\ \Sigma_{i=1}^{m}\lambda_i - \frac{1}{2}\Sigma_{i,j=1}^{m}\lambda_i\lambda_j y_i y_j (x_i.x_j)$$

$$subj\ to\ \forall_i\ \lambda_i \geq 0\ and\ \Sigma_{i=1}^{m}\lambda_i y_i = 0$$

SVM A SHORT INTRODUCTION

- Problem: classes not linearly separable

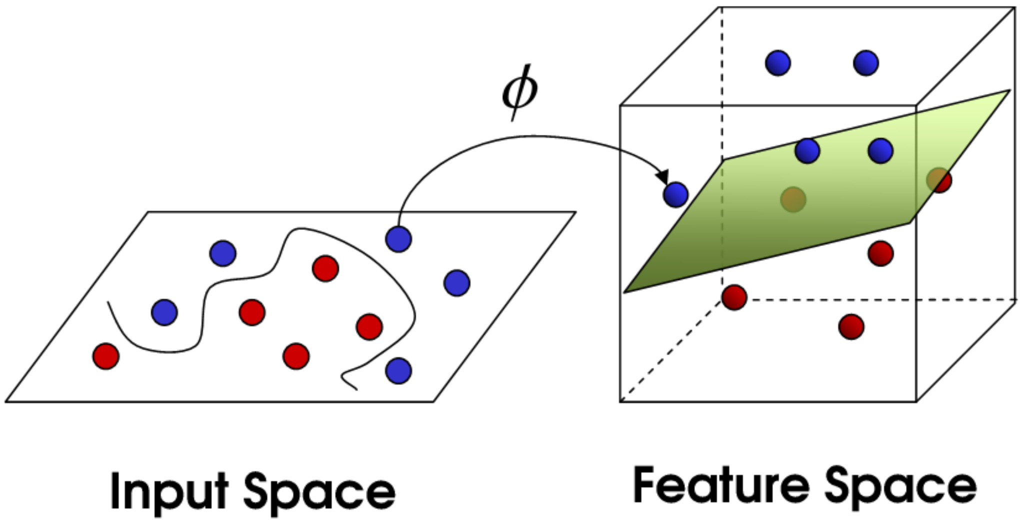


- Solution: get more dimensions

SVM A SHORT INTRODUCTION

- Get more dimensions
  - Project the inputs to a feature space

$$f(x) = sgn(\Sigma_{i=1}^{m} y_i \lambda_i (\Phi(x).\Phi(x_i)) + b)$$



SVM A SHORT INTRODUCTION

- The Kernel Trick: use a (positive definite) kernel as the dot product

$$f(x) = sgn(\Sigma_{i=1}^{m} y_i \lambda_i k(x, x_i) + b)$$

  - OK, as the input vectors only appear in the dot product
  - Again (as in Gaussian Process Optimization) some conditions on the kernel function must be met

SVM A SHORT INTRODUCTION

- Polynomial kernel

$$k(x, x') = (x.x')^d$$

- Gaussian kernel

$$k(x, x') = exp(-\frac{||x - x'||^2}{2\sigma^2})$$

- Neural Net kernel (pretty cool!)

$$k(x, x') = tanh(\kappa(x.x') + \Theta)$$

ACTIVE LEARNING

- Recap
  - Want to query as little points as possible and find the separating hyperplane

ACTIVE LEARNING

- Recap
  - Want to query as little points as possible and find the separating hyperplane
  - Query the most uncertain points first

ACTIVE LEARNING

- Recap
  - Want to query as little points as possible and find the separating hyperplane
  - Query the most uncertain points first
  - Request labels until only one hypothesis left in the version space

ACTIVE LEARNING

- Recap
  - Want to query as little points as possible and find the separating hyperplane
  - Query the most uncertain points first
  - Request labels until only one hypothesis left in the version space
  - One idea was to use a form of binary search to shrink the version space; that’s what we’ll do

ACTIVE LEARNING

- Back to SVMs
  - maximize

    $$sgn(\Sigma_{i=1}^{m} y_i \lambda_i k(x, x') + b)$$

    subj to

    $$\lambda_i \geq 0, \ \Sigma_{i=1}^{m} \lambda_i y + i = 0$$

  - Area(*V*) – the surface that the version space occupies on the hypersphere |**w**| = 1 (assume b = 0)

