Learning to benchmark

Alfred Hero

Dept of Electrical Engineering and Computer Science (EECS)
Dept of Biomedical Engineering (BME)
Dept of Statistics
Program in Applied and Interdisciplinary Mathematics
Program in Applied Physics
Program in Computational Medicine and Bioinformatics
University of Michigan - Ann Arbor

Jan 17, 2020

Acknowledgements

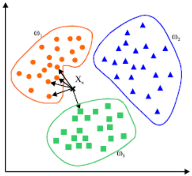
Students and former students who helped develop "learning to benchmark"

- Morteza Noshad - Stanford
- Easton Xu - Chinese Academy of Science
- Kevin Moon - Utah State
- Kumar Sricharan - Intuit, Inc
- Kristjan Greenewald - IBM AI
- Dennis Wei - IBM AI

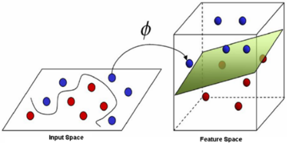
Sponsors

- AFRL ATR-Center Program
- ARO: MURI's on Value of Information, Non-Commutative Information, and Networked Interactions Governing Community Dynamics Programs
- DARPA Predicting Health and Disease, Biochronicity and Prometheus Programs
- NSF: Theoretical Foundations Program
- NIH: Biomedical Imaging Institute
- DOE: NNSA Consortia for Verfication Technology (CVT) and Exploitation of Technolological Innovations (ETI)

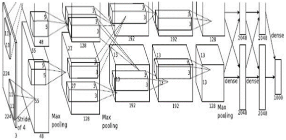
Benchmarks in Machine Learning



kNN classifier

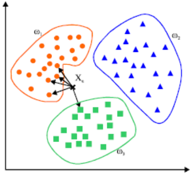


SVM classifier

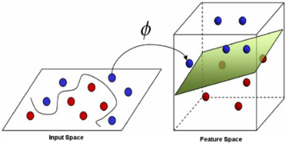


CNN classifier

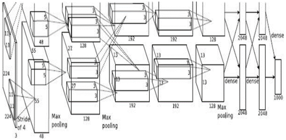
Benchmarks in Machine Learning



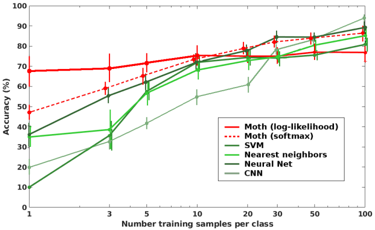
kNN classifier



SVM classifier

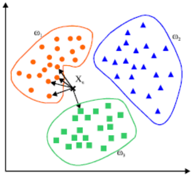


CNN classifier

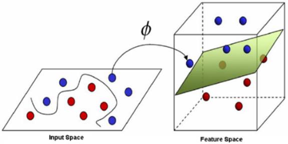


Accuracy for MNIST (Delahunt *et al* 2019)

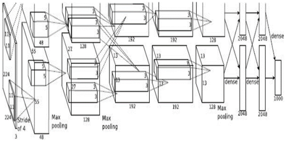
Benchmarks in Machine Learning



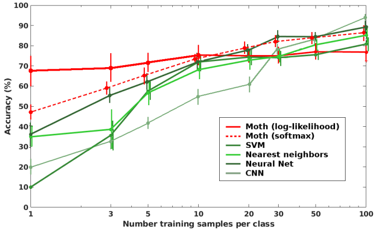
kNN classifier



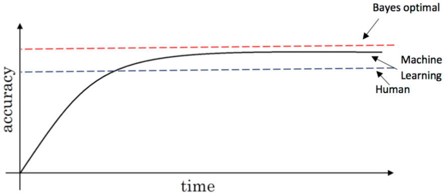
SVM classifier



CNN classifier



Accuracy for MNIST (Delahunt *et al* 2019)



Bayes-optimal benchmark
(Andrew Ng's Blog, Dec. 22, 2018)

Learning to benchmark classification accuracy without learning a classifier

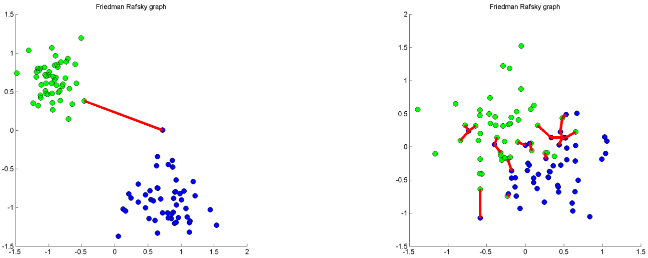


Figure: Friedman-Rafsky statistic converges to bound on Bayes classification error.

- Q: Can we find data-driven empirical upper and lower bounds on Bayes error?

[1] J. Friedman and L. Rafsky (1979), Multivariate generalizations of the Wald-Wolfowitz and Smirnov two-sample tests. The Annals of Statistics.

[2] N. Henze and M. D. Penrose (1999). On the multivariate runs test. Annals of Statsistics.

[3] V. Berisha, A. Wisler, A.O. Hero, and A. Spanias (2016), Empirically Estimable Classification Bounds Based on a Nonparametric Divergence Measure IEEE Transactions on Signal Processing.

Learning to benchmark classification accuracy without learning a classifier

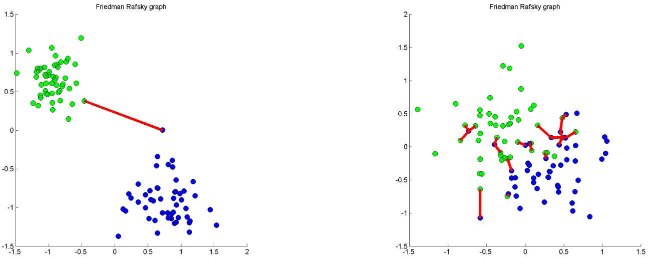


Figure: Friedman-Rafsky statistic converges to bound on Bayes classification error.

- Q: Can we find data-driven empirical upper and lower bounds on Bayes error?
- A: The FR statistic[1] directly estimates a bound[2] on Bayes error[3]

[1] J. Friedman and L. Rafsky (1979), Multivariate generalizations of the Wald-Wolfowitz and Smirnov two-sample tests. The Annals of Statistics.

[2] N. Henze and M. D. Penrose (1999). On the multivariate runs test. Annals of Statsistics.

[3] V. Berisha, A. Wisler, A.O. Hero, and A. Spanias (2016), Empirically Estimable Classification Bounds Based on a Nonparametric Divergence Measure IEEE Transactions on Signal Processing.

Benchmarking performance of Bayes classifier

Consider classification problem

- $Y \in \{0, 1\}$ an unknown label with priors $\{q, p\}$, $p + q = 1$.

$$P(Y = k) = p^k q^{1-k}, \quad k = 0, 1$$

- X an observed random variable with conditional distribution

$$f(x|Y = k) = [f_1(x)]^k [f_0(x)]^{1-k}, \quad k = 0, 1$$

Benchmarking performance of Bayes classifier

Consider classification problem

- $Y \in \{0,1\}$ an unknown label with priors $\{q,p\}$, $p+q=1$.

$$P(Y=k) = p^k q^{1-k}, \quad k=0,1$$

- X an observed random variable with conditional distribution

$$f(x|Y=k) = [f_1(x)]^k [f_0(x)]^{1-k}, \quad k=0,1$$

Let $C(x)$ be (Bayes) optimal classifier that minimizes avg 0-1 loss (probability of error)

$$C(x) = \mathrm{argmax}_{k\in\{0,1\}}\{P(Y=k|X=x)\}$$

Bayes error rate: best achievable misclassification error probability

Bayes error rate is avg missclassification error probability of Bayes classifier

$$\epsilon_p(f_0, f_1) = P(C(X) \neq Y)$$

Bayes error rate: best achievable misclassification error probability

Bayes error rate is avg missclassification error probability of Bayes classifier

$$\epsilon_p(f_0, f_1) = P(C(X) \neq Y)$$

Integral representation

$$\epsilon_p(f_0, f_1) = \frac{1}{2} - \frac{1}{2}\int |qf_0(x) - pf_1(x)|dx,$$

Bayes error rate: best achievable misclassification error probability

Bayes error rate is avg missclassification error probability of Bayes classifier

$$\epsilon_p(f_0, f_1) = P(C(X) \neq Y)$$

Integral representation

$$\epsilon_p(f_0, f_1) = \frac{1}{2} - \frac{1}{2} \int |q f_0(x) - p f_1(x)| dx,$$

Alternative representation as an f-divergence btwn distributions

$$\epsilon_p(f_0, f_1) = \frac{1 + |p - q|}{2} - \frac{1}{2} \int g(f_1(x)/f_0(x)) f_0(x) dx,$$

where $g(u)$ is the convex function

$$g(u) = |pu - q| - |p - q|.$$

The f-divergence between a pair of distributions

The f-divergence (Csiszár)[1], (Ali-Silvey)[2]:

$$D_g(f_1 \| f_0) = \int g\left(\frac{f_1(x)}{f_0(x)}\right) f_0(x) dx$$

where $g(u)$ is a convex function on $\mathbb{R}^+$ and $g(1) = 0$.

