True Error Bounds for (Stochastic) Neural Networks

John Langford and Rich Caruana

Neural Networks

Very amorphous $\Rightarrow$ very loose bounds

1. New approach: PAC-Bayes (McAllester)

2. Constructing Stochastic Neural Networks

3. Results

Lingo

1. $S = m$ examples drawn iid from D on $X \times Y$

2. hypothesis $= h : X \to Y$

3. "true" error $= e(h) = \mathsf{Pr}_D(h(x) \neq y)$

4. empirical error $= \widehat{e}(h) = \mathsf{Pr}_S(h(x) \neq y)$

PAC-Bayes: $\forall P \sim h$, with probability $1-\delta$:

$$\forall Q: \; E_Q e(h) < E_Q \hat{e}(h) + \sqrt{\frac{\mathrm{KL}(Q||P) + \ln \frac{2m}{\delta}}{2m-1}}$$

1. P = "Prior" distribution over h

2. Q = "Posterior" distribution over h

3. $\mathrm{KL}(Q||P)$ = KL divergence = $E_Q \ln \frac{q(x)}{p(x)}$

Reality:

$$\forall Q: \ \mathsf{KL}(E_Q \hat{e}(h) || E_Q e(h)) < \frac{\mathsf{KL}(Q||P) + \ln \frac{2m}{\delta}}{2m - 1}$$

For neural networks, must:

1. Pick "prior"

2. Pick "posterior"

3. Evaluate $E_Q \widehat{e}(h)$

4. Evaluate bound

“Prior”

1. For every weight, w_i: $N(0, s)$ = gaussian with large variance

2. Reality: pick a mixture of gaussians.

“Posterior”

1. Train Neural Network

2. For each w_i find sensitivity s_i

3. For all w_i, pick $N(w_i, s_i)$

Evaluate $E_Q\hat{e}(h)$

Repeat 500 times:

1. Pick random example

2. Pick random “Posterior” neural network

3. Evaluate

Count errors. Bound $E_Q\hat{e}(h)$

Evaluate Bound

1. Analytically calculate $\mathsf{KL}(Q||P)$.

2. binary search to find $E_Q e(h)$ constraint.

