Performance of Approximate Nearest Neighbor Classification

Gregory Shakhnarovich
Brown University

John W. Fisher
Massachusetts Institute of Technology

Introduction

Nearest-neighbor (NN) classifiers are often accurate but prohibitively expensive due to the cost of search. Recently proposed algorithms allow for much faster search at the cost of settling for an approximate, rather than exact, NN. We investigate the effect such approximations have on the classification error.

Problem definition

- We consider *binary* classification problems: data $X_N = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ drawn i.i.d. from $f(\mathbf{x}, y)$ over $\mathbb{R}^d \times \{1, 2\}$.
- Priors: $p_c = \Pr(y = c)$
- Compound density: $f(\mathbf{x}) = p_1 f_1(\mathbf{x}) + p_2 f_2(\mathbf{x})$.
- Posterior: $\Pr(y = c|\mathbf{x}) = \eta_c(\mathbf{x})$; shorthand $\eta \equiv \eta(\mathbf{x}) \equiv \eta_1(\mathbf{x})$.
- Test point $(\mathbf{x}_0, y_0) \sim f(\mathbf{x})$.
- NN classifier: find $\mathbf{x}' \in X_N$ such that

$$\rho = \|\mathbf{x}_0 - \mathbf{x}'\| = \min_{\mathbf{x} \in X_N} \|\mathbf{x}_0 - \mathbf{x}\|$$

  and predict $\widehat{y}_0 := y'$.
- ϵ-NN classifier: $\widehat{y}_0 := y'_\epsilon$ where

$$\|\mathbf{x}_0 - \mathbf{x}'_\epsilon\| \leq (1 + \epsilon)\rho.$$

  Note: the random variable ρ depends on f, N and $\mathbf{x}_0$.

Known results

- Conditional Bayes risk: $R^*(\mathbf{x}_0) = \min\{\eta, 1 - \eta\}$.
- Bayes risk is $R^* = E_{\mathbf{x}_0}[R^*(\mathbf{x}_0)]$
- NN risk for N-sample: $R_N = E_{\mathbf{x}_0, X}[R(\mathbf{x}_0;\ X_N)]$
- Cover and Hart's *asymptotic* bound [3]:

$$R_\infty \leq 2R^*(1 - R^*)$$

  Key idea of the proof: $\lim_{N\to\infty} \rho(N) = 0$, and so $y' \sim \eta$. Then, $R_\infty(\mathbf{x}_0) = 2\eta(1 - \eta)$, and the inequality follows by taking the expectation (and considering the variance term).
- Convergence of R_N to R_∞ can be arbitrarily slow [2, 4]; no distribution-independent results for R_N are known.

The question we are interested in:
How much worse is R_N^ϵ compared to R_N?
What is the accuracy/speed tradeoff between exact and approximate NN classification?

Why use an ϵ-NN classifier?

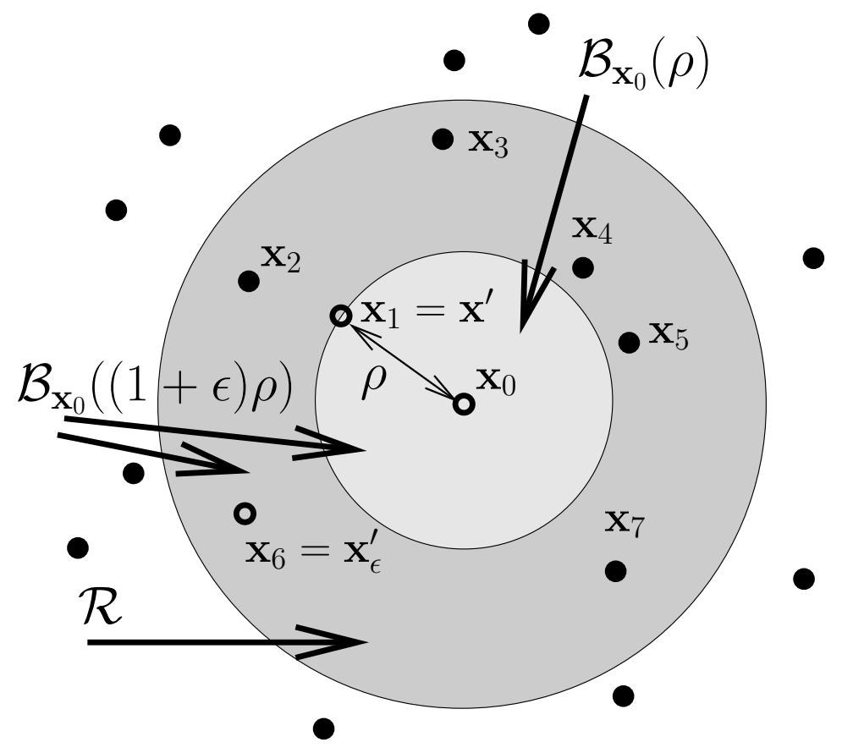
- In high dimensions exact NN search is reduced to brute force (linear) search.
- Locality sensitive hashing: search in $O(dN^{1/1+\epsilon})$.
- Newer algorithm [1]: $O(N^{1/(1+\epsilon)^2+o(1)}$.
- Other algorithms exist (Best Bin First, ANN, etc.), but with no known theoretical guarantees.