[1] I. Csiszár (1963), Eine informationstheoretische Ungleichung und ihre Anwendung auf den Beweis der Ergodizitat von Markoffschen Ketten. Magyar. Tud. Akad. Mat. Kutato Int. Kozl. 8:85108.

[2] S. M. Ali and S. D. Silvey (1966), A general class of coefficients of divergence of one distribution from another, J. Royal Stat. Soc., Ser.B , 28:131-142.

The f-divergence between a pair of distributions

The f-divergence (Csiszár)[1], (Ali-Silvey)[2]:

$$D_g(f_1 \| f_0) = \int g\left(\frac{f_1(x)}{f_0(x)}\right) f_0(x)dx$$

where $g(u)$ is a convex function on $\mathbb{R}^+$ and $g(1) = 0$.

Properties: if g is strictly convex then $D_g(f_1\|f_0)$ is

- reflexive non-negative: $D_g(f_1\|f_0) \geq 0$ with equality iff $f_1 = f_0$
- monotone: $D_g(f_1\|f_0)$ non-increasing under transformations $x \to T(x)$
- jointly convex: $D_g(f_1\|f_0)$ is convex in (f_0, f_1)

[1] I. Csiszár (1963), Eine informationstheoretische Ungleichung und ihre Anwendung auf den Beweis der Ergodizitat von Markoffschen Ketten. Magyar. Tud. Akad. Mat. Kutato Int. Kozl. 8:85108.

[2] S. M. Ali and S. D. Silvey (1966), A general class of coefficients of divergence of one distribution from another, J. Royal Stat. Soc., Ser.B , 28:131-142.

The f-divergence between a pair of distributions

The f-divergence (Csiszár)[1], (Ali-Silvey)[2]:

$$D_g(f_1 \| f_0) = \int g\left(\frac{f_1(x)}{f_0(x)}\right) f_0(x)dx$$

where $g(u)$ is a convex function on $\mathbb{R}^+$ and $g(1) = 0$.

Properties: if g is strictly convex then $D_g(f_1\|f_0)$ is

- reflexive non-negative: $D_g(f_1\|f_0) \geq 0$ with equality iff $f_1 = f_0$
- monotone: $D_g(f_1\|f_0)$ non-increasing under transformations $x \rightarrow T(x)$
- jointly convex: $D_g(f_1\|f_0)$ is convex in (f_0, f_1)

Examples: $g(u) = u\log(u)$ (KL); $g(u) = (1 - u^\alpha)\frac{1}{1-\alpha}$ (Rényi-α), etc.

[1] I. Csiszár (1963), Eine informationstheoretische Ungleichung und ihre Anwendung auf den Beweis der Ergodizitat von Markoffschen Ketten. Magyar. Tud. Akad. Mat. Kutato Int. Kozl. 8:85108.

[2] S. M. Ali and S. D. Silvey (1966), A general class of coefficients of divergence of one distribution from another, J. Royal Stat. Soc., Ser.B , 28:131-142.

Instances of f-divergences

The following are common instances of f-divergence[1]

- Total variation distance $g(u) = \frac{1}{2}|u-1|$

$$D^{TV}(f_1\|f_2) = \frac{1}{2}\int |f_1(x) - f_2(x)|dx$$

[1] Csiszár, I., and Shields, P. C. (2004). Information theory and statistics: A tutorial. Foundations and Trends in Communications and Information Theory, 1(4), 417-528.

Instances of f-divergences

The following are common instances of f-divergence[1]

- Total variation distance $g(u) = \frac{1}{2}|u-1|$

$$D^{TV}(f_1\|f_2) = \frac{1}{2}\int |f_1(x) - f_2(x)|dx$$

- α-divergence: $g(u) = (1-u^\alpha)\frac{1}{1-\alpha}$

$$D^R(f_1\|f_2) = \left(1 - \int f_1^\alpha(x) f_2^{1-\alpha}(x)dx\right)\frac{1}{1-\alpha}$$

[1] Csiszár, I., and Shields, P. C. (2004). Information theory and statistics: A tutorial. Foundations and Trends in Communications and Information Theory, 1(4), 417-528.

Instances of f-divergences

The following are common instances of f-divergence[1]

- Total variation distance $g(u) = \frac{1}{2}|u-1|$

$$D^{TV}(f_1\|f_2) = \frac{1}{2}\int |f_1(x) - f_2(x)|dx$$

- α-divergence: $g(u) = (1-u^{\alpha})\frac{1}{1-\alpha}$

$$D^{R}(f_1\|f_2) = \left(1 - \int f_1^{\alpha}(x) f_2^{1-\alpha}(x)dx\right)\frac{1}{1-\alpha}$$

- Kullback-Liebler divergence: $g(u) = u\log u$:

$$D^{KL}(f_1\|f_2) = \int f_1(x)\log\left(\frac{f_1(x)}{f_2(x)}\right)dx$$

[1] Csiszár, I., and Shields, P. C. (2004). Information theory and statistics: A tutorial. Foundations and Trends in Communications and Information Theory, 1(4), 417-528.

Instances of f-divergences

The following are common instances of f-divergence[1]

- Total variation distance $g(u) = \frac{1}{2}|u-1|$

$$D^{TV}(f_1\|f_2) = \frac{1}{2}\int |f_1(x) - f_2(x)|dx$$

- α-divergence: $g(u) = (1-u^{\alpha})\frac{1}{1-\alpha}$

$$D^{R}(f_1\|f_2) = \left(1 - \int f_1^{\alpha}(x) f_2^{1-\alpha}(x)dx\right)\frac{1}{1-\alpha}$$

- Kullback-Liebler divergence: $g(u) = u\log u$:

$$D^{KL}(f_1\|f_2) = \int f_1(x)\log\left(\frac{f_1(x)}{f_2(x)}\right)dx$$

- Hellinger-Bhattacharyya divergence $g(u) = (\sqrt{u}-1)^2$

$$D^{H}(f_1\|f_2) = \int \left(\sqrt{f_1(x)} - \sqrt{f_2(x)}\right)^2 dx$$

[1] Csiszár, I., and Shields, P. C. (2004). Information theory and statistics: A tutorial. Foundations and Trends in Communications and Information Theory, 1(4), 417-528.

Other instances of f-divergences

- Generalized total variation distance[1]: $g(u) = |pu - q|/2 - |p - q|/2$

$$D_p^{GTV} = \frac{1}{2}\int |pf_1(x) - qf_2(x)|dx + |p - q|/2$$

- Henze-Penrose divergence[2]: $g(u) = \frac{1}{4pq}\left[\frac{(pt-q)^2}{pt+q} - (p-q)^2\right]$

$$D_p^{HP} = \frac{1}{4pq}\left[\int \frac{(pf_1(x) - qf_2(x))^2}{pf_1(x) + qf_2(x)}dx - (p-q)^2\right].$$

[1] T. Kailath (1967), The divergence and Bhattacharyya distance measures in signal selection, IEEE T. Communication Technology, 15:1:5260

[2] N. Henze and M. D. Penrose (1999). On the multivariate runs test. Annals of Stats, 290-298.

f-divergences and Bayes error rate

These divergences can each be related to minimum probability of error

- Exact f-divergence representation

$$\epsilon_{p,q}(f_1, f_2) = \frac{1 + |p - q|}{2} - D_p^{GTV}(f_1(x) \| f_2(x))$$

[1] T. Kailath (1967), The divergence and Bhattacharyya distance measures in signal selection, IEEE T. Communication Technology, 15:1:5260

[2] Tishby, N., Pereira, F. C., and Bialek, W. (2000). The information bottleneck method. arXiv preprint physics/0004057.