General form

1. Pick “prior” (semigeneric)

2. Pick “posterior” (semigeneric)

3. Evaluate $E_Q \hat{e}(h)$ (generic)

4. Evaluate bound (generic)

Results

1. Learning problem

2. Neural Net architecture

3. Comparison

Learning Problem

1. $X = R^{25}$, $Y = \{-1, 1\}$

2. Pick $m = 100$ examples from D:

   (a) Pick Y uniformly

   (b) Pick $N(y, 1)$

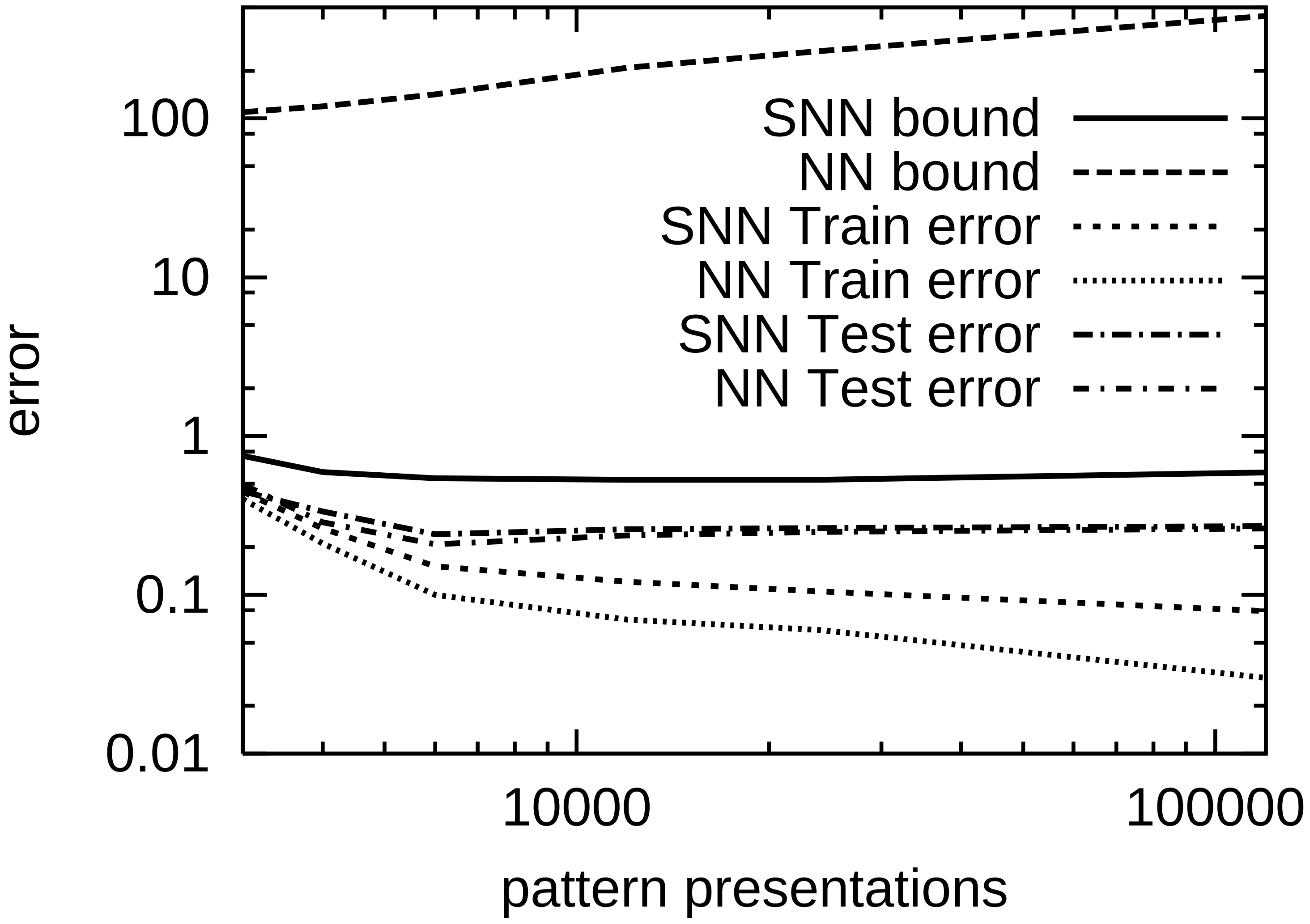
   (c) Pick $N(0, 1)^{24}$

Architecture

- 2 hidden nodes, sigmoid activation
- 1 output node, linear activation

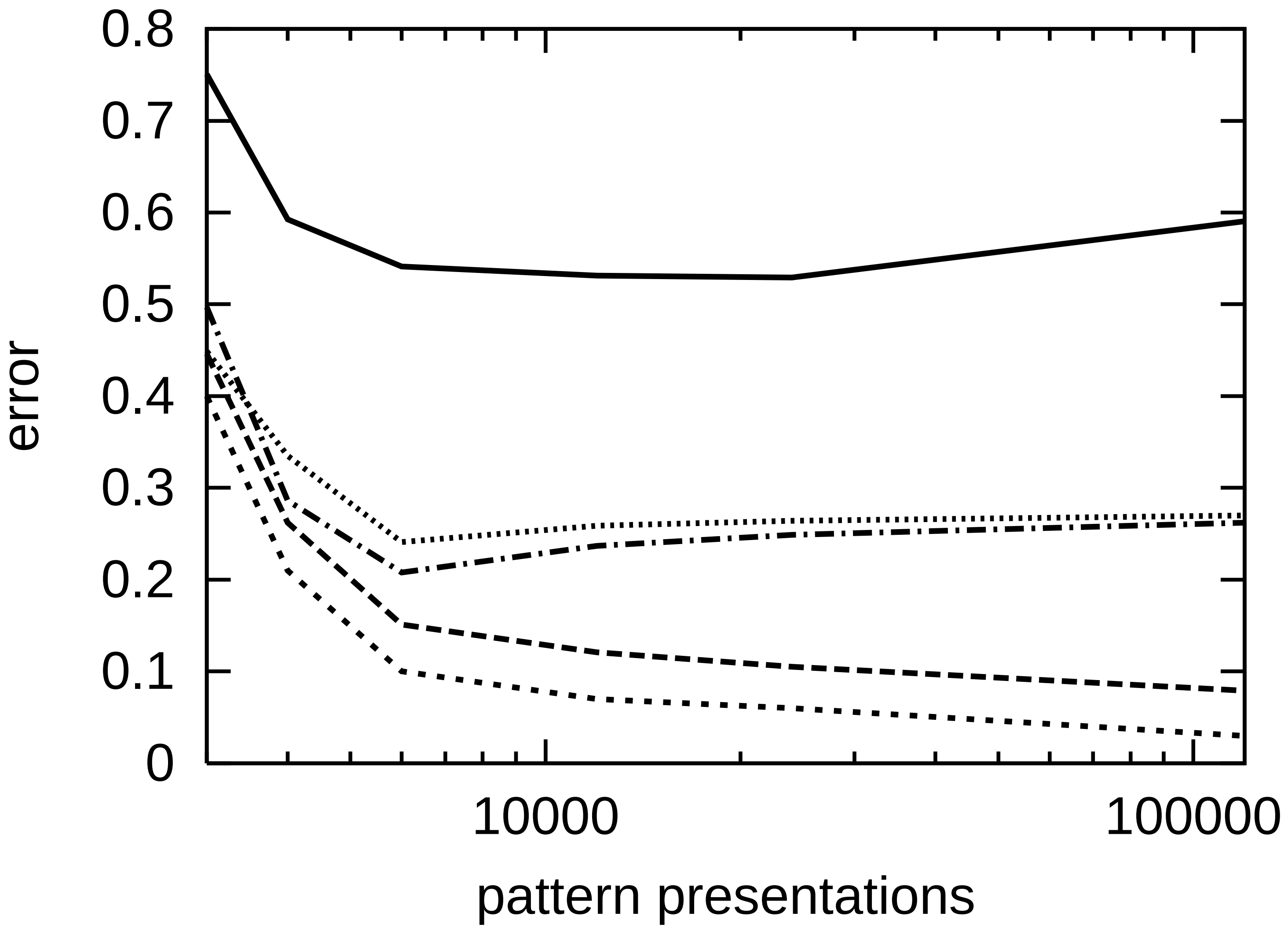
Intentional architecture/problem mismatch

Comparison



SNN bound
NN bound
SNN Train error
NN Train error
SNN Test error
NN Test error
100
10
1
0.1
0.01
error
10000
100000
pattern presentations

Close Comparison



0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
error
10000
100000
pattern presentations

Conclusion

1. Quantitative bounds for (Stochastic) Neural Networks

2. Improvable? (some slack exists)

3. SNN error relatable to NN error?

4. Generalizable? (Matthias Seeger)