The computational model of ϵ-NN

- The precise underlying model of choosing $\mathbf{x}'_\epsilon$ in "real" algorithms like LSH is not known. Empirically it seems to be biased towards lower $\|\mathbf{x}_0 - \mathbf{x}'_\epsilon\|$.

A simplifying model used in our experiments:

- Let $\mathbf{x}'$ be the exact NN of $\mathbf{x}_0$ in X_N, and let L be the number of $\mathbf{x} \in X$ s.t. $\mathbf{x} \in \mathcal{B}((1 + \epsilon)\rho$.
- We assume that the classifier selects one of them with probability $1/L$, and uses its label to predict y_0.


$\mathcal{B}_{\mathbf{x}_0}(\rho)$
$\mathbf{x}_3$
$\mathbf{x}_2$
$\mathbf{x}_4$
$\mathbf{x}_1 = \mathbf{x}'$
$\mathbf{x}_5$
ρ
$\mathbf{x}_0$
$\mathcal{B}_{\mathbf{x}_0}((1+\epsilon)\rho)$
$\mathbf{x}_7$
$\mathbf{x}_6 = \mathbf{x}'_\epsilon$
$\mathcal{R}$

With prob. $1/7$ $\mathbf{x}'_\epsilon = \mathbf{x}_i$, for each $i = 1, \ldots, 7$.

Equivalently, with prob $6/7$ $\mathbf{x}'_\epsilon \sim f(\mathbf{x}|\mathcal{R})$.

For our ongoing theoretical analysis, we use the following "inverse" sampling model:

1. Draw test point $(\mathbf{x}_0, y_0) \sim f(\mathbf{x}, y)$
2. Draw distance to NN $\rho \sim p(\rho|\mathbf{x}_0, N;\ f)$. This defines the probability mass $P_\mathcal{B} = \int_{B_{\mathbf{x}_0}((1+\epsilon)\rho)} f(\mathbf{x}) d\mathbf{x}$.
3. Draw L' from $\texttt{Binomial}\,(N - 1, P_\mathcal{B})$. $L = L' + 1$ would be the number of ϵ-NN of $\mathbf{x}_0$ (including $\mathbf{x}'$).
4. With probability $1/L$ the classifier sets $\mathbf{x}'_\epsilon = \mathbf{x}'$.
5. With probability $1 - 1/L$, $\mathbf{x}'_\epsilon$ is drawn from $f\left(\mathbf{x} \mid \mathbf{x} \in \mathcal{B}_{\mathbf{x}_0}((1 + \epsilon)\rho) \setminus \mathcal{B}_{\mathbf{x}_0}(\rho)\right)$.
6. Draw $\widehat{y}_0$ from $f(y|\mathbf{x}'_\epsilon)$.