[3] Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning (pp. 1096-1103). ACM.

f-divergences and Bayes error rate

These divergences can each be related to minimum probability of error

- Exact f-divergence representation

$$\epsilon_{p,q}(f_1, f_2) = \frac{1 + |p - q|}{2} - D_p^{GTV}(f_1(x) \| f_2(x))$$

- Bhattacharyya bound[1]

$$\frac{1}{2} - \frac{1}{2}\sqrt{1 - BC_p^2} \leq \epsilon_{p,q} \leq \frac{1}{2} BC_p,$$

where $BC_p = \frac{\sqrt{pq}}{2}(1 - D_p^H)$ is the Bhattacharyya coefficient

[1] T. Kailath (1967), The divergence and Bhattacharyya distance measures in signal selection, IEEE T. Communication Technology, 15:1:5260

[2] Tishby, N., Pereira, F. C., and Bialek, W. (2000). The information bottleneck method. arXiv preprint physics/0004057.

[3] Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning (pp. 1096-1103). ACM.

f-divergences and Bayes error rate

These divergences can each be related to minimum probability of error

- Exact f-divergence representation

$$\epsilon_{p,q}(f_1, f_2) = \frac{1 + |p - q|}{2} - D_p^{GTV}(f_1(x)\|f_2(x))$$

- Bhattacharyya bound[1]

$$\tfrac{1}{2} - \tfrac{1}{2}\sqrt{1 - BC_p^2} \leq \epsilon_{p,q} \leq \tfrac{1}{2} BC_p,$$

where $BC_p = \frac{\sqrt{pq}}{2}(1 - D_p^H)$ is the Bhattacharyya coefficient

- *Learning to benchmark* can be reduced to f-divergence estimation.
- f-divergences are widely used in signal processing[2] and machine learning[3].

[1] T. Kailath (1967), The divergence and Bhattacharyya distance measures in signal selection, IEEE T. Communication Technology, 15:1:5260

[2] Tishby, N., Pereira, F. C., and Bialek, W. (2000). The information bottleneck method. arXiv preprint physics/0004057.

[3] Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning (pp. 1096-1103). ACM.

HP bound tighter than Bhattacharya for $p = 1/2$. [1]

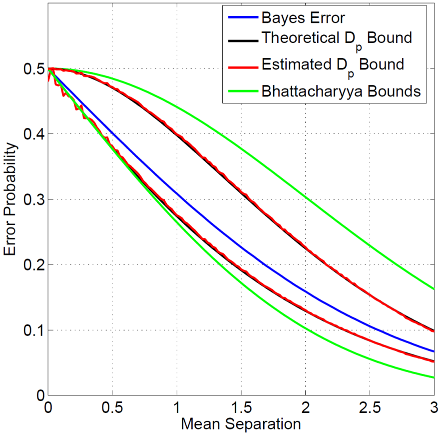


Figure: The HP bound using D_p is tighter than the Bhattacharrya bound using BC for bivariate normal distribution.

[1] V. Berisha, A. Wisler, A.O. Hero, and A. Spanias (2016), Empirically Estimable Classification Bounds Based on a Nonparametric Divergence Measure IEEE T. Signal Processing, 64:3:580-591.

Empirical Estimation of f-Divergence

- **Goal**: Accurate and computationally fast estimation of f-divergence
- **Density plug-in estimator of f-divergence:**

$$\widehat{D_g}(f_1 \| f_0) = \int g\left(\frac{\widehat{f_1}(x)}{\widehat{f_0}(x)}\right)\widehat{f_0}(x)dx$$

where

  - $\widehat{f_0}, \widehat{f_1}$ are density estimates, e.g., with kernel bandwidth parameter ϵ
  - Gabor kernel, histogram, k-NN kernel[1] (Devroye 2012)

- **Root mean squared error (RMSE)** decreases slowly in n=#samples

$$\mathrm{RMSE} = \sqrt{\mathrm{Bias}^2 + \mathrm{Variance}} = cn^{-1/2d}$$

[1] L. Devroye, G. Lugosi, "Combinatorial methods in density estimation," Springer 2012.

Empirical Estimation of f-Divergence

- **Goal**: Accurate and computationally fast estimation of f-divergence
- **Density plug-in estimator of f-divergence:**

$$\widehat{D_g}(f_1 \| f_0) = \int g\left(\frac{\widehat{f}_1(x)}{\widehat{f}_0(x)}\right)\widehat{f}_0(x)dx$$

where

- $\widehat{f}_0, \widehat{f}_1$ are density estimates, e.g., with kernel bandwidth parameter ϵ
- Gabor kernel, histogram, k-NN kernel[1] (Devroye 2012)

- **Root mean squared error (RMSE)** decreases slowly in n=#samples

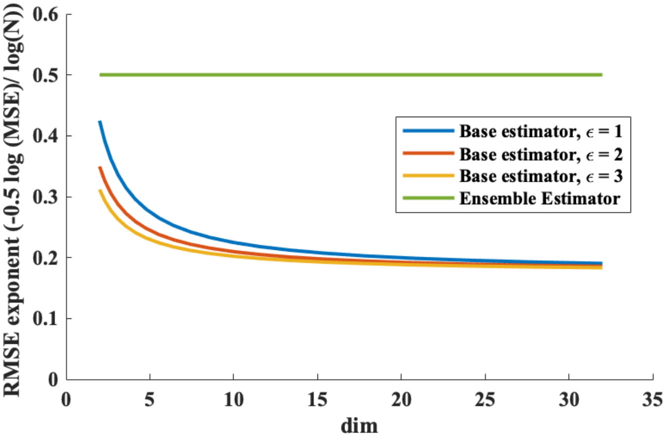
$$\mathrm{RMSE} = \sqrt{\mathrm{Bias}^2 + \mathrm{Variance}} = cn^{-1/2d}$$

$\Rightarrow$ Compare to optimal *parametric* RMSE rate:

$$\mathrm{RMSE} = \sqrt{\mathrm{MSE}} = cn^{-1/2}$$

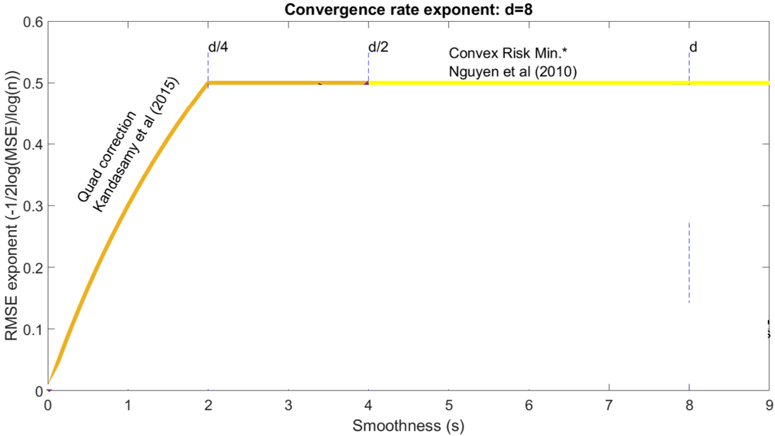
[1] L. Devroye, G. Lugosi, "Combinatorial methods in density estimation," Springer 2012.

Optimal $\sqrt{n}$ RMSE rates are achievable with ensemble estimation[1]



[1] K.R. Moon, K. Sricharan, K. Greenewald, and A.O. Hero (2018), "Ensemble Estimation of Information Divergence," Entropy

RMSE convergence rate comparisons[1,2]

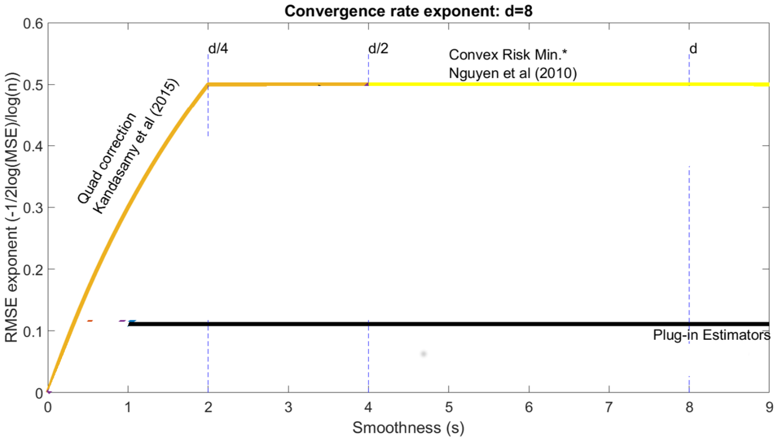


$\Rightarrow$ Kandasami's quadratic estimator achieves $O\left(\min\{n^{-3s/(2s+d)}, n^{-1/2}\}\right)$

[1] A. Krishnamurthy, K. Kandasamy, B. Póczos. Nonparametric estimation of Rényi divergence and friends, NIPS 2014.

