Simons Institute Representation Learning Workshop

Semi-Random Units for Learning Neural Networks with Guarantees

Bo Xie
Georgia Tech

joint work with Yingyu Liang,
Kenji Kawaguchi and Le Song

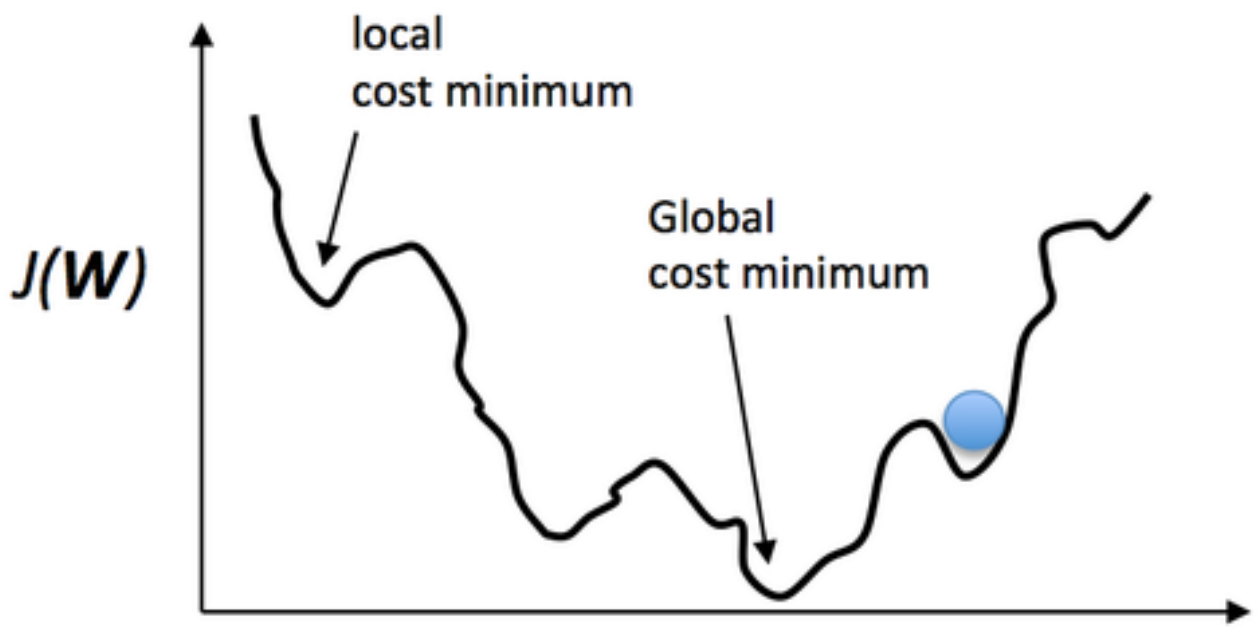
Learning neural networks

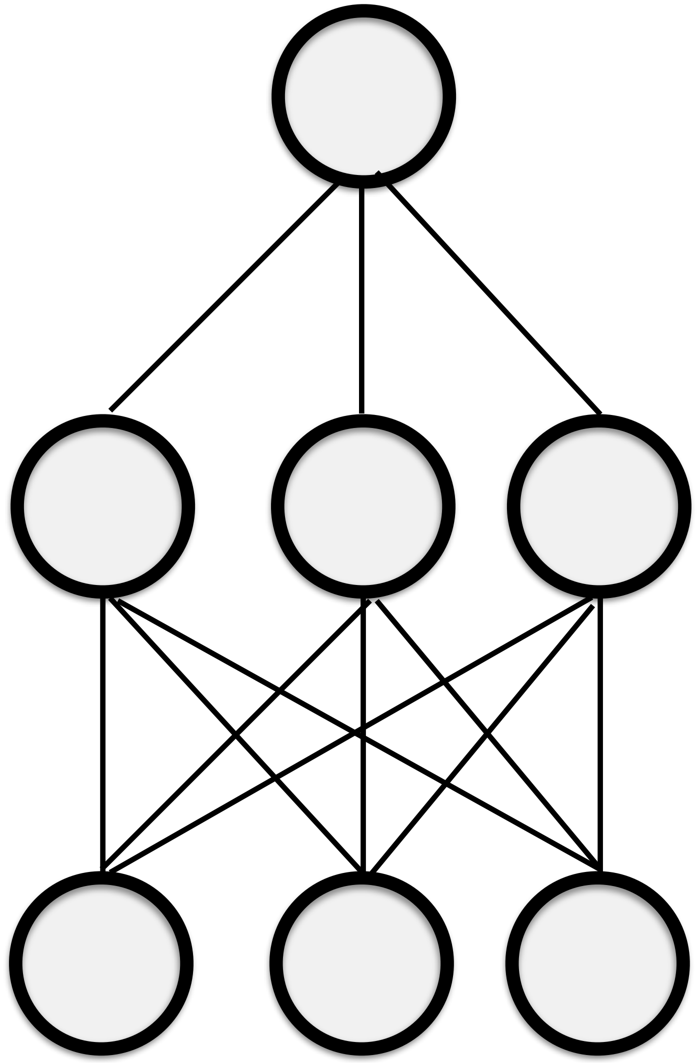