    (we use the duality between feature and version space)

ACTIVE LEARNING

- Back to SVMs
  - Area(V) – the surface that the version space occupies on the hypersphere $|\mathbf{w}| = 1$ (assume b = 0)

    (we use the duality between feature and version space)
  - Ideally, want to always query instances that would halve Area(V)
  - V^+,V^- - the version spaces resulting from querying a particular point and getting a + or – classification
  - Want to query points with Area(V^+) = Area(V^-)

ACTIVE LEARNING

- Bad Idea
  - Compute Area(V-) and Area(V+) for each point explicitly

ACTIVE LEARNING

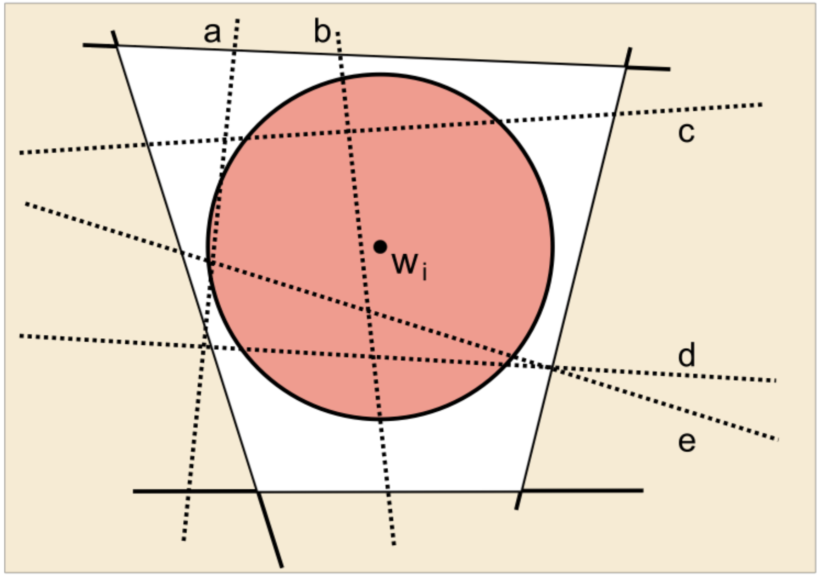
- Bad Idea
  - Compute Area(V-) and Area(V+) for each point explicitly
- A better one
  - Estimate the resulting areas using simpler calculations

ACTIVE LEARNING

- Bad Idea
  - Compute Area(V-) and Area(V+) for each point explicitly
- A better one
  - Estimate the resulting areas using simpler calculations
- Even better
  - Reuse values we already have

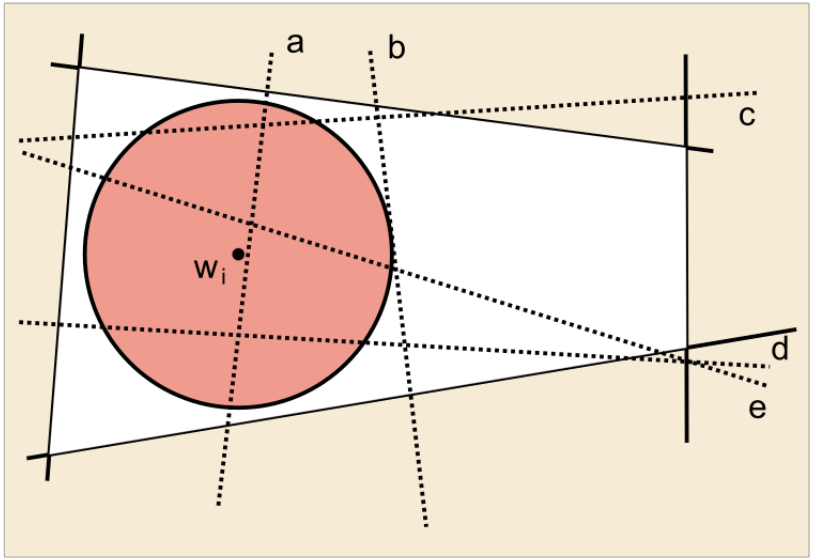
ACTIVE LEARNING

- Simple Margin
  - Each data point has a corresponding hyperplane
  - How close this hyperplane is to $\mathbf{w}_i$ will tell us how much it bisects the current version space
  - Choose **x** closest to **w**