[2] X. Nguyen, M. Wainwright, M. Jordan "Estimating divergence functionals and the likelihood ratio by convex risk minimization." IEEE Trans on Information Theory, 2010.

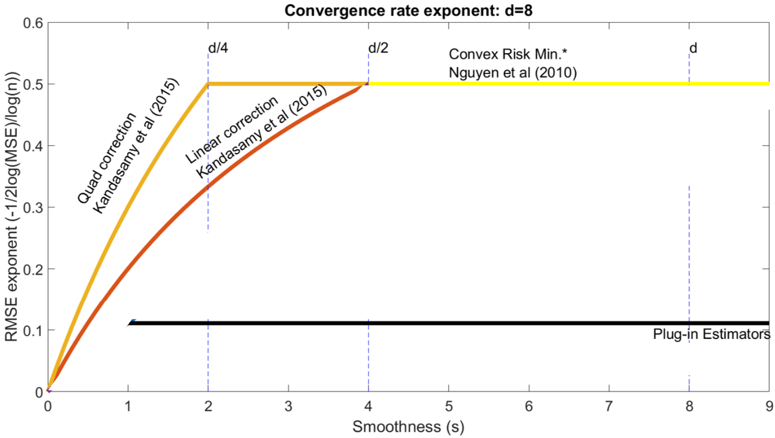
RMSE convergence rate comparisons[1]



⇒ Density plug-in is a weak learner: RMSE rate is $O(n^{-1/d})$

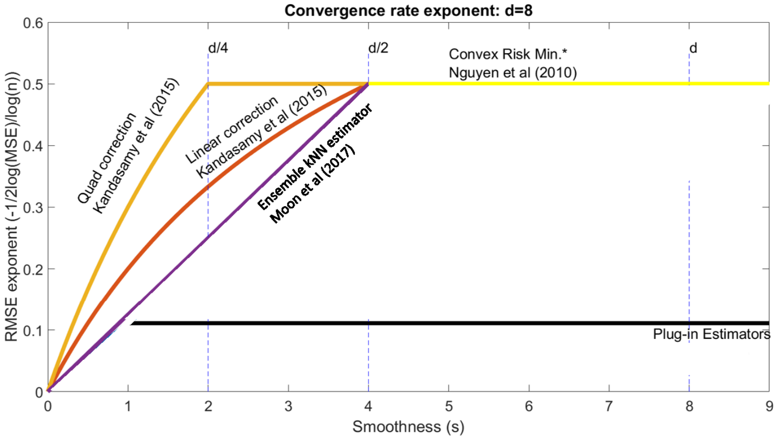
[1] K Moon, K Sricharan, K Greenewald, A. Hero. Ensemble Estimation of Information Divergence," Entropy, vol. 20, no. 8, p. 560, July 2018.

RMSE convergence rate comparisons[1]



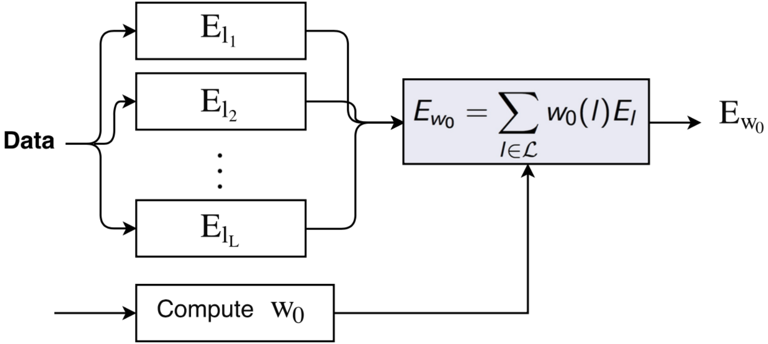
[1] A. Krishnamurthy, K. Kandasamy, B. Póczos. Nonparametric estimation of Rényi divergence and friends, NIPS 2014.

RMSE convergence rate comparisons[1]



[1] K Moon, K Sricharan, K Greenewald, A. Hero. Ensemble Estimation of Information Divergence," Entropy, vol. 20, no. 8, p. 560, July 2018.

Boosting ensembles concept



- $\{E_{l_i}\}_{i=1}^{L}$ ensemble of base estimators (weak learners)
- $\mathbf{w}_0 = (w_0(l))_{l=1}^{L}$ a vector of boosting weights
- $E_{\mathbf{w}_0}$: combined base estimators (boosted learner)

Choice of boosting weights

Most boosting approaches use *data-dependent* weights:

- Boosting classifiers with Adaboost[1] and other objective functions.

[1] Y. Freund and R. E. Schapire (1996). Experiments with a new boosting algorithm. Intl Conf on Machine Learning. pp. 148-156.

[2] Bickel, P. J., Ritov, Y. A., and Zakai, A. (2006). Some theory for generalized boosting algorithms. J. of Machine Learning Research, 705-732.

[3] Moon, Sricharan, Greenewald, Hero. "Ensemble estimation of information divergence." Entropy 20, no. 8 (2018): 560.

Choice of boosting weights

Most boosting approaches use *data-dependent* weights:

- Boosting classifiers with Adaboost[1] and other objective functions.

Under some conditions such methods achieve Bayes optimal performance[2]

[1] Y. Freund and R. E. Schapire (1996). Experiments with a new boosting algorithm. Intl Conf on Machine Learning. pp. 148-156.

[2] Bickel, P. J., Ritov, Y. A., and Zakai, A. (2006). Some theory for generalized boosting algorithms. J. of Machine Learning Research, 705-732.

[3] Moon, Sricharan, Greenewald, Hero. "Ensemble estimation of information divergence." Entropy 20, no. 8 (2018): 560.

Choice of boosting weights

Most boosting approaches use *data-dependent* weights:

- Boosting classifiers with Adaboost[1] and other objective functions.

Under some conditions such methods achieve Bayes optimal performance[2]

Alternative: can solve an *offline* inverse problem for rate-optimal weights[3]

[1] Y. Freund and R. E. Schapire (1996). Experiments with a new boosting algorithm. Intl Conf on Machine Learning. pp. 148-156.

[2] Bickel, P. J., Ritov, Y. A., and Zakai, A. (2006). Some theory for generalized boosting algorithms. J. of Machine Learning Research, 705-732.

[3] Moon, Sricharan, Greenewald, Hero. "Ensemble estimation of information divergence." Entropy 20, no. 8 (2018): 560.

Choice of boosting weights

Most boosting approaches use *data-dependent* weights:

- Boosting classifiers with Adaboost[1] and other objective functions.

Under some conditions such methods achieve Bayes optimal performance[2]

Alternative: can solve an *offline* inverse problem for rate-optimal weights[3]

This can be applied to different base estimation methods:

- Kernel density estimates (KDE)
- k-NN density estimates
- NN ratio estimates
- Locality sensitive hashing (LSH) density estimates

---

[1] Y. Freund and R. E. Schapire (1996). Experiments with a new boosting algorithm. Intl Conf on Machine Learning. pp. 148-156.

[2] Bickel, P. J., Ritov, Y. A., and Zakai, A. (2006). Some theory for generalized boosting algorithms. J. of Machine Learning Research, 705-732.

[3] Moon, Sricharan, Greenewald, Hero. "Ensemble estimation of information divergence." Entropy 20, no. 8 (2018): 560.

