
On generalization bounds, projection profile, and margin distribution

Ashutosh Garg

Sariel Har-Peled

Dan Roth

University of Illinois, Urbana-Champaign

danr@cs.uiuc.edu

<http://L2R.cs.uiuc.edu/~danr>

Learning with high dimensional data

- ◇ Identifying phrase structure

[_{NP} He] [_{VP} reckons] [_{NP} the current account deficit] [_{VP} will narrow]
[_{PP} to] [_{NP} only # 1.8 billion] [_{PP} in] [_{NP} September]

- ◇ Information Extraction Tasks

afternoon, Dr. Ab C will talk in Ms. De. F class..

- ◇ Prepositional Phrase Attachment

buy **shirt with sleeves**, buy shirt with a credit card

- ◇ Context Sensitive Spelling Correction

Illinois' **bored** of education

board

Learning with high dimensional data

[_{NP} He] [_{VP} reckons] [_{NP} the current account deficit] [_{VP} will narrow]
[_{PP} to] [_{NP} only # 1.8 billion] [_{PP} in] [_{NP} September]

afternoon, Dr. Ab C will talk in Ms. De. F class..

buy shirt with sleeves, buy shirt with a credit card

Illinois' bored of education

board

Features include: (patterns of) words; POS tags; relational information (location; order; structure...)

In many of these problems dimensionality is 10^5 or more

Easiness of Learning

We learn well from relatively small number of examples in very high dimensional spaces? *Should we believe it?*

Some high dimensional problems are naturally constrained and become, *effectively, low dimensional problems.*

[Roth, Zelenko'00; Garg, Roth'01, Vempala'00]

In these cases, although learning is done in high dimension, *generalization ought to depend on the true, lower dimensionality of the problem.*

Not exploited by current theories

This work

Introduces a way to **analyze learning in high dimension** in a way that exploits the lower, effective dimensionality of the data.

Random projection methods are used to explicitly exploit the margin distribution

Exhibits generalization bounds that are (sometimes) realistic (**< 0.5**) for real problems in NLP and vision

Standard Bounds

VC dimension based bounds (hyperplanes)

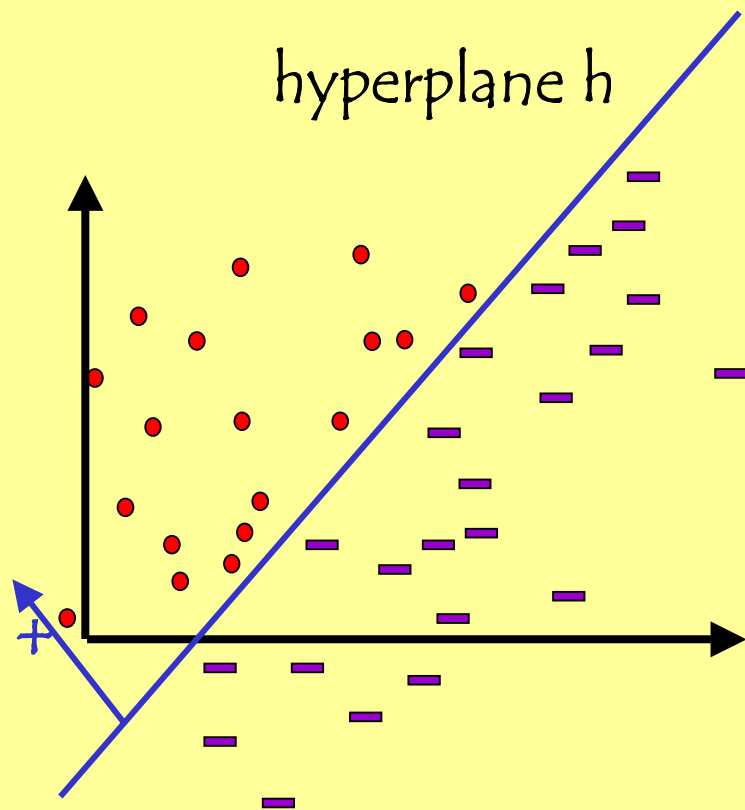
VC(n,m)

$$ERR_D \leq ERR_S + \sqrt{\frac{[n(\ln(2m/n) + 1) - \ln(\delta/4)]}{m}}$$

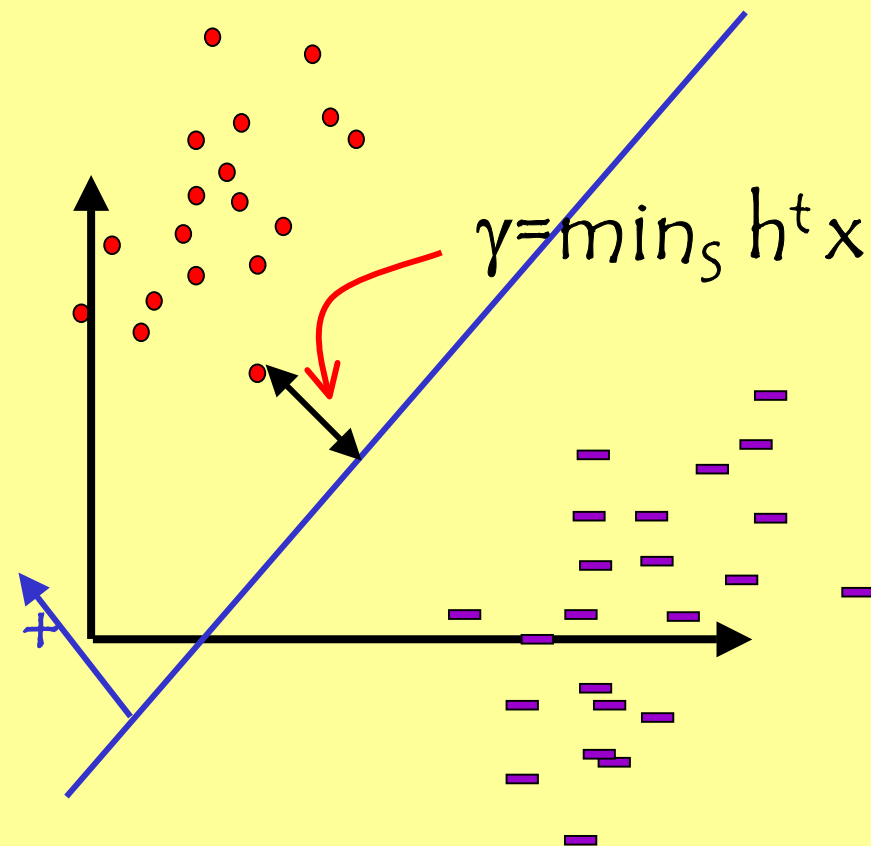
Margin Based bounds (data dependent; γ - margin)

$$ERR_D \leq ERR_S + (2/m) \left(\frac{1}{\gamma^2} \log(32m) \log(8em\gamma^2) + \log(8m/\delta) \right)$$