Neural networks are extremely successful in learning many nonlinear functions

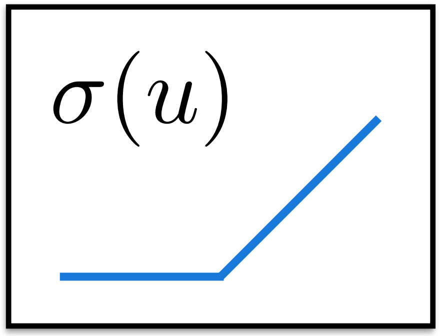
Most are trained with simple Stochastic Gradient Descent (SGD)

Highly non-convex objective function

Why SGD work so well?







Learning neural networks

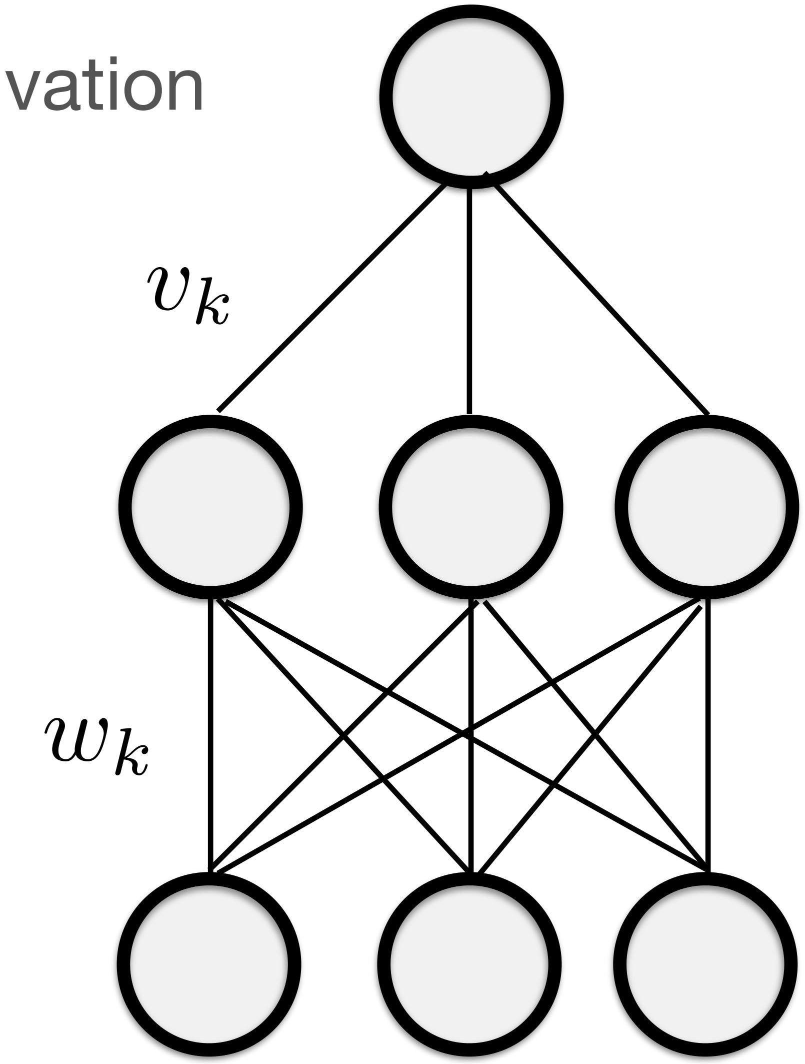
One-hidden-layer neural networks with ReLU activation