Locality sensitive hashing (LSH) plug-in estimator

$$\widehat{D}_g(f_1 \| f_0) := \sum_{i: M_i > 0} g\left(\frac{N_i/N}{M_i/M}\right) M_i/M$$

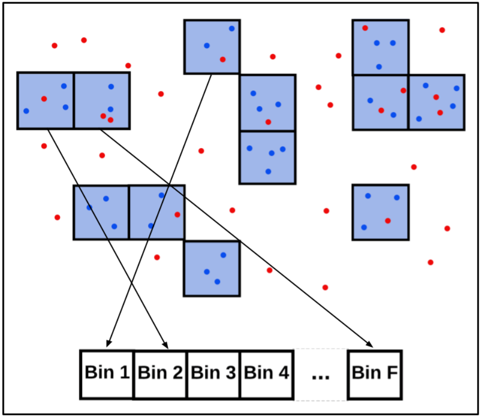


Figure: LSH quantizes **X** data with cell resolution ϵ and random displacement b

Locality sensitive hashing plug-in estimator: bias and variance

Theorem (Bias Expansion)

If f_0 and f_1 are d-times differentiable, the mean of $\widehat{D}_g$ has representation

$$\mathbb{E}[\widehat{D}_g] = D(f_1 \| f_0) + \mathbb{B}(\widehat{D}_g)$$

$$\mathbb{B}(\widehat{D}_g) = \sum_{i=1}^{d} C_i \epsilon^i + O\left(\frac{1}{n\epsilon^d}\right).$$

Locality sensitive hashing plug-in estimator: bias and variance

Theorem (Bias Expansion)

If f_0 and f_1 are d-times differentiable, the mean of $\widehat{D}_g$ has representation

$$\mathbb{E}[\widehat{D}_g] = D(f_1 \| f_0) + \mathbb{B}(\widehat{D}_g)$$

$$\mathbb{B}(\widehat{D}_g) = \sum_{i=1}^{d} C_i \epsilon^i + O\left(\frac{1}{n\epsilon^d}\right).$$

Theorem (Variance)

The variance of the hash-based estimator decreases at least as fast as $1/n$

$$\mathbb{V}(\widehat{D}_g) \leq O\left(\frac{1}{n}\right).$$

Locality sensitive hashing plug-in estimator: bias and variance

Theorem (Bias Expansion)

If f_0 and f_1 are d-times differentiable, the mean of $\widehat{D}_g$ has representation

$$\mathbb{E}[\widehat{D}_g] = D(f_1 \| f_0) + \mathbb{B}(\widehat{D}_g)$$

$$\mathbb{B}(\widehat{D}_g) = \sum_{i=1}^{d} C_i \epsilon^i + O\left(\frac{1}{n\epsilon^d}\right).$$

Theorem (Variance)

The variance of the hash-based estimator decreases at least as fast as $1/n$

$$\mathbb{V}(\widehat{D}_g) \leq O\left(\frac{1}{n}\right).$$

$\Rightarrow$ Choosing $\epsilon = O\left(n^{-1/2d}\right)$ forces bias remainder to $O\left(\frac{1}{n\epsilon^d}\right) = O(1/\sqrt{n})$

Locality sensitive hashing plug-in estimator: bias and variance

Theorem (Bias Expansion)

If f_0 and f_1 are d-times differentiable, the mean of $\widehat{D}_g$ has representation

$$\mathbb{E}[\widehat{D}_g] = D(f_1 \| f_0) + \mathbb{B}(\widehat{D}_g)$$

$$\mathbb{B}(\widehat{D}_g) = \sum_{i=1}^{d} C_i \epsilon^i + O\left(\frac{1}{n\epsilon^d}\right).$$

Theorem (Variance)

The variance of the hash-based estimator decreases at least as fast as $1/n$

$$\mathbb{V}(\widehat{D}_g) \leq O\left(\frac{1}{n}\right).$$

$\Rightarrow$ Choosing $\epsilon = O\left(n^{-1/2d}\right)$ forces bias remainder to $O\left(\frac{1}{n\epsilon^d}\right) = O(1/\sqrt{n})$

$\Rightarrow$ This makes the slowest term in the bias decay as $\mathbb{B}(\widehat{D}_g) = O(n^{-1/2d})$

Ensemble bias reduction: an inverse problem

- Let $\{\widehat{D}_g^{\epsilon(t)}\}_{t\in\mathcal{L}}$ be a set of $L = |\mathcal{L}|$ base learners.
- $\epsilon(t) = tn^{-1/2d}$ is a set of bandwidth parameters.
- $\mathcal{L} := \{t_1, ..., t_L\}$ is a set of scale factors.

Define: Ensemble divergence estimator $\widehat{D}_{\mathbf{w}} := \sum_{j=1}^{L} w_j \widehat{D}_{\epsilon(t_j)} = \mathbf{w}^T \hat{\mathbf{D}}_\epsilon$

Ensemble bias reduction: an inverse problem

- Let $\{\widehat{D}_g^{\epsilon(t)}\}_{t\in\mathcal{L}}$ be a set of $L = |\mathcal{L}|$ base learners.
- $\epsilon(t) = tn^{-1/2d}$ is a set of bandwidth parameters.
- $\mathcal{L} := \{t_1, ..., t_L\}$ is a set of scale factors.

Define: Ensemble divergence estimator $\widehat{D}_{\mathbf{w}} := \sum_{j=1}^{L} w_j \widehat{D}_{\epsilon(t_j)} = \mathbf{w}^T \hat{\mathbf{D}}_\epsilon$

Bias of ensemble divergence estimator:

$$\mathbb{B}\left[\widehat{D}_{\mathbf{w}}\right] = \sum_{i=1}^{d} C_i n^{-i/2d} \sum_{j=1}^{L} w_j t_j^i + O\left(\frac{1}{\sqrt{n}}\right)$$

Ensemble bias reduction: an inverse problem

- Let $\{\widehat{D}_g^{\epsilon(t)}\}_{t\in\mathcal{L}}$ be a set of $L = |\mathcal{L}|$ base learners.
- $\epsilon(t) = tn^{-1/2d}$ is a set of bandwidth parameters.
- $\mathcal{L} := \{t_1, ..., t_L\}$ is a set of scale factors.

Define: Ensemble divergence estimator $\widehat{D}_{\mathbf{w}} := \sum_{j=1}^{L} w_j \widehat{D}_{\epsilon(t_j)} = \mathbf{w}^T \hat{\mathbf{D}}_\epsilon$

Bias of ensemble divergence estimator:

$$\mathbb{B}\left[\widehat{D}_{\mathbf{w}}\right] = \sum_{i=1}^{d} C_i n^{-i/2d} \sum_{j=1}^{L} w_j t_j^i + O\left(\frac{1}{\sqrt{n}}\right)$$

Bias reduced to $O\left(\frac{1}{\sqrt{n}}\right)$ if $\{w_j\}_{j=1}^{L}$ selected to solve linear system $\mathbf{Aw} = \mathbf{0}$:

Ensemble bias reduction: an inverse problem

- Let $\{\widehat{D}_g^{\epsilon(t)}\}_{t\in\mathcal{L}}$ be a set of $L = |\mathcal{L}|$ base learners.
- $\epsilon(t) = tn^{-1/2d}$ is a set of bandwidth parameters.
- $\mathcal{L} := \{t_1, ..., t_L\}$ is a set of scale factors.

Define: Ensemble divergence estimator $\widehat{D}_{\mathbf{w}} := \sum_{j=1}^{L} w_j \widehat{D}_{\epsilon(t_j)} = \mathbf{w}^T \hat{\mathbf{D}}_\epsilon$

Bias of ensemble divergence estimator:

$$\mathbb{B}\left[\widehat{D}_{\mathbf{w}}\right] = \sum_{i=1}^{d} C_i n^{-i/2d} \sum_{j=1}^{L} w_j t_j^i + O\left(\frac{1}{\sqrt{n}}\right)$$

Bias reduced to $O\left(\frac{1}{\sqrt{n}}\right)$ if $\{w_j\}_{j=1}^{L}$ selected to solve linear system $\mathbf{Aw} = \mathbf{0}$:

$$\begin{bmatrix} t_1 & \dots & \dots & t_L \\ t_1^2 & \ddots & \ddots & t_L^2 \\ \vdots & \ddots & \ddots & \vdots \\ t_1^d & \dots & \dots & t_L^d \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ \vdots \\ w_L \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}$$

Ensemble bias reduction: an inverse problem

- Let $\{\widehat{D}_g^{\epsilon(t)}\}_{t\in\mathcal{L}}$ be a set of $L = |\mathcal{L}|$ base learners.
- $\epsilon(t) = tn^{-1/2d}$ is a set of bandwidth parameters.
- $\mathcal{L} := \{t_1, ..., t_L\}$ is a set of scale factors.