Intuition



Hard Problem



Easy Problem

Standard Bounds

VC dimension based bounds (hyperplanes)

VC(n,m)

$$ERR_D \leq ERR_S + \sqrt{\frac{n(\ln(2m/n) + 1) - \ln(\delta/4)}{m}}$$

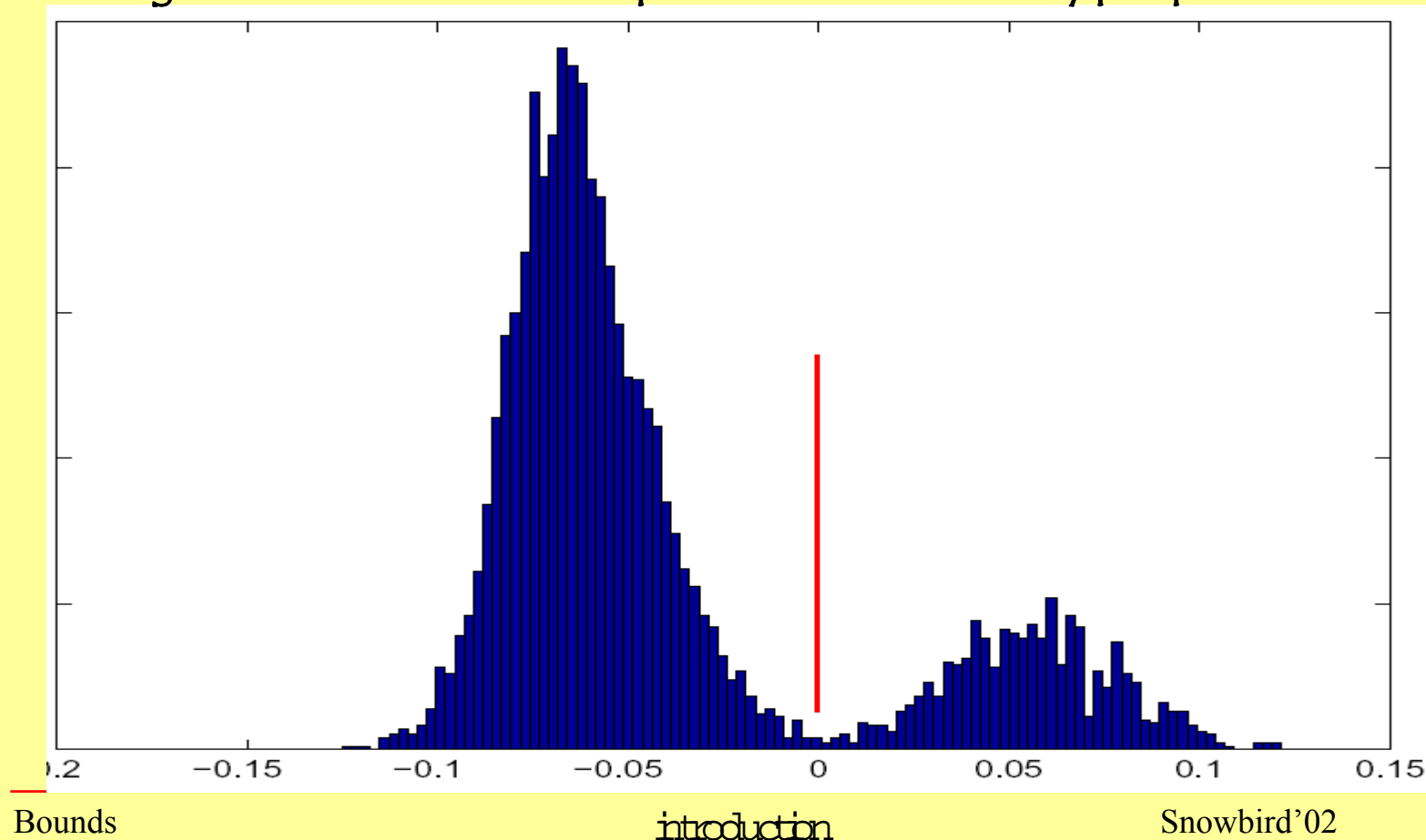
Margin Based bounds (data dependent; γ - margin)

$$ERR_D \leq ERR_S + (2/m) \left(\frac{1}{\gamma^2} \log(32m) \log(8em\gamma^2) + \log(8m/\delta) \right)$$

Typically: $1 \ll VC \text{ bounds} \ll \text{Margin Based bound}$

Real Data

17,000 dimensional context sensitive spelling
Histogram of distance of points from the hyperplane



Standard Bounds

VC dimension based bounds (hyperplanes)

VC(n,m)

$$ERR_D \leq ERR_S + \sqrt{\frac{n(\ln(2m/n) + 1) - \ln(\delta/4)}{m}}$$

Margin Based bounds (data dependent; γ - margin)

$$ERR_D \leq ERR_S + (2/m) \left(\frac{1}{\gamma^2} \log(32m) \log(8em\gamma^2) + \log(8m/\delta) \right)$$