Asymptotic behavior of ϵ-NN

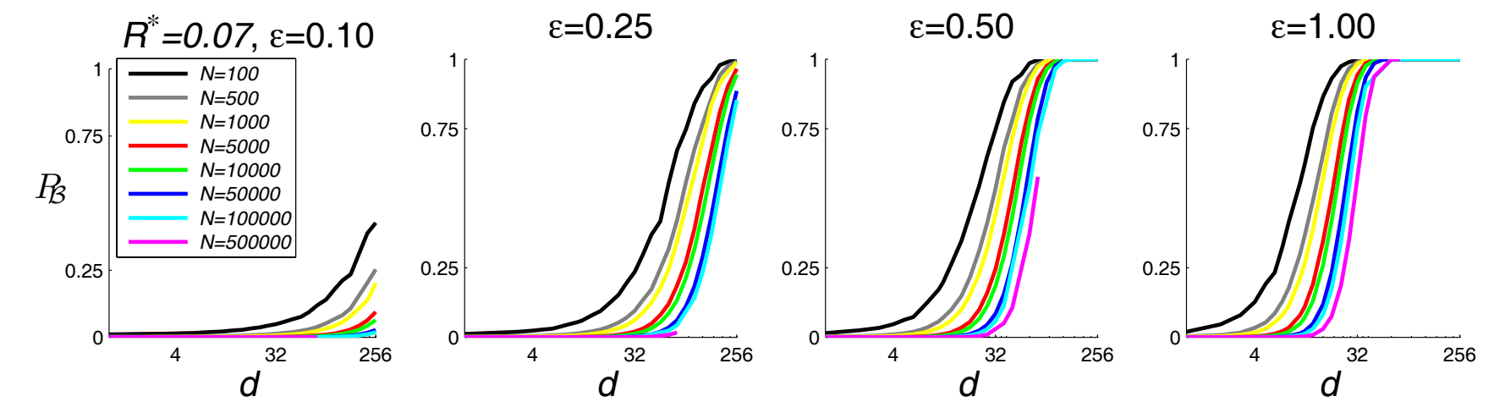
- If $\lim_{N\to\infty} \rho = 0$ then also $\lim_{N\to\infty}(1 + \epsilon)\rho = 0$ (by dominated convergence theorem).
- Thus, we can extend Cover's asymptotic result to ϵ-NN:

$$R_\infty^\epsilon = R_\infty.$$

Experiments

Gaussians: full covariance

- Both classes: $f_c(\mathbf{x}) = N(\mathbf{x};\ \mu_c, \sigma^2\mathbf{I})$.
- Bayes risk $R^* = \frac{1}{2}\left[1 - \texttt{erf}\left(\sqrt{2}\|\mu_1 - \mu_2\|/4\sigma\right)\right]$.
- The mass of $\mathcal{B}_{\mathbf{x}_0}((1 + \epsilon)\rho)$ grows too fast:



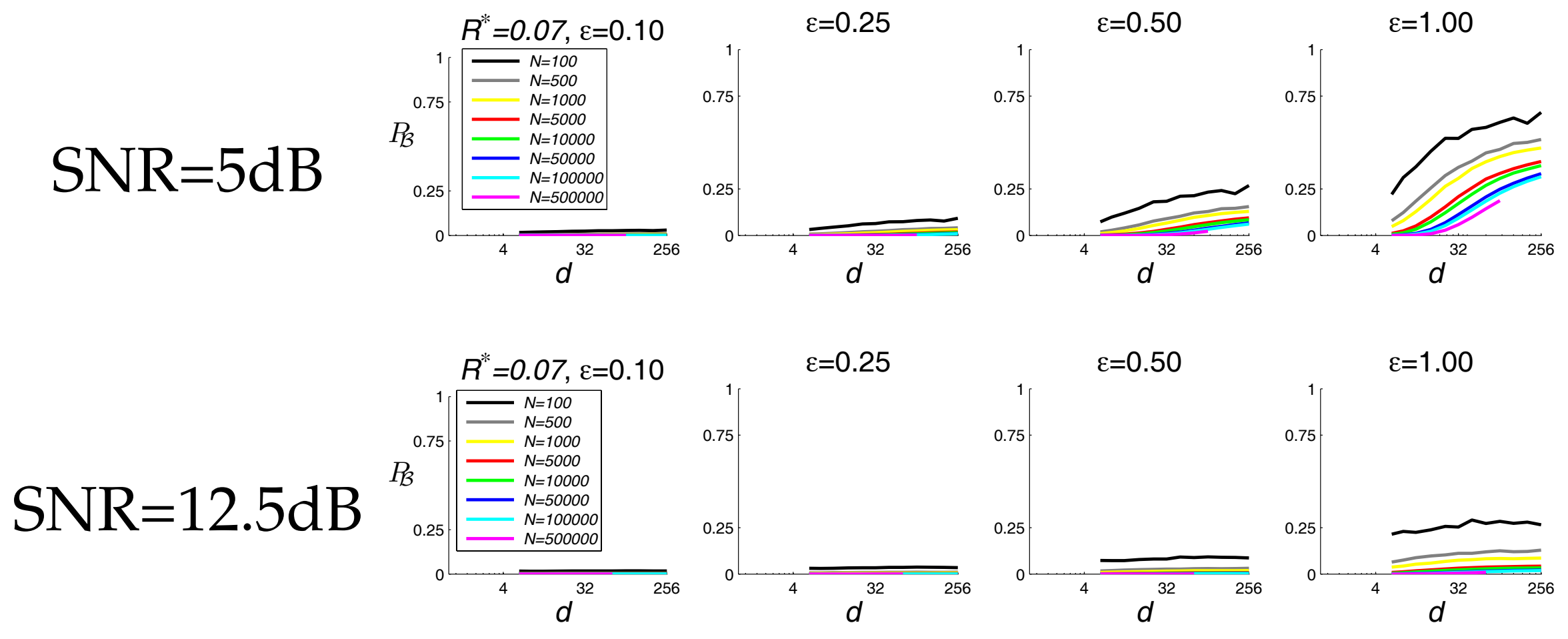
- Accuracy results (not shown) reflect this: even with $\epsilon = 0.1$ and $N = 500$, most of the training set is included in $\mathcal{B}_{\mathbf{x}_0}((1 + \epsilon)\rho)$, and the classifier is reduced to guessing.