Define: Ensemble divergence estimator $\widehat{D}_{\mathbf{w}} := \sum_{j=1}^{L} w_j \widehat{D}_{\epsilon(t_j)} = \mathbf{w}^T \hat{\mathbf{D}}_\epsilon$

Bias of ensemble divergence estimator:

$$\mathbb{B}\left[\widehat{D}_{\mathbf{w}}\right] = \sum_{i=1}^{d} C_i n^{-i/2d} \sum_{j=1}^{L} w_j t_j^i + O\left(\frac{1}{\sqrt{n}}\right)$$

Bias reduced to $O\left(\frac{1}{\sqrt{n}}\right)$ if $\{w_j\}_{j=1}^L$ selected to solve linear system $\mathbf{Aw} = \mathbf{0}$:

$$\begin{bmatrix} t_1 & \dots & \dots & t_L \\ t_1^2 & \ddots & \ddots & t_L^2 \\ \vdots & \ddots & \ddots & \vdots \\ t_1^d & \dots & \dots & t_L^d \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ \vdots \\ w_L \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}$$

$\Rightarrow$ For large d, Chebychev methods used to stabilize solution (Noshad '19)

Ensemble estimator variance needs also to be controled

Variance of ensemble divergence estimator is quadratic in **w**

$$\mathbb{V}(\widehat{D}_{\mathbf{w}}) = \mathbb{V}(\mathbf{w}^T \hat{\mathbf{D}}_{\epsilon}) = \mathbf{w}^T \mathrm{cov}(\hat{\mathbf{D}}_{\epsilon}) \mathbf{w} \leq \|\mathbf{w}\|^2 \lambda_{max}.$$

Ensemble estimator variance needs also to be controled

Variance of ensemble divergence estimator is quadratic in **w**

$$\mathbb{V}(\widehat{D}_{\mathbf{w}}) = \mathbb{V}(\mathbf{w}^T \hat{\mathbf{D}}_\epsilon) = \mathbf{w}^T \mathrm{cov}(\hat{\mathbf{D}}_\epsilon)\mathbf{w} \leq \|\mathbf{w}\|^2 \lambda_{max}.$$

$\Rightarrow$ Select **w** as solution to linearly constrained quadratic program

$$\begin{aligned} \min_{\mathbf{w}} \quad & \|\mathbf{w}\|_2 \\ \textit{subject to} \quad & \sum_{j=1}^{L} w_j = 1, \\ & \sum_{j=1}^{L} w_j t_j^i = 0, i \in [d] \end{aligned}$$

Ensemble estimator variance needs also to be controled

Variance of ensemble divergence estimator is quadratic in $\mathbf{w}$

$$\mathbb{V}(\widehat{D}_{\mathbf{w}}) = \mathbb{V}(\mathbf{w}^T \hat{\mathbf{D}}_\epsilon) = \mathbf{w}^T \mathrm{cov}(\hat{\mathbf{D}}_\epsilon)\mathbf{w} \leq \|\mathbf{w}\|^2 \lambda_{max}.$$

$\Rightarrow$ Select $\mathbf{w}$ as solution to linearly constrained quadratic program

$$\begin{aligned} \min_{\mathbf{w}} \quad & \|\mathbf{w}\|_2 \\ \textit{subject to} \quad & \sum_{j=1}^{L} w_j = 1, \\ & \sum_{j=1}^{L} w_j t_j^i = 0, \, i \in [d] \end{aligned}$$

- If $L > d$, the solution $\mathbf{w}^*$ ensures MSE of $O(1/n)$.
- Weights are computed offline, not dependent on data or data's distribution

Benchmark learner for multiclass classification

Simulation: $K = 4$ classes in concentric sphere regions over $d = 20$ dimensions

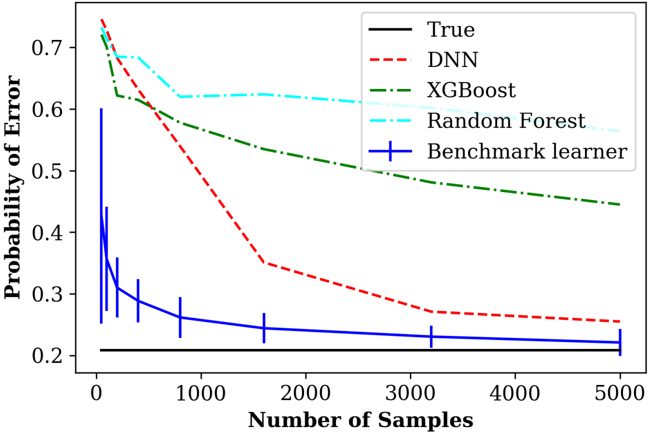
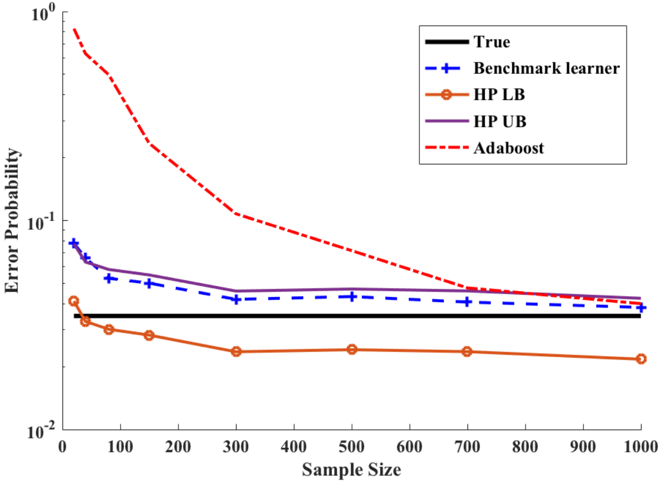


Figure: Benchmark learner indicates small margin for improvement. DNN: 5 hidden layers with [20,64,65,10,40 RELU neurons trained with ADAM and 10% dropout.

Benchmark learner as a minibatch stopping rule

Simulation: classification of 2 mean shifted 10 dim Gaussian densities

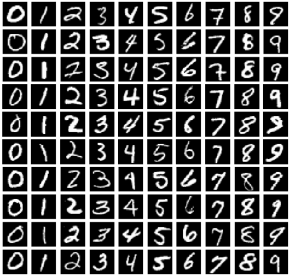


Ref: Noshad and Hero, AISTAT 2018

Benchmarking MNIST digit classification

MNIST handwritten digit corpus:

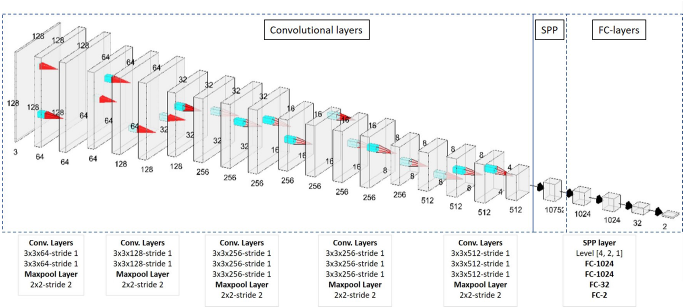
- $K = 10$ classes
- $d = 784$ dimensions
- $n = 60,000$ samples



Papers	Method	Error rate
(Cireşan et al., 2010)	Single 6-layer DNN	0.35%
(Ciresan et al., 2011)	Ensemble of 7 CNNs and training data expansion	0.27%
(Cireşan et al., 2012)	Ensemble of 35 CNNs	0.23%
(Wan et al., 2013)	Ensemble of 5 CNNs and DropConnect regularization	0.21%
Benchmark learner	Ensemble ϵ-ball estimator	0.14%

Table 1: Comparison of error probabilities of several the state of the art deep models with the benchmark learner, for the MNIST handwriting image classification dataset

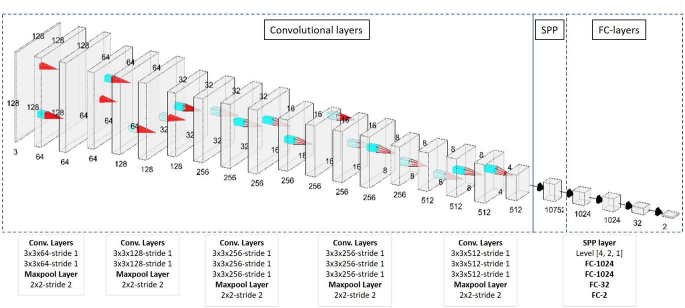
Mutual information estimation: application to DNN information bottleneck