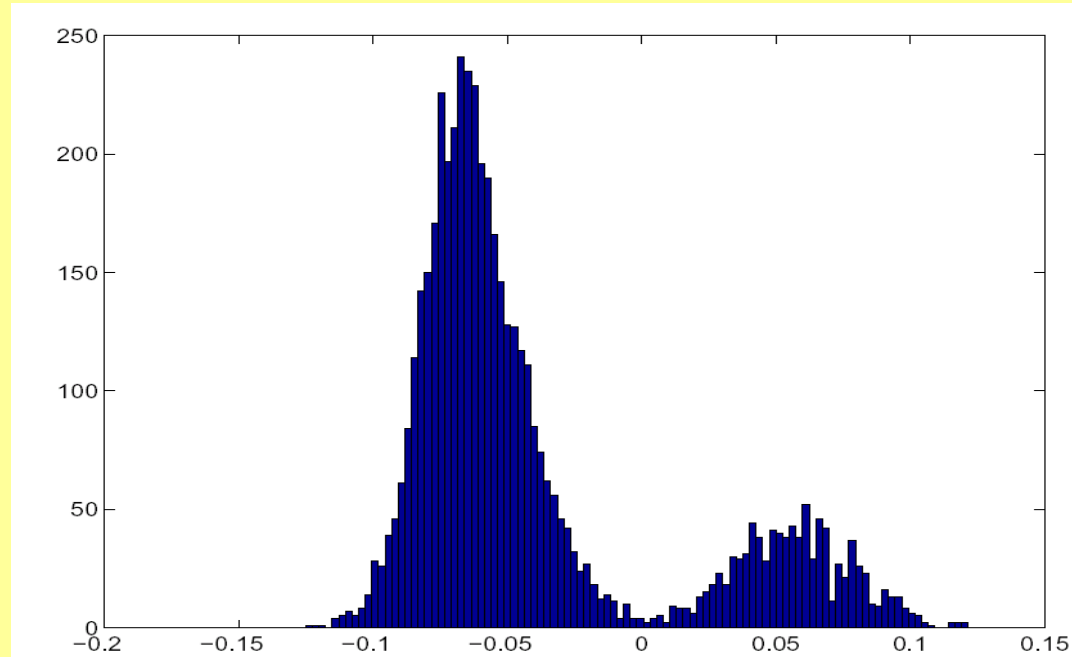
Typically: $1 \ll VC \text{ bounds} \ll \text{Margin Based bound}$

Value of bounds: algorithmic insight; model selection

This work

Even for:
17,000 dimensional
context sensitive spelling

Can get bounds
that are < 0.5 ,
using a 1000-5000 examples.



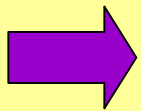
Key Idea: Projection Profile (I)

Learn a Hyperplane h from sample S , in high dimension n

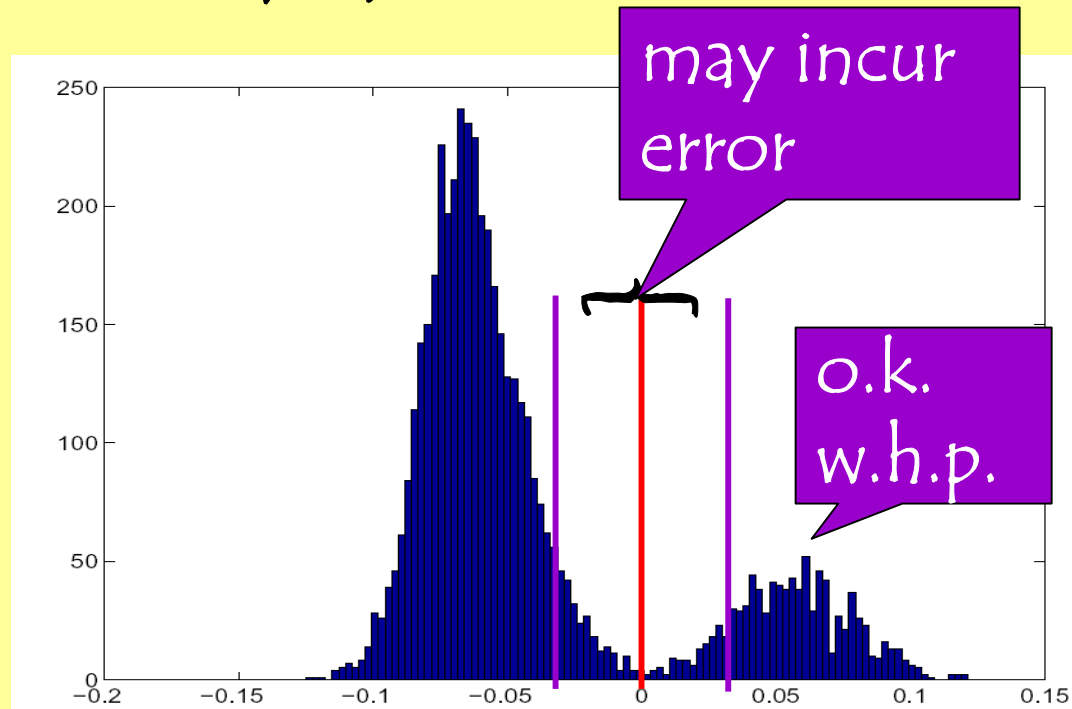
Analysis: Project S and h randomly to low dimension (k)

w.h.p (k, S): small distortion of distances.

(Johnson-Lindenstrauss)



Small error in the lower dimension



Key Idea: Projection Profile (II)

Expected amount of error introduced in projection captured by: $a_k(\mathbf{D}, \mathbf{h}) = \int_{\mathbf{x} \in \mathbf{D}} u(\mathbf{x}) d\mathbf{D}$

where: $u_k(\mathbf{x}) = \min \left\{ \exp \left(-\frac{k}{|\mathbf{l}(\mathbf{x})|} \right), \frac{1}{k|\mathbf{l}^2(\mathbf{x})}, 1 \right\}$ $\mathbf{l}(\mathbf{x}) = \mathbf{h}^t \mathbf{x}$
the profile:

$$P(\mathbf{D}, \mathbf{h}) = (a_1(\mathbf{D}, \mathbf{h}), a_2(\mathbf{D}, \mathbf{h}), \dots, a_k(\mathbf{D}, \mathbf{h}), \dots)$$

Key Idea: Projection Profile (II)

Expected amount of error introduced in projection captured by: $a_k(\mathbf{D}, \mathbf{h}) = \int_{\mathbf{x} \in \mathbf{D}} u(\mathbf{x}) d\mathbf{D}$

where: $u_k(\mathbf{x}) = \min \left\{ \exp \left(-\frac{k}{|\mathbf{l}(\mathbf{x})|} \right), \frac{1}{k|\mathbf{l}^2(\mathbf{x})}, 1 \right\}$ $\mathbf{l}(\mathbf{x}) = \mathbf{h}^t \mathbf{x}$
the profile:

$$P(\mathbf{D}, \mathbf{h}) = (a_1(\mathbf{D}, \mathbf{h}), a_2(\mathbf{D}, \mathbf{h}), \dots, a_k(\mathbf{D}, \mathbf{h}), \dots)$$

gives the tradeoff between dimensionality and accuracy
Resulting bound:

$$ERR_D \leq ERR_S + \min_k \{ \hat{u}_k + VC(k, m) \}$$

Rest of the talk

- ◇ Some details
 - Random projection
 - Random projection for classification
 - Projection profile of a sample
- ◇ Analysis
- ◇ Future/Questions

Random Projection

Random Matrix: $R[k \times n]$ with $r_{ij} \sim N(0, 1/k)$
 $x \in \mathfrak{R}^n$, $x' = Rx \in \mathfrak{R}^k$

Theorem [Johnson Lindenstrauss 84]:

$u, v \in \mathfrak{R}^n$; $[u', v'] = R[u, v]$, projections to \mathfrak{R}^k . For any c

$$\Pr \left[(1 - c) \leq \frac{\|u' - v'\|^2}{\|u - v\|^2} \leq (1 + c) \right] \geq 1 - e^{-c^2 k / 8}$$

where the probability is over the selection of the random matrix R .