$$f(x) = \sum_{k=1}^{n} v_k \sigma(w_k^\top x)$$

Least-squares loss

$$L(f) = \frac{1}{2m} \sum_{l=1}^{m} (y_l - f(x_l))^2$$

Main results:



For "nice" neural weights, with high probability, any stationary point is a global optimum

The structure of the gradient

Gradient w.r.t. first layer weights

$$\frac{\partial L}{\partial w_k} = \frac{1}{m} \sum_{l=1}^{m} \left(f(x_l) - y_l\right) v_k \sigma'(w_k^\top x_l) x_l$$

The structure of the gradient

Gradient w.r.t. first layer weights

$$\frac{\partial L}{\partial w_k} = \frac{1}{m}\sum_{l=1}^{m} \left(f(x_l) - y_l\right) v_k \sigma'(w_k^\top x_l) x_l$$

$$\begin{bmatrix} \frac{\partial L}{\partial w_1} \\ \cdots \\ \frac{\partial L}{\partial w_k} \\ \cdots \\ \frac{\partial L}{\partial w_n} \end{bmatrix} = \begin{pmatrix} v_1\sigma'(w_1^\top x_1)x_1 & \cdots & v_1\sigma'(w_1^\top x_m)x_m \\ \cdots & \cdots & \cdots \\ v_k\sigma'(w_k^\top x_1)x_1 & \cdots & v_k\sigma'(w_k^\top x_m)x_m \\ \cdots & \cdots & \cdots \\ v_n\sigma'(w_n^\top x_1)x_1 & \cdots & v_n\sigma'(w_n^\top x_m)x_m \end{pmatrix} \times \frac{1}{m} \begin{pmatrix} f(x_1) - y_1 \\ \cdots \\ f(x_m) - y_m \end{pmatrix}$$

The structure of the gradient

Gradient w.r.t. first layer weights

$$\frac{\partial L}{\partial w_k} = \frac{1}{m} \sum_{l=1}^{m} (f(x_l) - y_l)\, v_k \sigma'(w_k^\top x_l) x_l$$

$$\begin{bmatrix} \frac{\partial L}{\partial w_1} \\ \cdots \\ \frac{\partial L}{\partial w_k} \\ \cdots \\ \frac{\partial L}{\partial w_n} \end{bmatrix} = \begin{pmatrix} v_1\sigma'(w_1^\top x_1)x_1 & \cdots & v_1\sigma'(w_1^\top x_m)x_m \\ \cdots & \cdots & \cdots \\ v_k\sigma'(w_k^\top x_1)x_1 & \cdots & v_k\sigma'(w_k^\top x_m)x_m \\ \cdots & \cdots & \cdots \\ v_n\sigma'(w_n^\top x_1)x_1 & \cdots & v_n\sigma'(w_n^\top x_m)x_m \end{pmatrix} \times \frac{1}{m} \begin{pmatrix} f(x_1) - y_1 \\ \cdots \\ f(x_m) - y_m \end{pmatrix}$$

$$\frac{\partial L}{\partial W} = D\ r$$

The structure of the gradient

Gradient w.r.t. first layer weights

$$\frac{\partial L}{\partial w_k} = \frac{1}{m} \sum_{l=1}^{m} (f(x_l) - y_l)\, v_k \sigma'(w_k^\top x_l) x_l$$