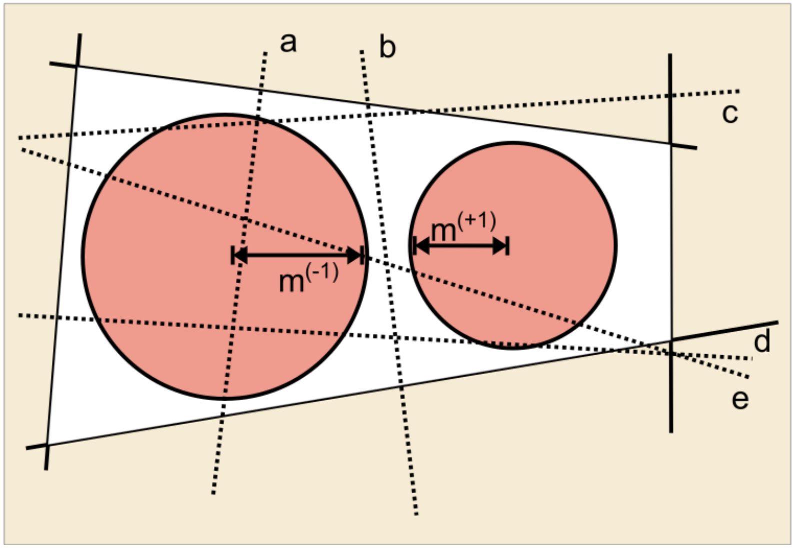
ACTIVE LEARNING

- Simple Margin
  - If V_i is highly non-symmetric and/or $\mathbf{w}_i$ is not centrally placed the result might be ugly



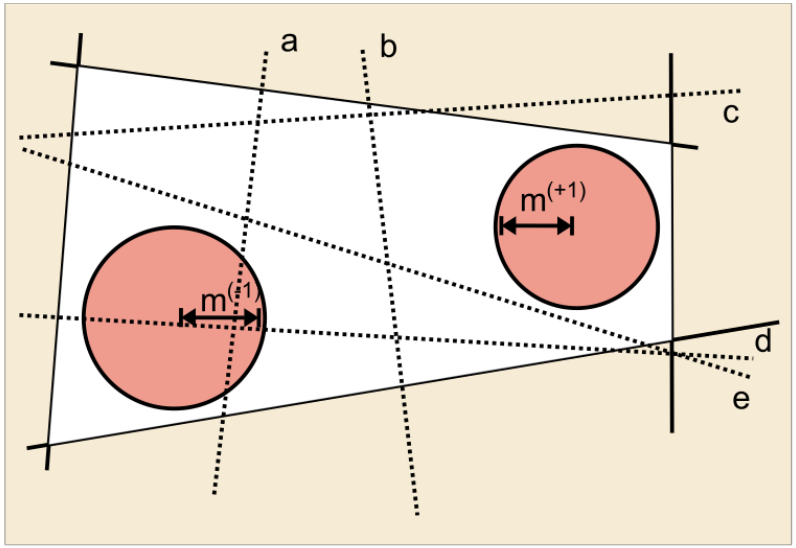
ACTIVE LEARNING

- MaxMin Margin
  - Use the fact that an SVMs margin is proportional to the resulting version space's area
  - The algorithm: for each unlabeled point compute the two margins of the potential version spaces *V*$^+$ and *V*$^-$. Request the label for the point with the largest min(m$^+$, m$^-$)



ACTIVE LEARNING

- MaxMin Margin
  - A better approximation of the resulting split
  - Both MaxMin and Ratio (coming next) computationally more intensive than Simple
  - But can still do slightly better, still without explicitly computing the areas



ACTIVE LEARNING

- Ratio Margin
  - Similar to MaxMin, but considers the fact that the shape of the version space might make the margins small even if they are a good choice
  - Choose the point with the largest resulting

$$min(\frac{m^-}{m^+}, \frac{m^+}{m^-})$$

  - Seems to be a good choice

ACTIVE LEARNING

- Implementation
  - Once we have computed the SVM to get $V^{+/-}$, we can use the distance of any support vector x from the hyperplane