Plan

- ◇ Project a sample and the hyperplane
- ◇ Bound empirical error in the projected space (k)

Random Projection: A Classification Version

Lemma:

\mathbf{h} : n -dimensional classifier, $\mathbf{x} \in \mathbb{R}^n$; $\|\mathbf{h}\| = \|\mathbf{x}\| = 1$, $l(\mathbf{x}) = \mathbf{h}^T \mathbf{x}$

The probability of misclassifying \mathbf{x} due to the random projection R , is

$$P[\text{sgn}(\mathbf{h}^T \mathbf{x}) \neq \text{sgn}(\mathbf{h}'^T \mathbf{x}')] \leq \min \left\{ \exp \left(-\frac{l^2(\mathbf{x})k}{8(2 + |l(\mathbf{x})|)^2} \right), \frac{1}{kl^2(\mathbf{x})}, 1 \right\}$$

Intuition: (A Classification Version of RP)

$$P[\text{sgn}(\mathbf{h}^T \mathbf{x}) \neq \text{sgn}(\mathbf{h}'^T \mathbf{x}')] \leq \exp\left(-\frac{l^2(\mathbf{x})k}{|8(2+|l(\mathbf{x})|)^2}\right)$$

Since $\|\mathbf{h}\| = \|\mathbf{x}\| = 1$, $l = \mathbf{h}^T \mathbf{x}$ $l' = \mathbf{h}'^T \mathbf{x}'$

we have $\|\mathbf{h} - \mathbf{x}\|^2 = \|\mathbf{h}\|^2 + \|\mathbf{x}\|^2 - 2\mathbf{h}^T \mathbf{x} = 2 - 2l$

$$\|\mathbf{h}' - \mathbf{x}'\|^2 = \|\mathbf{h}'\|^2 + \|\mathbf{x}'\|^2 - 2l'$$

JL: With probability at least $1 - \exp(-c^2 k/8)$

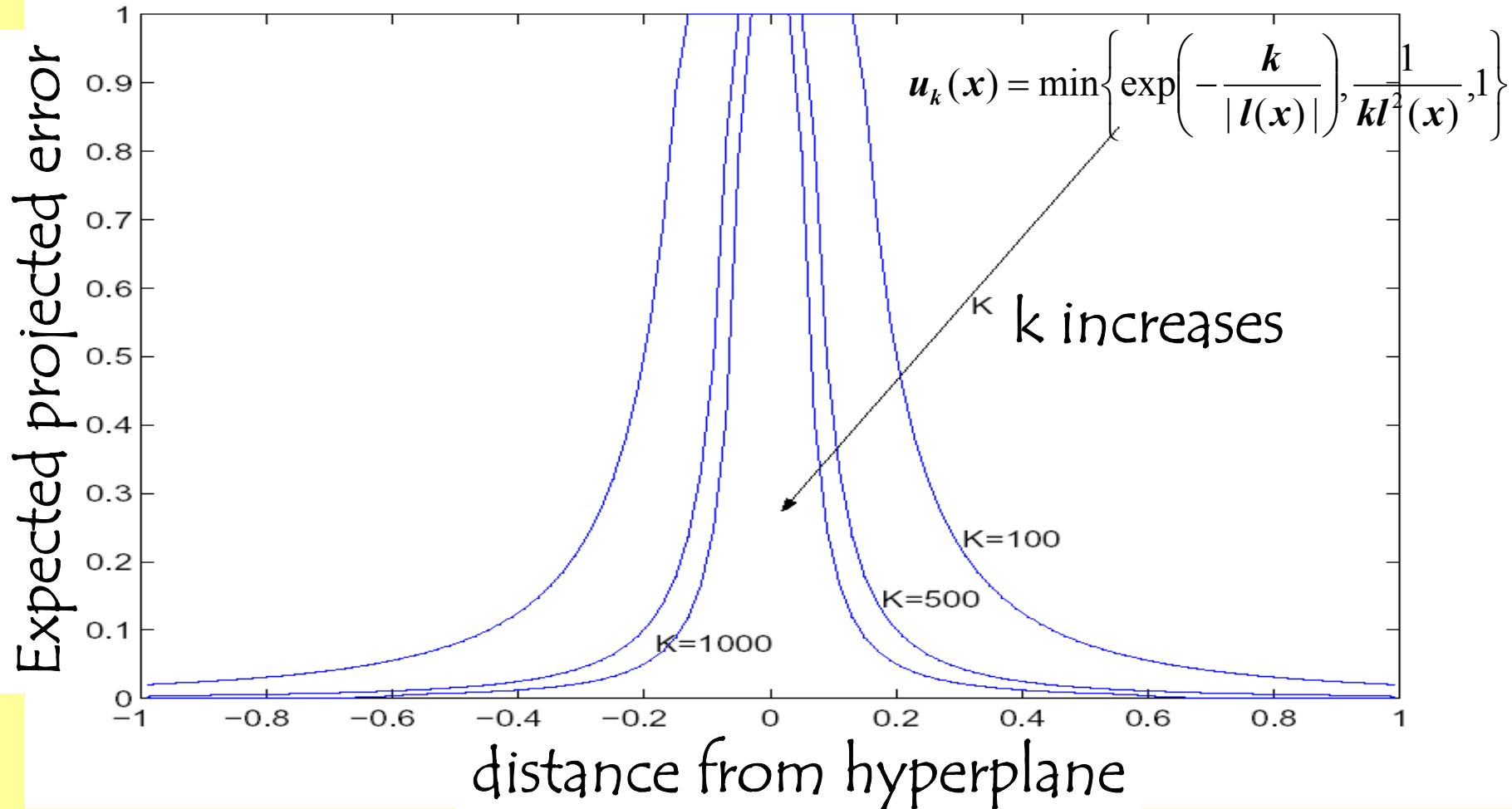
$$(1-c) \|\mathbf{h}\|^2 \leq \|\mathbf{h}'\|^2 \leq (1+c) \|\mathbf{h}\|^2,$$

$$(1-c) \|\mathbf{x}\|^2 \leq \|\mathbf{x}'\|^2 \leq (1+c) \|\mathbf{x}\|^2$$

$$(1-c) \|\mathbf{h} - \mathbf{x}\|^2 \leq \|\mathbf{h}' - \mathbf{x}'\|^2 \leq (1+c) \|\mathbf{h} - \mathbf{x}\|^2.$$

Can find c in JL so that l and l' have same sign.