Gaussians: low intrinsic dimension

- The protocol: embed 5-dimensional Gaussian in a linear subspace of $\mathbb{R}^d$, with Gaussian noise:

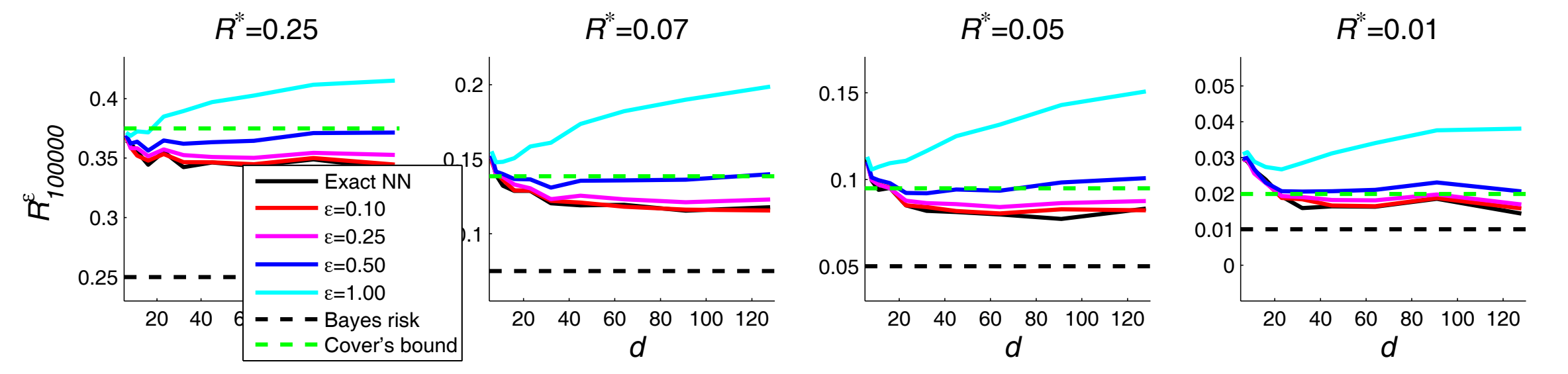
$$f_c(\mathbf{x}) = N\left(\mathbf{x};\ \mu_c, \begin{bmatrix} \mathbf{I}_5 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\right) + N(\mathbf{x};\ 0, \sigma_n \mathbf{I}).$$

- σ_n set to achieve desired SNR=$10\log_1 0(5/d\sigma^2)$.
- More reasonable behavior of $P_\mathcal{B}$:

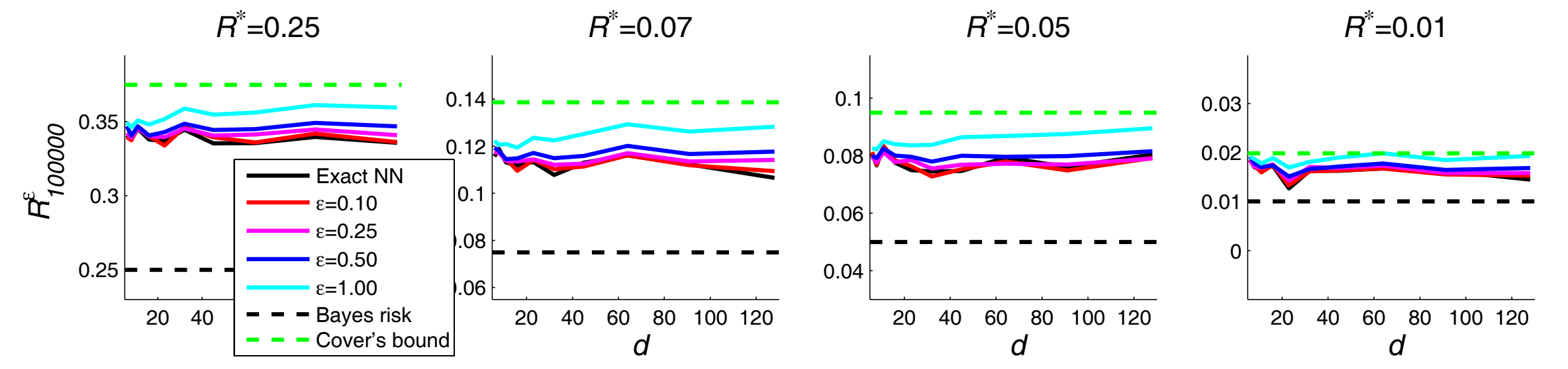


- Accuracy of classification, $N = 100000$

5dB

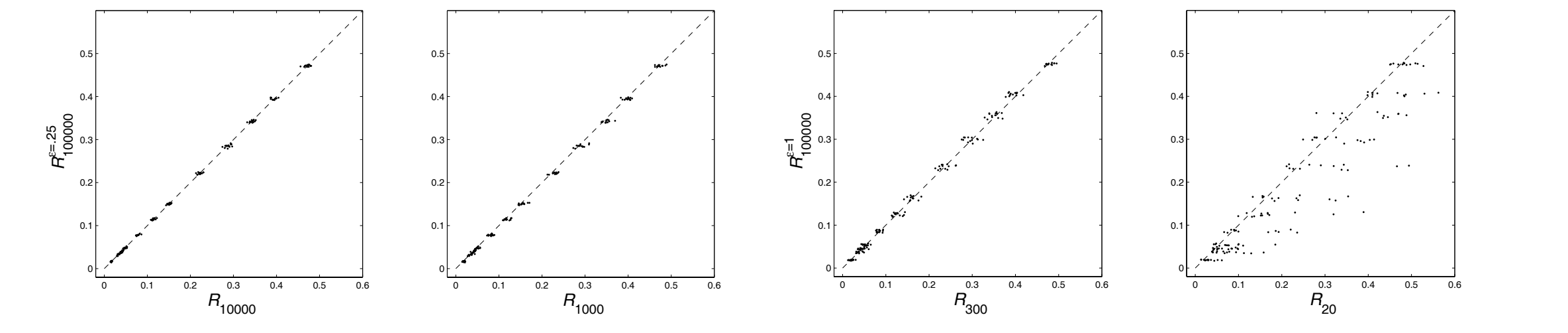


12.5dB



- A "fair" comparison (with equal computation):