$$\begin{bmatrix} \frac{\partial L}{\partial w_1} \\ \cdots \\ \frac{\partial L}{\partial w_k} \\ \cdots \\ \frac{\partial L}{\partial w_n} \end{bmatrix} = \begin{pmatrix} v_1\sigma'(w_1^\top x_1)x_1 & \cdots & v_1\sigma'(w_1^\top x_m)x_m \\ \cdots & \cdots & \cdots \\ v_k\sigma'(w_k^\top x_1)x_1 & \cdots & v_k\sigma'(w_k^\top x_m)x_m \\ \cdots & \cdots & \cdots \\ v_n\sigma'(w_n^\top x_1)x_1 & \cdots & v_n\sigma'(w_n^\top x_m)x_m \end{pmatrix} \times \frac{1}{m} \begin{pmatrix} f(x_1) - y_1 \\ \cdots \\ f(x_m) - y_m \end{pmatrix}$$

$$\frac{\partial L}{\partial W} = D r$$

non-singular?

The intuition

Key inequality

$$\|r\| \leq \frac{1}{s_m(D)} \left\| \frac{\partial L}{\partial W} \right\|$$

training error

minimum singular value

norm of gradient

The intuition

Key inequality

$$\|r\| \leq \frac{1}{s_m(D)} \left\| \frac{\partial L}{\partial W} \right\|$$

Need to lower bound minimum singular value

Bounding the error

Key inequality

$$\|r\| \leq \frac{1}{s_m(D)} \left\| \frac{\partial L}{\partial W} \right\|$$

Need to lower bound minimum singular value

Directly analyze the singular value

$$G_n = D^\top D / n$$

it is a function of the weights;
difficult to analyze

Bounding the error

Key inequality

$$\|r\| \leq \frac{1}{s_m(D)} \left\| \frac{\partial L}{\partial W} \right\|$$

Need to lower bound minimum singular value

Directly analyze the singular value

$$G_n = D^\top D/n \qquad\qquad G = \mathbb{E}_w[G_n]$$

introduce an intermediate variable
that has uniform weights

Bounding the error

Key inequality

$$\|r\| \leq \frac{1}{s_m(D)} \left\| \frac{\partial L}{\partial W} \right\|$$

Need to lower bound minimum singular value

Directly analyze the singular value

$$G_n = D^\top D / n \qquad\qquad G = \mathbb{E}_w[G_n]$$

Decompose into two parts

$$\lambda_m(G_n) \geq \underbrace{\lambda_m(G)}_{\text{I. ideal spectrum}} - \underbrace{\|G - G_n\|}_{\text{II. discrepancy}}$$

Bounding the first term

Kernel function associated with ReLU

$$
\begin{aligned}
G_{ij} &= \mathbb{E}_w\left[\sigma'(w^\top x_i)\sigma'(w^\top x_j)\right]\langle x_i, x_j\rangle \\
&= \left(\frac{1}{2} - \frac{\arccos\langle x_i, x_j\rangle}{2\pi}\right)\langle x_i, x_j\rangle \\
&= \sum_{u=1}^{\infty} \gamma_u \phi_u(x_i)\phi_u(x_j)
\end{aligned}
$$

spherical harmonics decomposition

Bounding the first term

Kernel function associated with ReLU

$$G_{ij} = \mathbb{E}_w \left[\sigma'(w^\top x_i)\sigma'(w^\top x_j)\right] \langle x_i, x_j \rangle$$
$$= \left(\frac{1}{2} - \frac{\arccos \langle x_i, x_j \rangle}{2\pi}\right) \langle x_i, x_j \rangle$$
$$= \sum_{u=1}^{\infty} \gamma_u \phi_u(x_i) \phi_u(x_j)$$