Contribution of points to error



Projection Error for a Sample (I)

Definition (projection error):

Given a classifier h , a sample S , and a random matrix R , the classification error caused by R is defined by:

$$Err_{proj}(h, R, S) = \frac{1}{|S|} \sum_{x \in S} I(\text{sign}(h^T x) \neq \text{sign}(h'^T x')).$$

Lemma: With probability $> 1 - \delta$ (over the choice of R)

The projection error for sample S , $|S|=m$ is bounded by:

$$Err_{proj}(h, R, S) \leq \frac{1}{m \delta} \sum_1^m 3 \exp\left(-\frac{l^2(x)k}{|8(2+|l(x)|)|^2}\right)$$

Proof idea

- ◇ Bound the expectation of the projection error with respect to the choice of the random matrix

$$E[Err_{proj}(\mathbf{h}, \mathbf{R}, \mathbf{S})]$$

- ◇ Use Markov inequality

Projection Error for a Sample (II)

Can now establish: The difference between the classification performance on two samples in high dimension is similar to difference in low dimension

Lemma:

S_1, S_2 be two samples in \mathfrak{R}^n , $|S_1|=|S_2|=m$;

S'_1, S'_2 the projected sets. Then, with probability $>1-2\delta$

$$P[| \mathbf{Err}(h, S_1) - \mathbf{Err}(h, S_2) | > \varepsilon] < P[| \mathbf{Err}(h', S'_1) - \mathbf{Err}(h', S'_2) | > \rho]$$

Where $\rho = \varepsilon - \mathbf{Err}(h, S_1) - \mathbf{Err}(h, S_2)$

Final Bound

Using Vapnik's doubling trick –

- once on the n dimensional data and
- once on the projected data, can now bound

$$\Pr[\sup_{h \in H} | \bar{Err}(h) - Err(h, S_1) | > \varepsilon]$$

To yield the final bound.

Empirical error

Random Proj.
error

VC at
dimension k

$$ERR_D \leq ERR_S + \min_k \{ \hat{u}_k + VC(k, m) \}$$

Analysis

- ◇ The expected probability of error for a k -dimensional image of x of distance $l(x) =$ from an n -dimensional hyperplane:

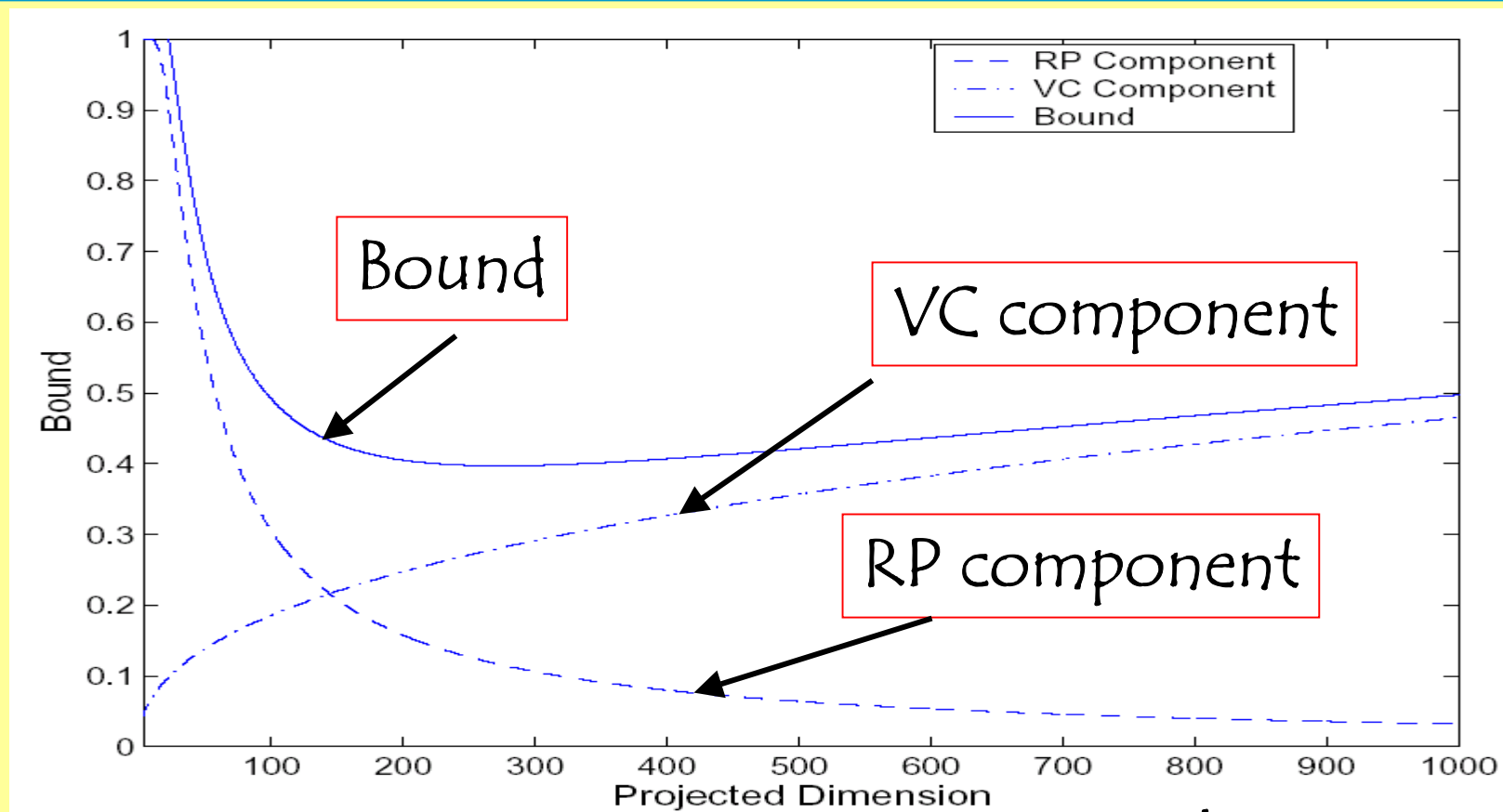
$$\min \left\{ \exp \left(-\frac{l^2(\mathbf{x})k}{8(2+|l(\mathbf{x})|)^2} \right), \frac{1}{kl^2(\mathbf{x})}, 1 \right\}$$