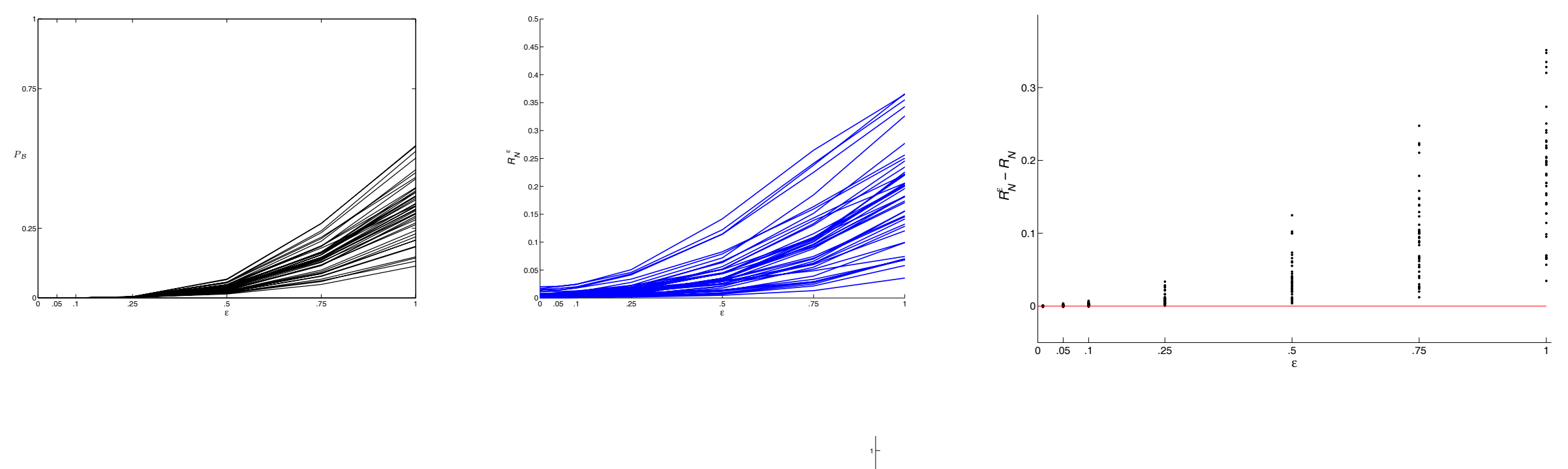
$\epsilon = .25$ $\epsilon = 1$



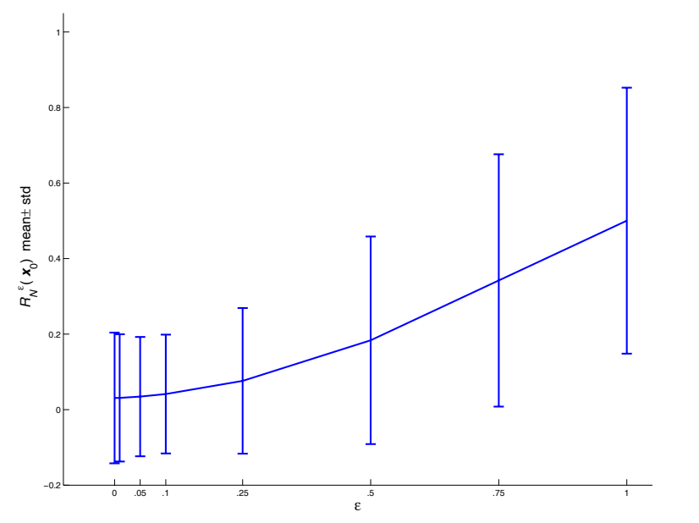
MNIST data

- 28×28 grayscale images of handwritten digits.
- 45 binary classification problems; $N \approx 12,000$

$P_\mathcal{B}$ R^ϵ_{100000} $R^\epsilon_{100000} - R_{100000}$



- 10-class problem (*does not exactly comply with assumptions here*); $N = 60,000$.



Conclusions

- When intrinsic dimension of data is high, ϵ-NN becomes meaningless even for small ϵ.
- When there is low dimensional structure in data, using moderate values of ϵ incurs only limited loss in accuracy for large N.
- For small ϵ there may be a small gain in performance, (we conjecture it is due to reduced variance of the risk).
- Theoretical analysis (current work):
  - Distribution-specific bounds on R_N^ϵ, similar to [6, 7].
  - Distribution-independent bounds. Quantities of interest: $R_N^\epsilon - R_N$,R_N^ϵ/R_N, or $(R_N^\epsilon - R_N)/R_\infty$like in [5].
  - Adjustment of the overly pessimistic sampling model to a particular search algorithm, e.g. LSH.

References

[1] A. Andoni and P. Indyk. New LSH-based algorithm for approximate nearest neighbor. Technical Report MIT-CSAIL-TR-2005-073, MIT, Cambridge, MA, December 2005.

[2] T. M. Cover. Rates of Convergence for Nearest Neighbor Procedures. In *Proc. 1st Ann. Hawaii Conf. Systems Theory*, pages 413–415, January 1968.

[3] T. M. Cover and P. E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13:21–27, January 1967.

[4] L. Devroye. On the inequality of Cover and Hart in nearest neighbor discrimination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3:75–78, 1981.

[5] K. Fukunaga and D. M. Hummels. Bias of nearest neighbor error estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(1):103–112, January 1987.

[6] D. Psaltis, R. R. Snapp, and S. S. Venkatesh. On the Finite Sample Performance of the Nearest Neighbor Classifier. *IEEE Transactions on Information Theory*, 40(3):820–837, May 1994.

[7] R. R. Snapp and S. S. Venkatesh. Asymptotic derivation of the finite-sample risk of the k nearest neighbor classifier. Technical Report UVM-CS-1998-0101, University of Vermont, Burlingotn, Burlington, VT, October 1997.