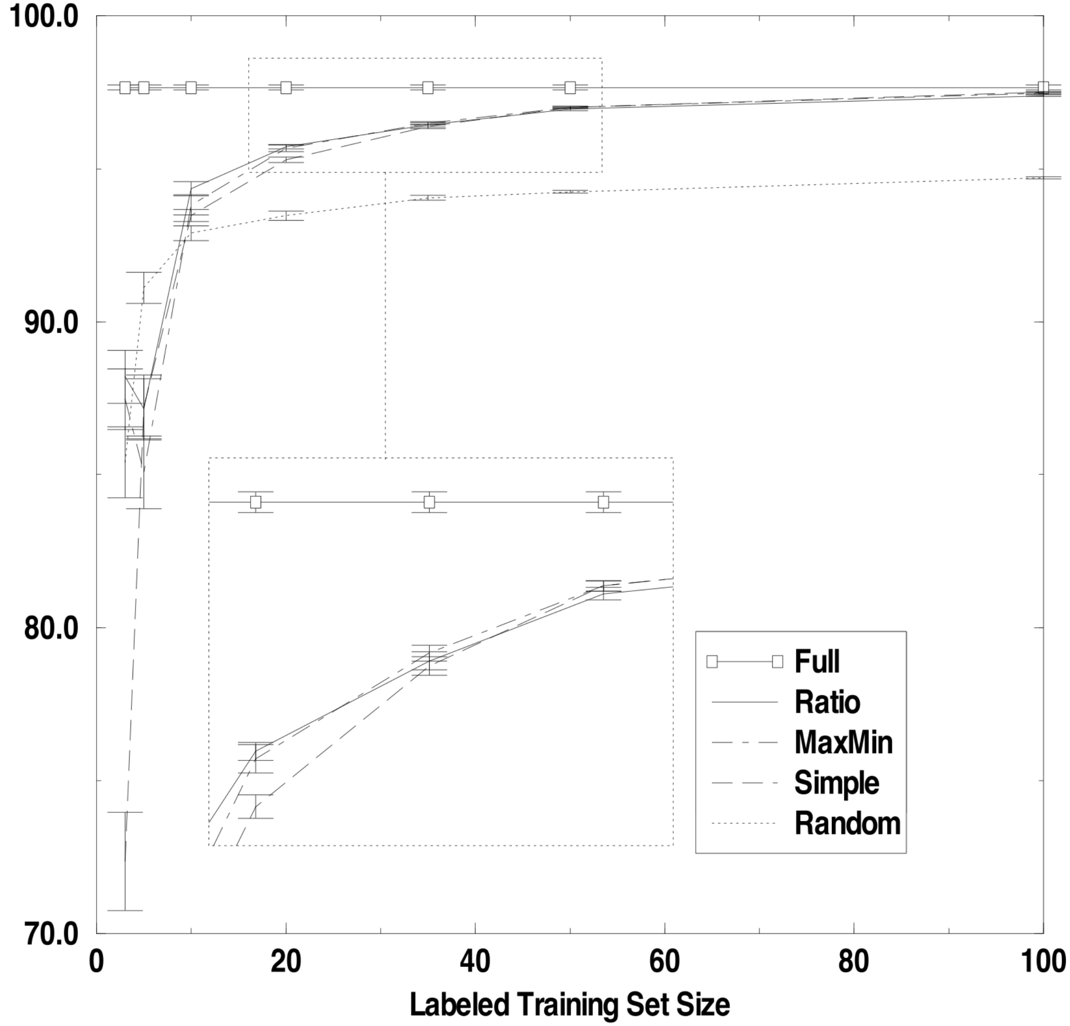
$$||\Sigma y_i \lambda_i k(x, x_i) + b||$$

    to get the margins
  - Good, as many lambdas are 0s

PRACTICAL RESULTS

- Article text Classification
  - Reuters Data Set, around 13000 articles
  - Multi-class classification of articles by topics
  - Around 10000 dimensions (word vectors)
  - Sample 1000 unlabelled examples, randomly choose two for a start
  - Polynomial kernel classification
  - Active Learning: Simple, MaxMin & Ratio
  - Articles transformed to vectors of word frequencies (“bag of words”)