- ◇ Given a probability distribution over the instance space, can compute the distribution over the margin

$$\int_{x \in D} \min \left\{ \exp \left(-\frac{l^2(\mathbf{x})k}{8(2+|l(\mathbf{x})|)^2} \right), \frac{1}{kl^2(\mathbf{x})}, 1 \right\}$$

- ◇ E.g., if $l \sim N(0.3, 0.1)$ can compute this analytically

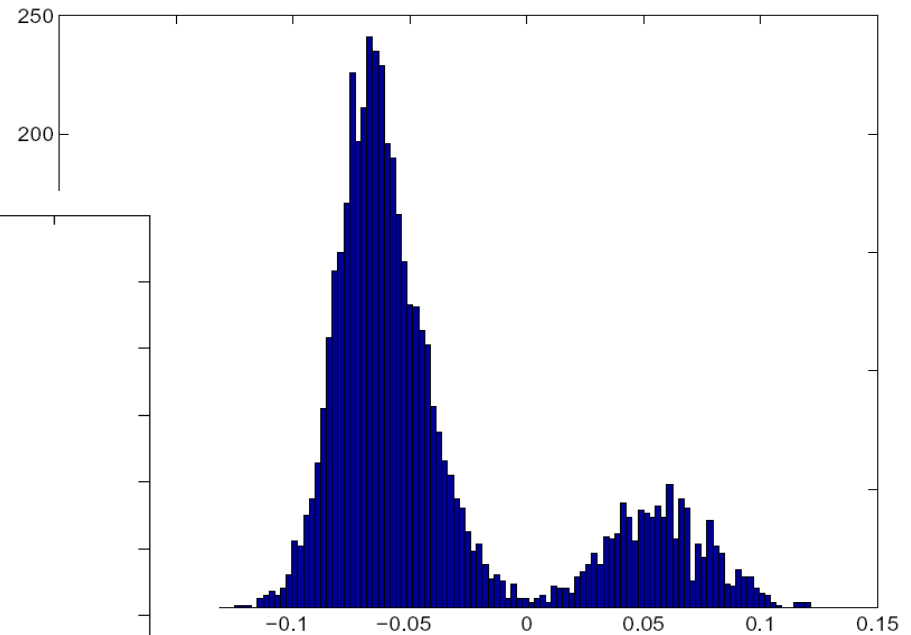
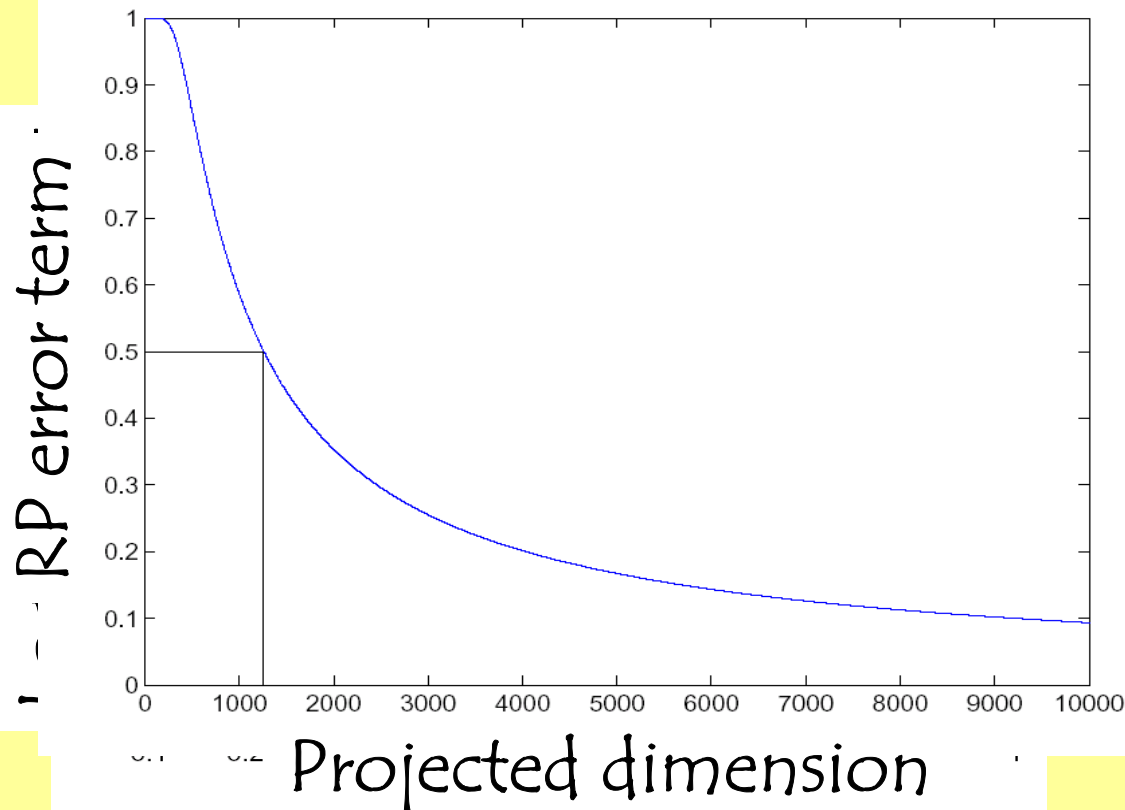
Generalization Bound, $l \sim N$