With high probability

$$\boxed{\lambda_m(G) \geq m\gamma_m/2}$$

The spectrum of ReLU in between $O(1/m)$ and $O(1/\sqrt{m})$

Bounding the second term

The difference between true weights and the expected one

$$\|G - G_n\| \leq O(\rho(L_2(W))$$

Bounding the second term

The difference between true weights and the expected one

$$\|G - G_n\| \leq O(\rho(L_2(W))$$

Weight discrepancy

Difference of expected and actual weights

$$(L_2(W))^2 = \frac{1}{n^2} \sum_{i,j=1}^{n} k(w_i, w_j)^2 - \mathbb{E}_{u,v}\left[k(u,v)^2\right]$$

where

$$k(x, y) = \frac{1}{2} - \frac{\arccos\langle x, y\rangle}{2\pi}$$

A bound on the minimum singular value

With high probability

$$s_m(D)^2 \geq nm\gamma_m/2 - cn\rho(L_2(W))$$

A simplified result

With high probability

$$s_m(D)^2 \geq nm\gamma_m/2 - cn\rho(L_2(W))$$

Suppose n and d are large enough and weight discrepancy is small

$$n = \tilde{\Omega}(1/\gamma_m) \qquad d = \tilde{\Omega}(1/\gamma_m) \qquad L_2(W) = \tilde{O}(n^{-1/4}d^{-1/4})$$

Then with high probability

$$s_m(D)^2 \geq \Omega(m)$$

Final error

For n and d large enough

For any W that has small weight discrepancy

With high probability

$$\frac{1}{2m}\sum_{l=1}^{m}\left(f(x_l) - y_l\right)^2 \leq O\left(\left\|\frac{\partial L}{\partial W}\right\|^2\right)$$

Final error

For n and d large enough

For any W that has small weight discrepancy

With high probability

$$\frac{1}{2m}\sum_{l=1}^{m}(f(x_l)-y_l)^2 \leq O\left(\left\|\frac{\partial L}{\partial W}\right\|^2\right)$$

small gradient means small error!

Final error

For n and d large enough

For any W that has small weight discrepancy

With high probability

$$\frac{1}{2m}\sum_{l=1}^{m}\left(f\left(x_l\right)-y_l\right)^2 \leq O\left(\left\|\frac{\partial L}{\partial W}\right\|^2\right)$$

n and d are between $O(\sqrt{m})$ and $O(m)$

Most W satisfy weight discrepancy small enough

Recap

Analyzed optimization landscape of one-hidden layer network

Technical difficulty on ensuring small weight discrepancy

Next: semi-random units

Semi-random units

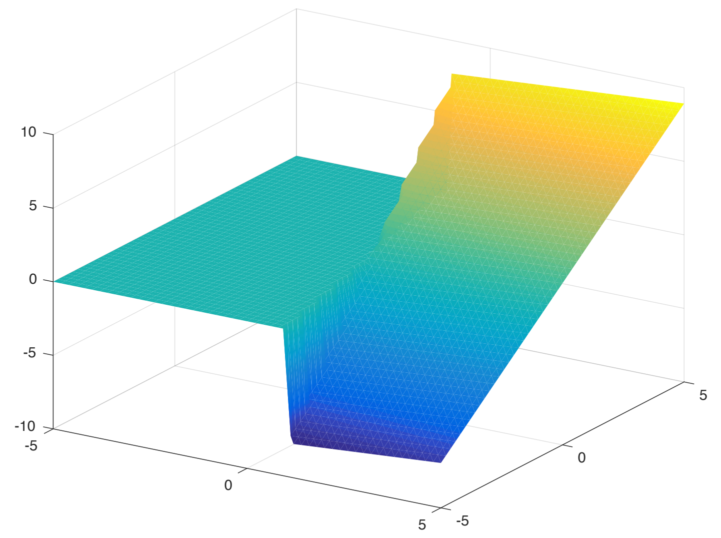
The main technical difficulty comes from the nonlinearity part

Decouple ReLU: semi-random units

$$\sigma(w^\top x) = \mathbb{I}\left[w^\top x > 0\right] w^\top x$$

replace by random projections!