Convolutional neural network (CNN) for image classification[1]

- DNNs have remarkable empirical performance,

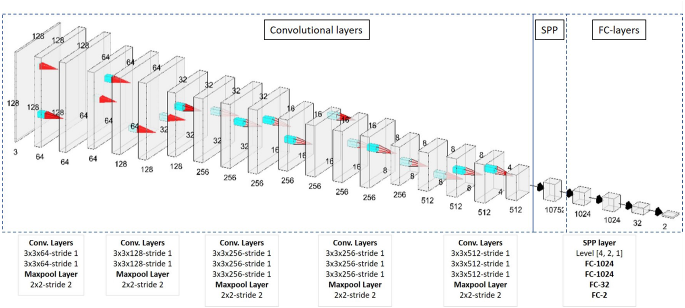
Mutual information estimation: application to DNN information bottleneck



Convolutional neural network (CNN) for image classification[1]

- DNNs have remarkable empirical performance, but there is limited understanding of why DNN perform so well

Mutual information estimation: application to DNN information bottleneck



Convolutional neural network (CNN) for image classification[1]

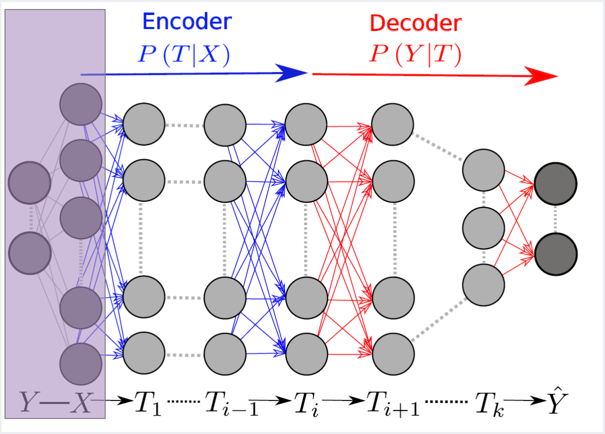
- DNNs have remarkable empirical performance, but there is limited understanding of why DNN perform so well

The compositional learning hypothesis:
DNN's learn in two phases:

- Phase 1: learn the easy cases (**memorize**)
- Phase 2: generalize to the hard cases (**compress**)

[1] B. DuFumier. A new deep learning approach to solar flare prediction. ENSTA internship report, Sept. 2018

Tishby's framework: encoder/decoder information bottleneck

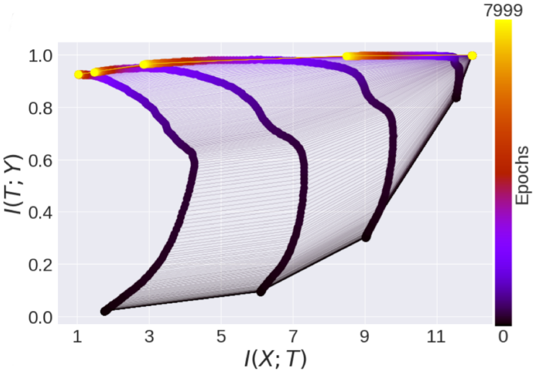


- Encoder I/O: input X, ouptut T (features)
- Decoder I/O: input T, output Y (labels)

[1] R Schwartz-Ziv and N Tishby. "Opening the black box of deep neural networks via information." arXiv 2017

[2] AM Saxe, Y. Bansal, J. Dapello, M. Advani, A. Kolchinsky, BD Tracey, DD. Cox, "On the information bottleneck theory of deep learning," ICLR 2018

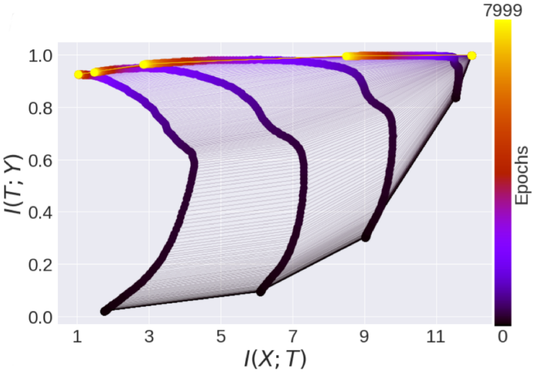
Information plane: a layer-by-layer plot of discrimination vs. compression



- Plot of training-trajectories of $[I(X;T_i), I(T_i;Y)]$ for different layers T_i

$$I(X;T) = \int f_{XT} \log\left(\frac{f_{XT}}{f_X f_T}\right), \quad I(T;Y) = \int f_{TY} \log\left(\frac{f_{TY}}{f_T f_Y}\right)$$

Information plane: a layer-by-layer plot of discrimination vs. compression



- Plot of training-trajectories of $[I(X;T_i), I(T_i;Y)]$ for different layers T_i

$$I(X;T) = \int f_{XT} \log\left(\frac{f_{XT}}{f_X f_T}\right), \quad I(T;Y) = \int f_{TY} \log\left(\frac{f_{TY}}{f_T f_Y}\right)$$

- Schwartz-Ziv&Tishby[1] observed **memorization→compression** for *tanh* activation (MLP 10-8-6-4-2 and classification of 10D Gaussian)

[1] R Schwartz-Ziv and N Tishby. Opening the black box of deep neural networks via information. arXiv 2017

Does memorization→compression depend on activation function?

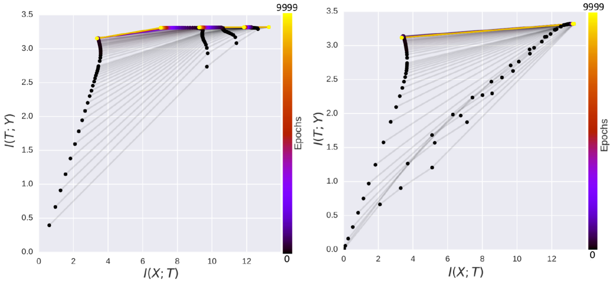


Figure: Figure 1.C (*tanh*) and 1.D (*ReLU*) from Saxe *et al*[1]

- 784-1024-20-20-20-10 MLP trained on MNIST dataset
- Output layer: *sigmoid*. Hidden layers: *tanh* at left and *ReLu* at right.
- Trained using SGD on cross–entropy loss with minibatch size 128
- Learning rate= 0.001 and, at convergence, achieved error rate 0.98

[1] Saxe, Bansal, Dapello, Advani, Kolchinsky, Tracey, and Cox, "On the information bottleneck theory of deep learning," ICLR, 2018.

Does memorization→compression depend on activation function?

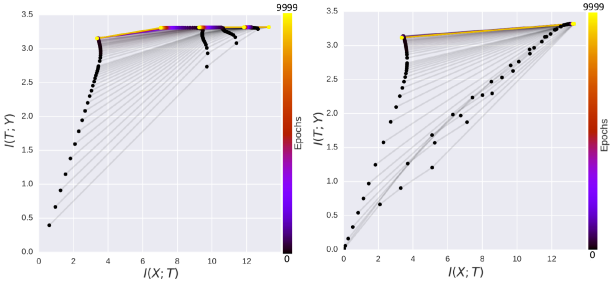


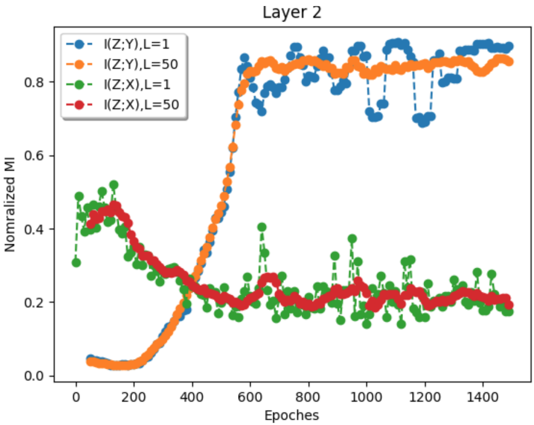
Figure: Figure 1.C (*tanh*) and 1.D (*ReLU*) from Saxe *et al*[1]

- 784-1024-20-20-20-10 MLP trained on MNIST dataset
- Output layer: *sigmoid*. Hidden layers: *tanh* at left and *ReLu* at right.
- Trained using SGD on cross–entropy loss with minibatch size 128
- Learning rate= 0.001 and, at convergence, achieved error rate 0.98

- Saxe *et al* claim that *ReLU* inner layers **exhibit no compression**

[1] Saxe, Bansal, Dapello, Advani, Kolchinsky, Tracey, and Cox, "On the information bottleneck theory of deep learning," ICLR, 2018.