Bound dominated by VC component in the projected dimension

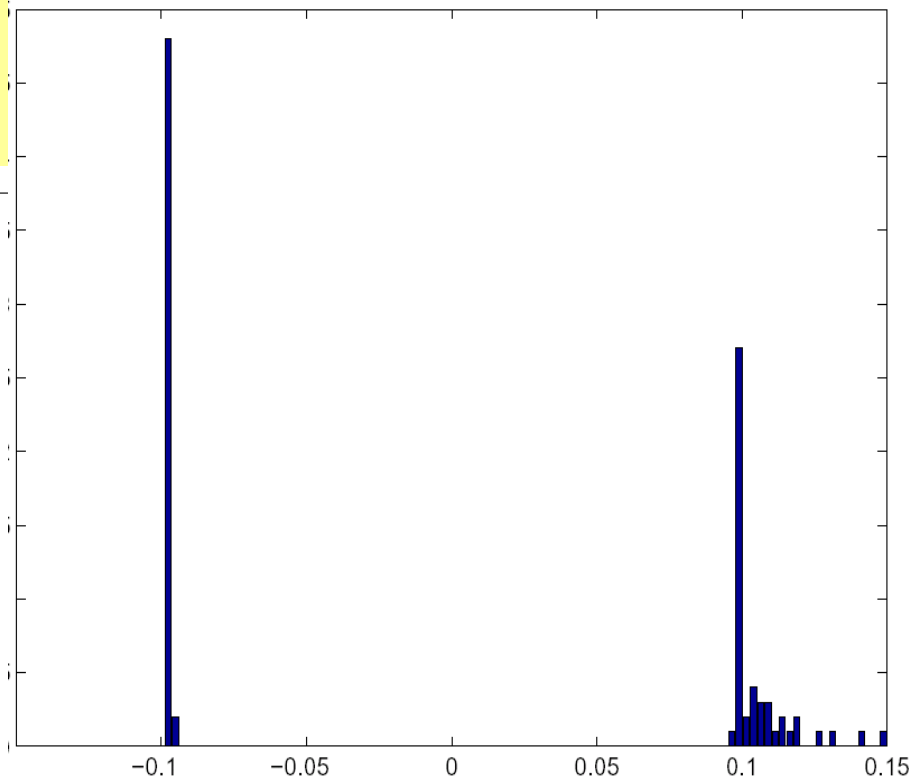
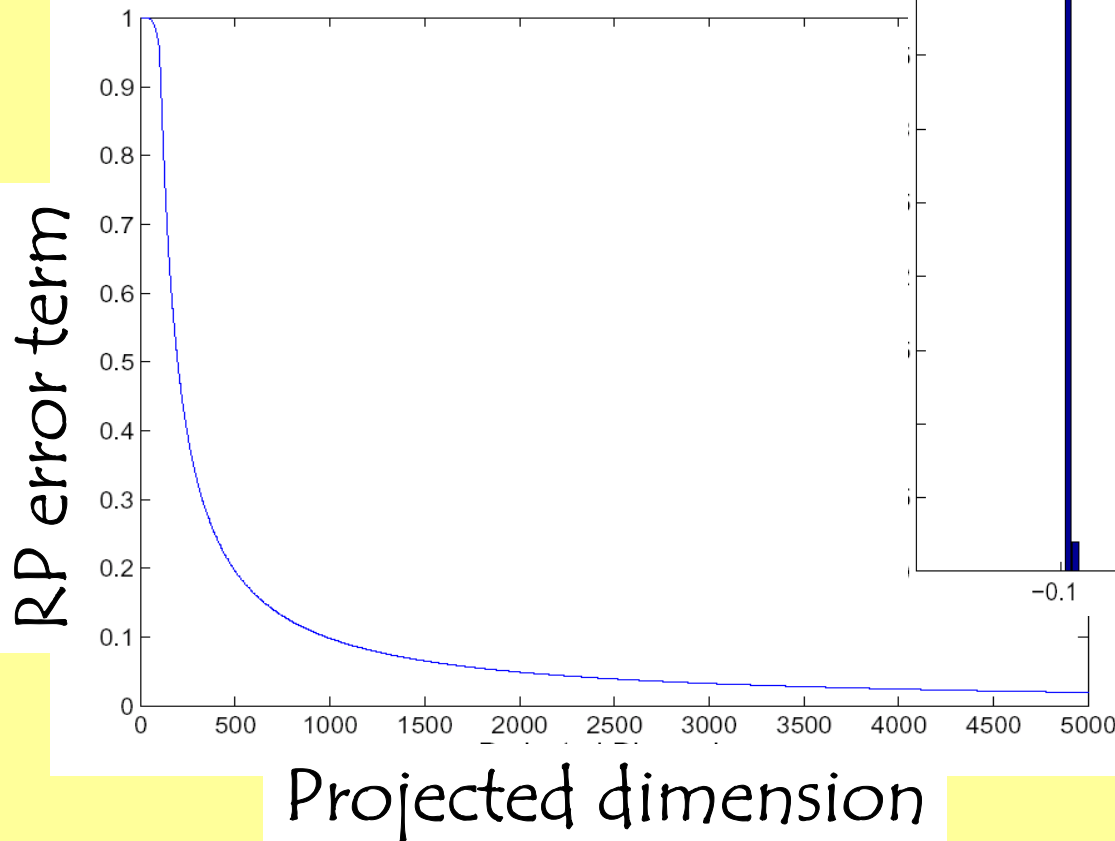
Real Data (I)

17,000 context
Sensitive spelling



Real Data (II)