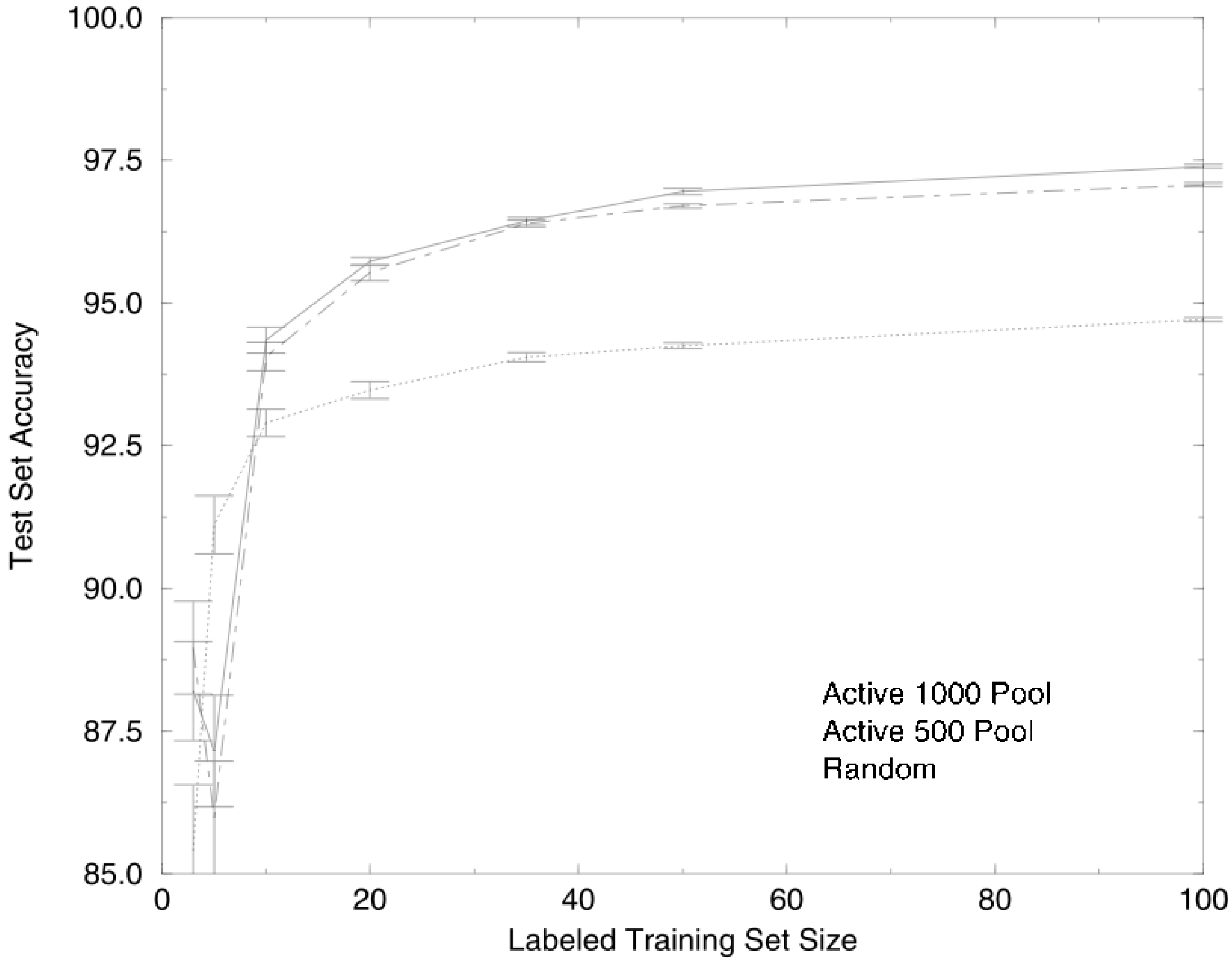
PRACTICAL RESULTS



PRACTICAL RESULTS

	Simple	MaxMin	Ratio	Equivalent Random size
Earn	86.39 ± 1.65	87.75 ± 1.40	90.24 ± 2.31	34
Acq	77.04 ± 1.17	77.08 ± 2.00	80.42 ± 1.50	> 100
Money-fx	93.82 ± 0.35	94.80 ± 0.14	94.83 ± 0.13	50
Grain	95.53 ± 0.09	95.29 ± 0.38	95.55 ± 1.22	13
Crude	95.26 ± 0.38	95.26 ± 0.15	95.35 ± 0.21	> 100
Trade	96.31 ± 0.28	96.64 ± 0.10	96.60 ± 0.15	> 100
Interest	96.15 ± 0.21	96.55 ± 0.09	96.43 ± 0.09	> 100
Ship	97.75 ± 0.11	97.81 ± 0.09	97.66 ± 0.12	> 100
Wheat	98.10 ± 0.24	98.48 ± 0.09	98.13 ± 0.20	> 100
Corn	98.31 ± 0.19	98.56 ± 0.05	98.30 ± 0.19	