Information plane for MLP/*ReLU* using ensemble MI estimation



- 10-8-6-4-2 MLP/*ReLU* trained on 10,000 samples of 10D Gaussian
- MI with $L = 1$ (green&blue) is the Schwartz-Ziv&Tishby MI estimate
- Proposed ensemble MI implementation[1] (red&orange) is more stable

[1] Noshad, Yu, Hero, "Scalable MI estimation using dependence graphs," ICASSP 2019.

Ensemble estimation provides confirmatory evidence

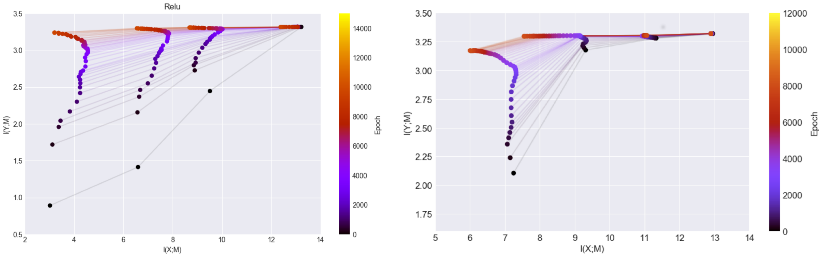


Figure: Left: **MLP/ReLU** 784-1024-20-20-20-10. Right: **CNN/ReLU** 784-4-8-16-10

- MLP and CNN trained on MNIST dataset[1]

⇒ Memorization→Compression phenonomon occurs in both MLP and CNN

[1] Noshad, Yu, Hero, "Scalable MI estimation using dependence graphs," ICASSP 2019.

Summary

Main takeaways

- Learning to benchmark involves 2 types of meta-learning
  - Meta-learning v0: Learning ensembles of weak base-learners (Freund&Schapire 1996)
  - Meta-learning v1: Learning the Bayes error rate (BER)[1][2][3]

[1] Moon, Sricharan, Greenewald, Hero. "Ensemble estimation of information divergence." *Entropy*, 20, no. 8, 2018.

[2] Noshad and Hero, "Scalable hash-based estimation of divergence measures," AISTATS 2018.

[3] Noshad, Zeng, Hero, "Scalable mutual information estimation using dependence graphs," IEEE ICASSP, 2019

Summary

Main takeaways

- Learning to benchmark involves 2 types of meta-learning
    - Meta-learning v0: Learning ensembles of weak base-learners (Freund&Schapire 1996)
    - Meta-learning v1: Learning the Bayes error rate (BER)[1,2,3]
- LSH ensemble method achieves rate optimal performance in both computational complexity and sample complexity

---

[1] Moon, Sricharan, Greenewald, Hero. "Ensemble estimation of information divergence." *Entropy*, 20, no. 8, 2018.

[2] Noshad and Hero, "Scalable hash-based estimation of divergence measures," AISTATS 2018.

[3] Noshad, Zeng, Hero, "Scalable mutual information estimation using dependence graphs," IEEE ICASSP, 2019

Summary

Main takeaways

- Learning to benchmark involves 2 types of meta-learning
  - Meta-learning v0: Learning ensembles of weak base-learners (Freund&Schapire 1996)
  - Meta-learning v1: Learning the Bayes error rate (BER)[1][2][3]
- LSH ensemble method achieves rate optimal performance in both computational complexity and sample complexity
- Benchmark learning applications demonstrated:
  - Performance monitoring: learning sufficient sample size
  - Feature learning: performing data-driven feature selection
  - Interpretable learning: exploring DNN compositional learning hypothesis

[1] Moon, Sricharan, Greenewald, Hero. "Ensemble estimation of information divergence." *Entropy*, 20, no. 8, 2018.

[2] Noshad and Hero, "Scalable hash-based estimation of divergence measures," AISTATS 2018.

[3] Noshad, Zeng, Hero, "Scalable mutual information estimation using dependence graphs," IEEE ICASSP, 2019

Supplement: Chebyshev stabilization of ensemble wieghts ($L = 10$)

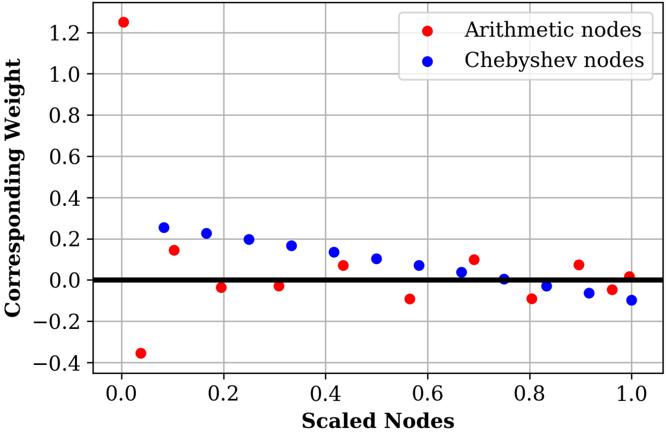


Figure: For $L = 10$ the arithmetic nodes (bandwidth scaled by $k, k+1, \ldots$) give weights with higher dynamic range than the proposed Chebyshev node approach.

Supplement: Chebyshev stabilization of ensemble wieghts ($L = 10$)

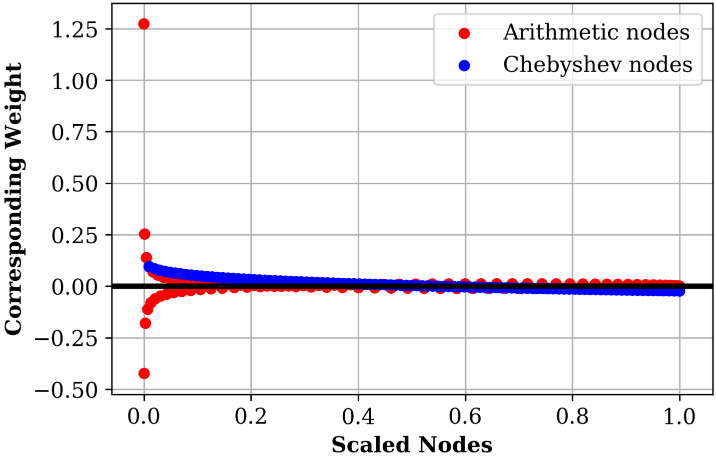


Figure: For $L = 100$ the arithmetic nodes (bandwidth scaled by $k, k+1, \ldots$) give weights with much higher dynamic range than the proposed Chebyshev node approach.

Supplement: Chebyshev wieghts improve MSE of benchmark learner

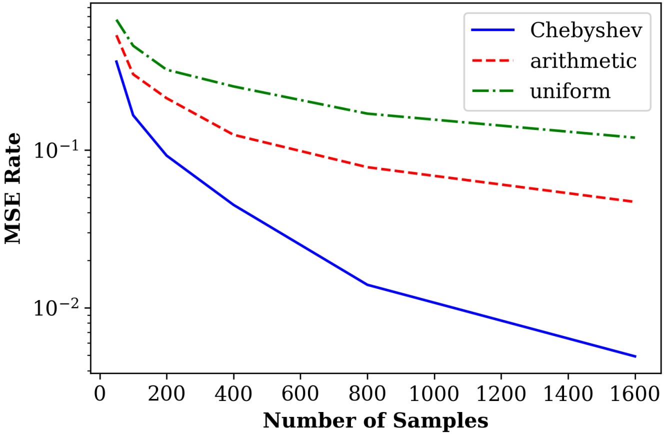


Figure: For a binary classification problem (mean of Gaussian isotropic dsn in dim $d = 100$) the proposed Chebyshev node approach provides significant improvement of MSE in Bayes estimation error rate.

Thanks

Questions?