RBF kernel face detection
Infinite dimension



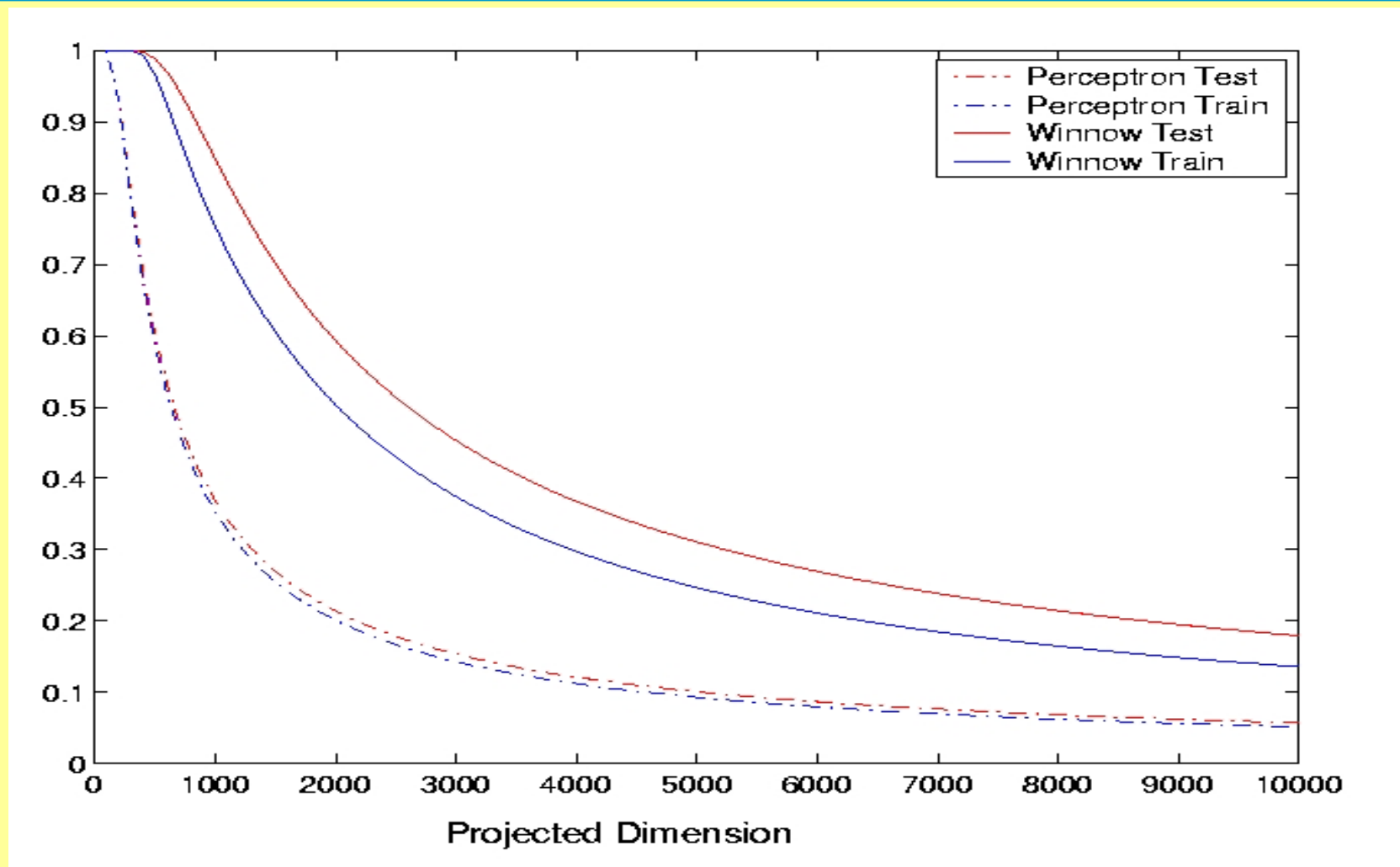
Conclusions

- ◇ Understanding learning in high dimensional spaces
- ◇ Analysis of error based on
 - Prediction preserving projection into low dimension
 - Standard VC argument at low dimension
- ◇ Projection profile
 - depends on distribution of distance of points to hyperplane
- ◇ Gives informative bounds for some real world (very) high dimensional problems
- ◇ Algorithmic implications? Better than random proj. ?

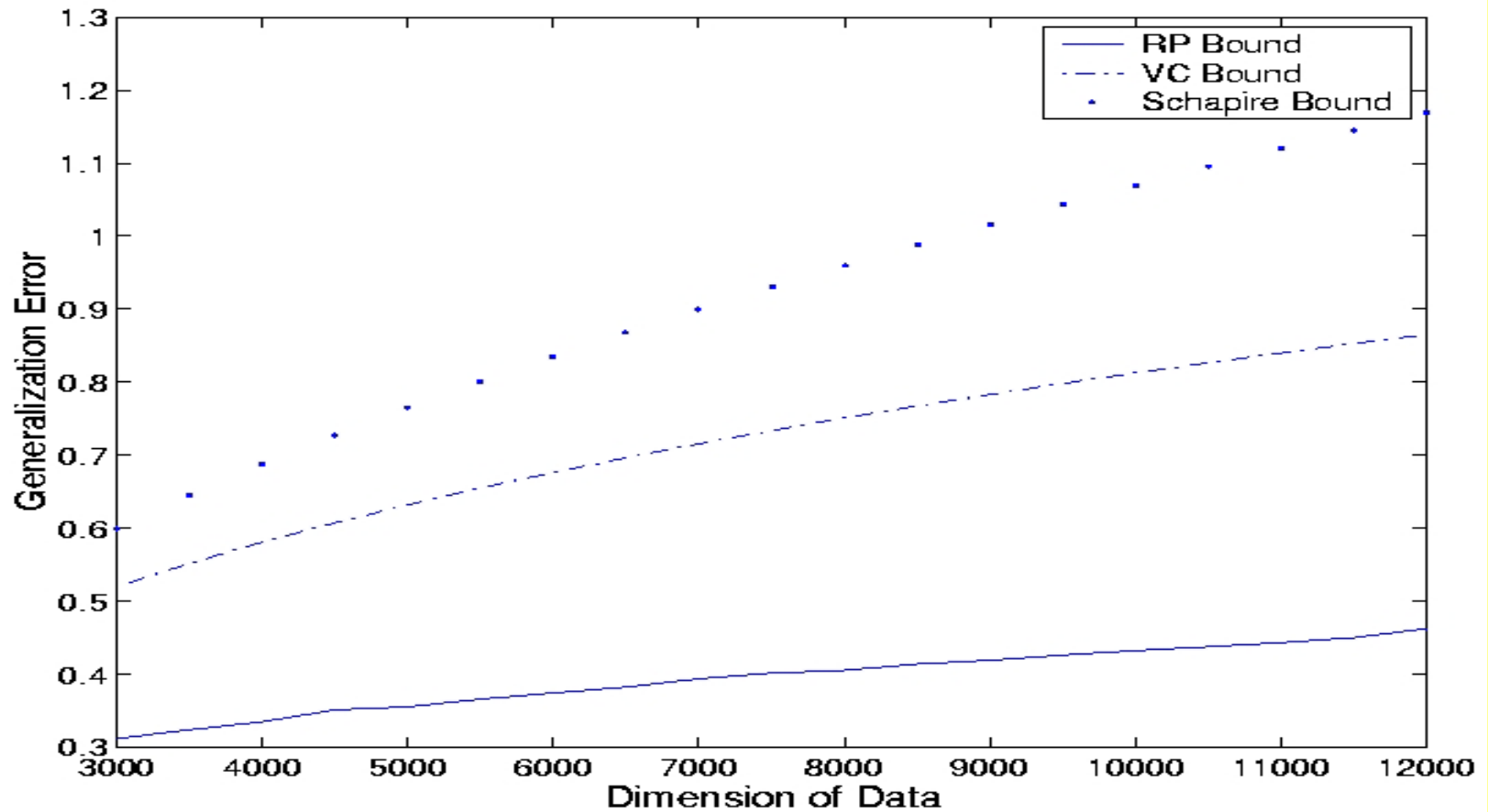
Puzzle

- ◇ Is it really the margin?
- ◇ Example: Winnow vs. Perceptron.
- ◇ Perceptron tries to maximize the margin; Winnow does not.
- ◇ Indeed, Winnow's margin distribution is worse.
- ◇ Yet, Winnow performs consistently better.

Puzzle



Comparison



Real Generalization