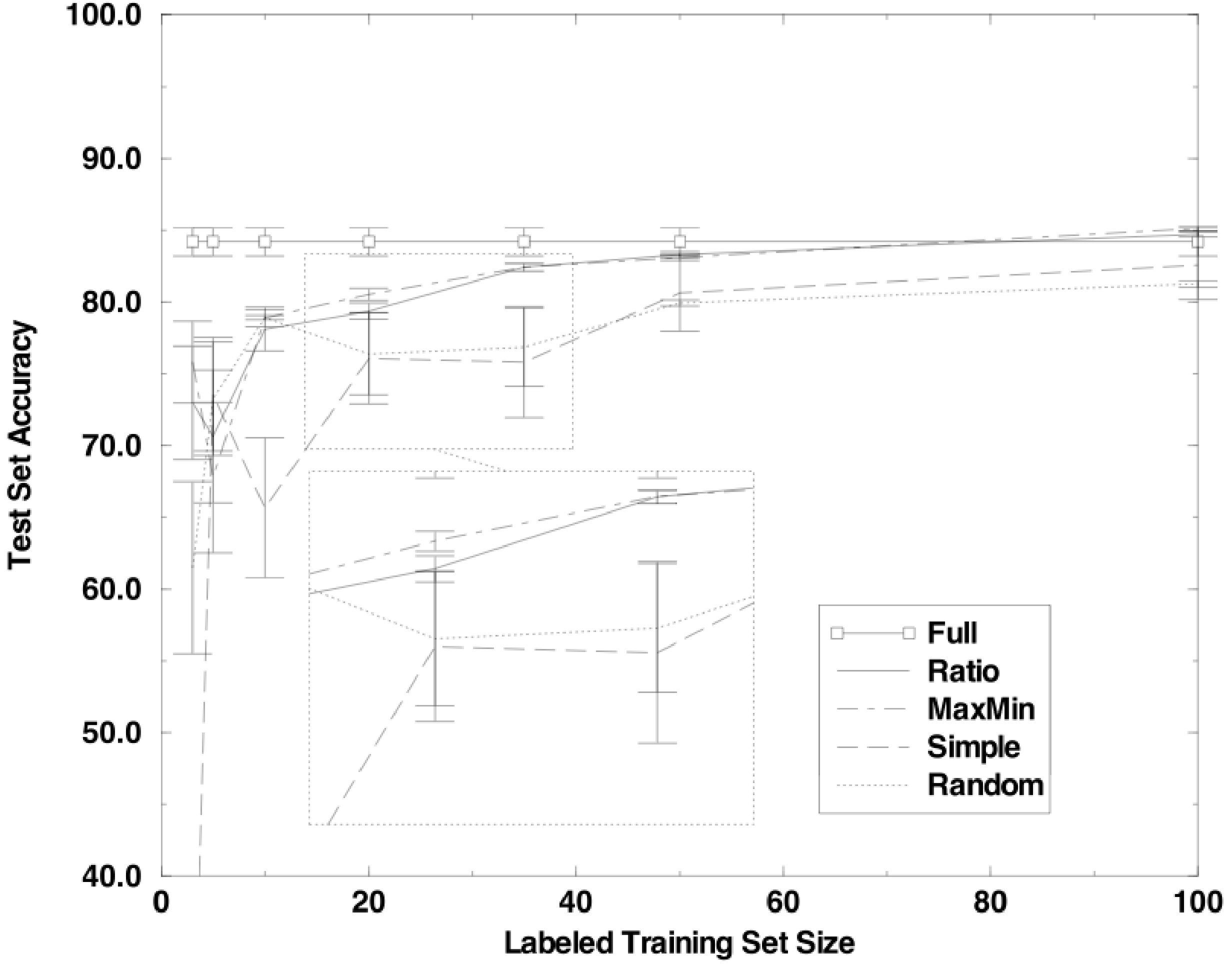
PRACTICAL RESULTS



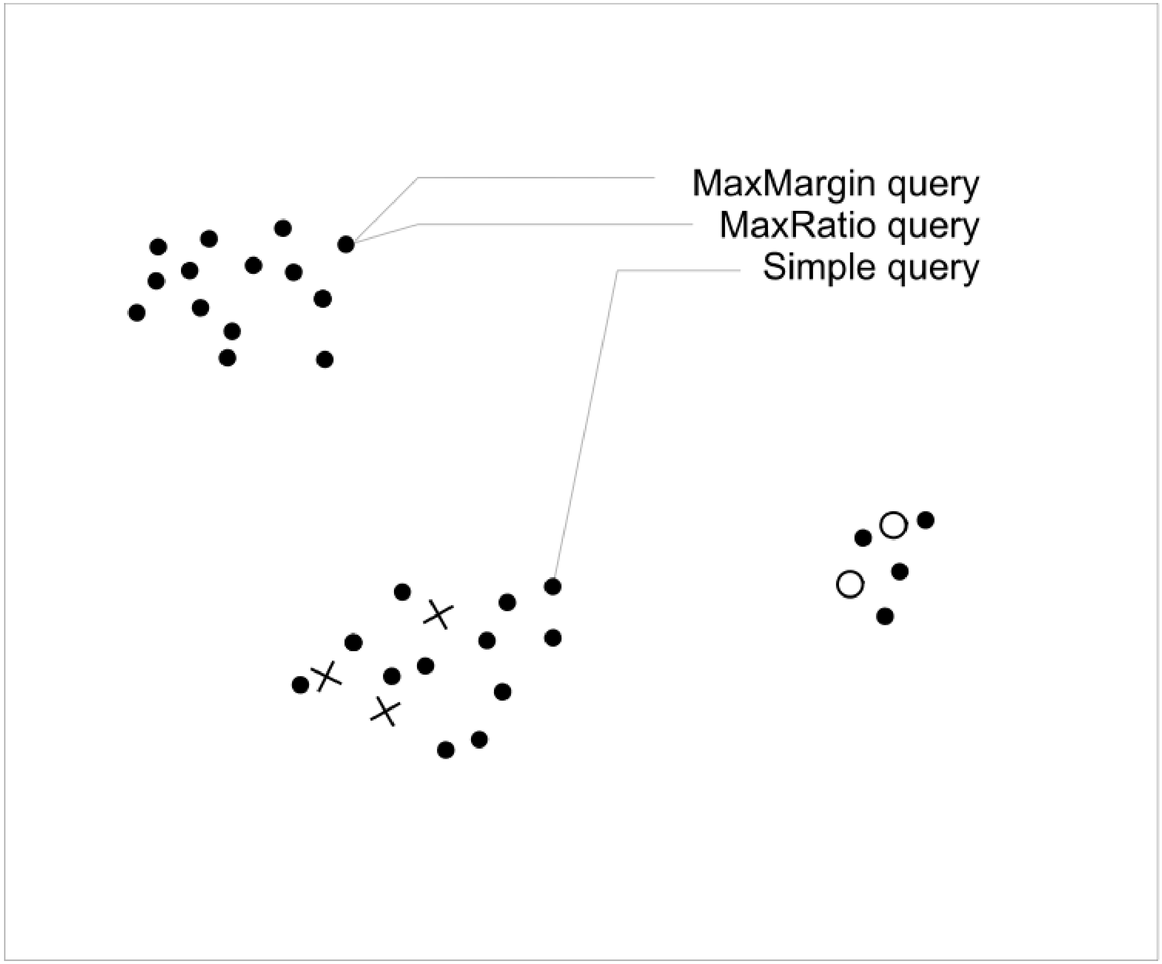
PRACTICAL RESULTS

- Usenet text classification
  - Five comp.* groups, 5000 documents, 10000 dimensions
  - 2500 randomly selected for testing, 500 of the remaining for active learning
  - Generally similar results; Simple turns out unstable

PRACTICAL RESULTS



PRACTICAL RESULTS



THE END

- SVMs for pattern classification
- Active Learning
  - Simple Margin
  - MinMax Margin
  - Ratio Margin
- All better than passive learning, but MinMax and Ratio can be computationally intensive
- Good results in text classification (also in handwriting recognition etc)