$$\sigma(w^\top x) = \mathbb{I}\left[r^\top x > 0\right] w^\top x$$

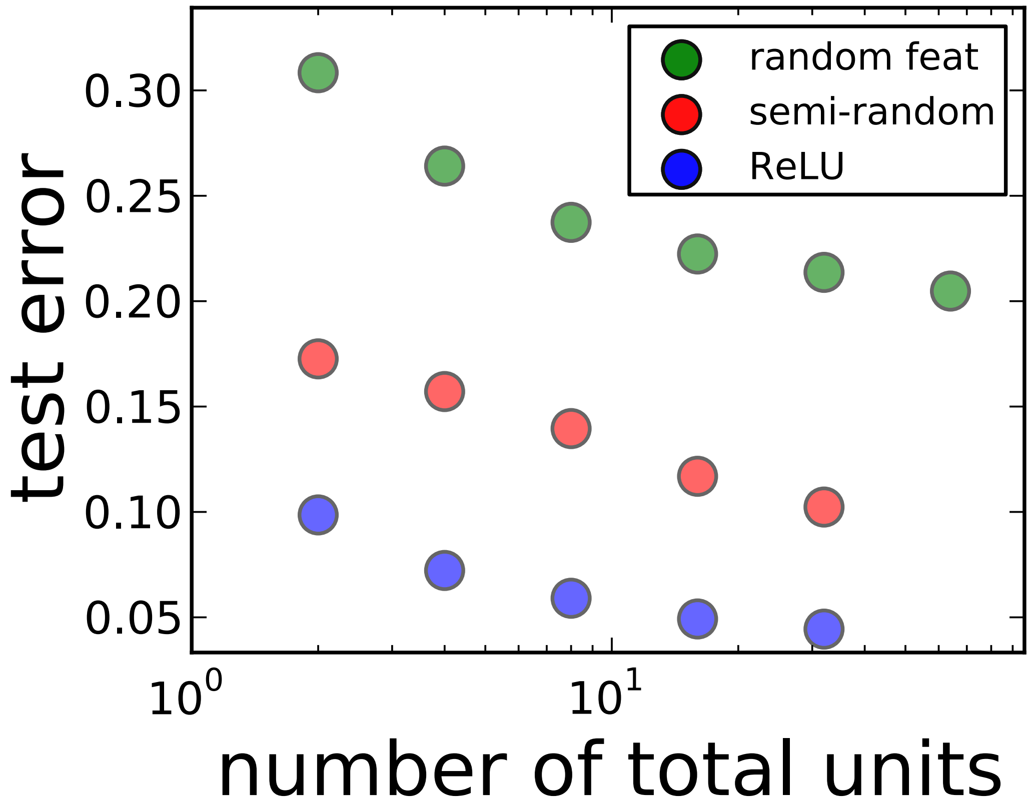


Semi-random units

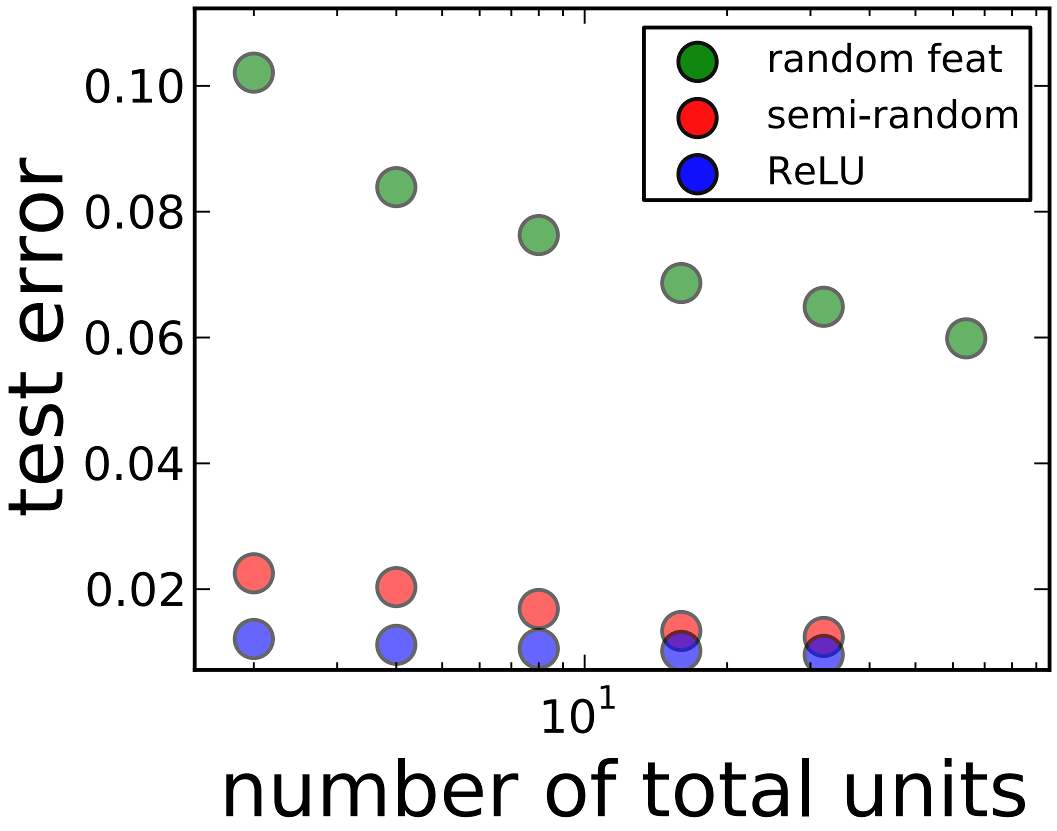
Properties of semi-random units

- It sits between fully-random features and fully-adjustable units
- Linear in the parameters, but nonlinear in the input
- Guaranteed to converge to global optimum w.h.p.
- Has universal approximation ability

Matching the performance of ReLU



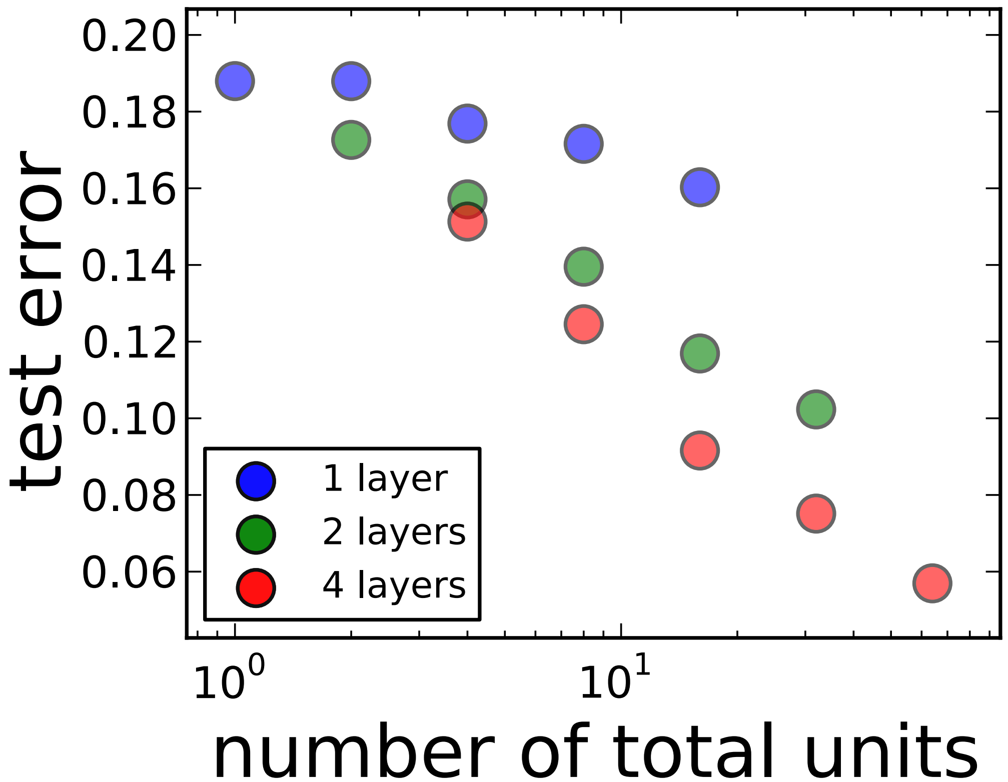
Covtype dataset



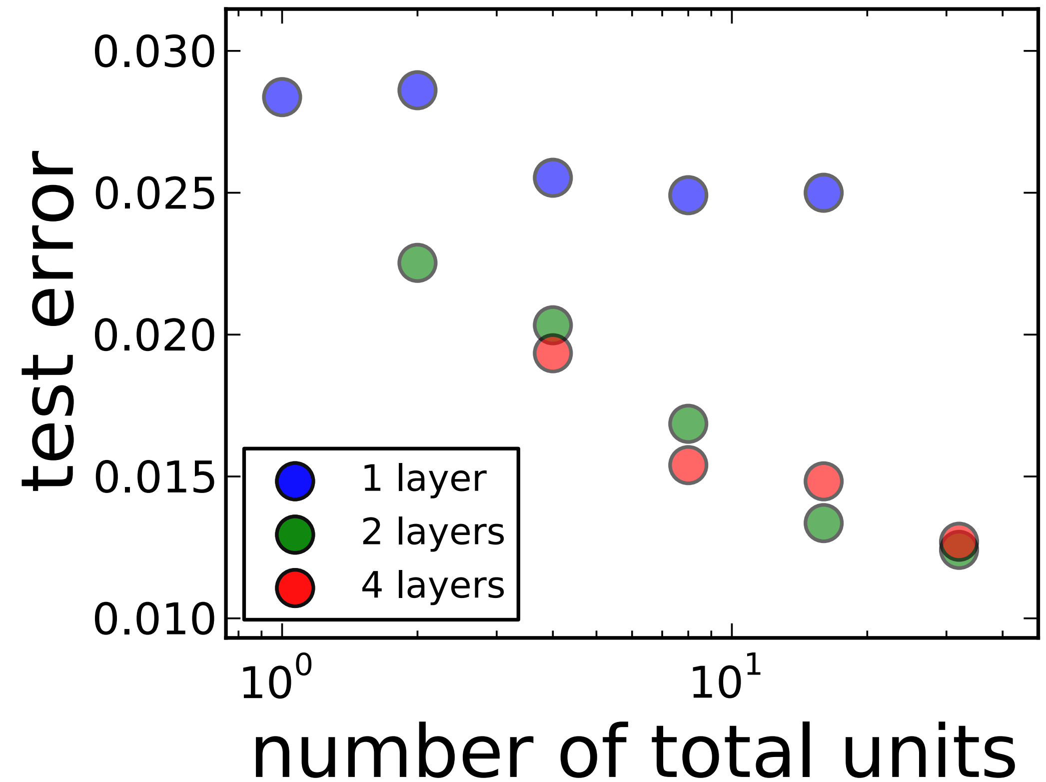
Webspam dataset

Experiment results

Width vs depth; depth helps more



Covtype dataset



Webspam dataset

Experiment results

Image classification benchmarks

neuron type	MNIST	CIFAR10	SVHN
ReLU	0.70	16.3	3.9
RF	8.80	59.2	73.9
RF 2×	5.71	55.8	70.5
RF 4×	4.10	49.8	58.4
RF 16×	2.69	40.7	37.1
SR	0.97	21.4	7.6
SR 2×	0.78	17.4	6.9
SR 4×	0.71	18.7	6.4

Conclusion

For one-hidden-layer neural network, under weight diversity condition, any critical points are w.h.p. global optimal

The result depends on the spectrum decay of the kernel associated with the activation function

Propose semi-random units and networks with these units are guaranteed to converge to global optimal

Matching the performance of ReLU with slightly more units but much better than random features