Deep Networks

Deep compositions of nonlinear functions

\[ h = h_m \circ h_{m-1} \circ \cdots \circ h_1 \]

e.g.,

\[ h_i : x \mapsto \sigma(W_i x) \]

\[ \sigma(v)_i = \frac{1}{1 + \exp(-v_i)}, \]

\[ r(v)_i = \max\{0, v_i\} \]
Deep Networks

Representation learning
Depth provides an effective way of representing useful features.

Rich non-parametric family
Depth provides parsimonious representations. Nonlinear parameterizations provide better rates of approximation. Some functions require much more complexity for a shallow representation.

But...
- Optimization?
  - Nonlinear parameterization.
  - Apparently worse as the depth increases.
- Generalization?
  - What determines the statistical complexity of a deep network?
Outline

- Deep residual networks
  - Representing with near-identities
  - Global optimality of stationary points
- Optimization in deep linear residual networks
  - Gradient descent
  - Symmetric maps and positivity
  - Regularized gradient descent and positive maps
- Statistical complexity of deep networks
  - VC theory: Number of parameters
  - Margins analysis: Size of parameters
  - Understanding generalization failures
Outline

- **Deep residual networks**
  - Representing with near-identities
  - Global optimality of stationary points

- **Optimization in deep linear residual networks**
  - Gradient descent
  - Symmetric maps and positivity
  - Regularized gradient descent and positive maps

- **Statistical complexity of deep networks**
  - VC theory: Number of parameters
  - Margins analysis: Size of parameters
  - Understanding generalization failures
Deeper Networks

Revolution of Depth

ImageNet Classification top-5 error (%)

(ILSVRC'15 ResNet, 3.57) → (ILSVRC'14 GoogleNet, 6.7) → (ILSVRC'14 VGG, 7.3) → (ILSVRC'13, 11.7) → (ILSVRC'12 AlexNet, 16.4) → (shallow, 25.8) → (ILSVRC'11, 28.2) → (ILSVRC'10)

(Deep Residual Networks. Kaiming He. 2016)
Deeper Networks

Revolution of Depth

AlexNet, 8 layers (ILSVRC 2012)

11x11 conv, 96, /4, pool/2
5x5 conv, 256, pool/2
3x3 conv, 384
3x3 conv, 384
3x3 conv, 256, pool/2
fc, 4096
fc, 4096
fc, 1000

(Deep Residual Networks. Kaiming He. 2016)
Deeper Networks

Revolution of Depth

AlexNet, 8 layers (ILSVRC 2012)

VGG, 19 layers (ILSVRC 2014)

GoogleNet, 22 layers (ILSVRC 2014)

(Deep Residual Networks. Kaiming He. 2016)
Revolution of Depth

- AlexNet, 8 layers (ILSVRC 2012)
- VGG, 19 layers (ILSVRC 2014)
- ResNet, 152 layers (ILSVRC 2015)

(Deep Residual Networks. Kaiming He. 2016)
Deep Residual Networks

Deep network component:

\[ H(x) = \text{relu} \left( \text{weight layer} \left( \text{relu} \left( \text{weight layer} (x) \right) \right) \right) \]

Residual network component:

\[ H(x) = F(x) + x \]

(Deep Residual Networks. Kaiming He. 2016)
Deep Residual Networks

Advantages

- With zero weights, the network computes the identity.
- Identity connections provide useful feedback throughout the network.

(Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. 2016)
Deep Residual Networks

Training deep plain nets vs deep residual nets: CIFAR-10

(Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. 2016)

Large improvements over plain nets (e.g., ImageNet Large Scale Visual Recognition Challenge, Common Objects in Context Detection Challenge).
Some intuition: linear functions

Products of near-identity matrices

Every invertible* \( A \) can be written as

\[
A = (I + A_m) \cdots (I + A_1),
\]

where \( \|A_i\| = O(1/m) \).

(Hardt and Ma, 2016)

* Provided \( \det(A) > 0 \).
Some intuition: linear functions

Products of near-identity matrices

For a linear Gaussian model,

\[ y = Ax + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2 I), \]

consider choosing \( A_1, \ldots, A_m \) to minimize quadratic loss:

\[ \mathbb{E}\| (I + A_m) \cdots (I + A_1)x - y \|^2. \]

If \( \|A_i\| < 1 \), every stationary point of the quadratic loss is a global optimum:

\[ \forall i, \quad \nabla A_i \mathbb{E}\| (I + A_m) \cdots (I + A_1)x - y \|^2 = 0 \]

\[ \Rightarrow \quad A = (I + A_m) \cdots (I + A_1). \]

(Hardt and Ma, 2016)
Outline

- Deep residual networks
  - Representing with near-identities
  - Global optimality of stationary points
- Optimization in deep linear residual networks
- Statistical complexity of deep networks

Steve Evans
Berkeley, Stat/Math

Phil Long
Google

arXiv:1804.05012
Representing with near-identities

Result

The computation of a smooth invertible map $h$ can be spread throughout a deep network,

$$h_m \circ h_{m-1} \circ \cdots \circ h_1 = h,$$

so that all layers compute near-identity functions:

$$\|h_i - \text{Id}\|_L = O\left(\frac{\log m}{m}\right).$$

Definition: the Lipschitz seminorm of $f$ satisfies, for all $x, y$,

$$\|f(x) - f(y)\| \leq \|f\|_L \|x - y\|.$$

Think of the functions $h_i$ as near-identity maps that might be computed as

$$h_i(x) = x + A\sigma(Bx).$$
Representing with near-identities

Theorem

Consider a function \( h : \mathbb{R}^d \to \mathbb{R}^d \) on a bounded domain \( \mathcal{X} \subset \mathbb{R}^d \). Suppose that \( h \) is

1. Differentiable,
2. Invertible,
3. Smooth: For some \( \alpha > 0 \) and all \( x, y, u \), \( \| Dh(y) - Dh(x) \| \leq \alpha \| y - x \| \).
4. Lipschitz inverse: For some \( M > 0 \), \( \| h^{-1} \|_L \leq M \).
5. Positive orientation: For some \( x_0 \), \( \det(Dh(x_0)) > 0 \).

Then for all \( m \), there are \( m \) functions \( h_1, \ldots, h_m : \mathbb{R}^d \to \mathbb{R}^d \) satisfying \( \| h_i - \text{Id} \|_L = O(\log m/m) \) and \( h_m \circ h_{m-1} \circ \cdots \circ h_1 = h \) on \( \mathcal{X} \).

- \( Dh \) is the derivative; \( \| Dh(y) \| \) is the induced norm:
  \[
  \| f \| := \sup \left\{ \frac{\| f(x) \|}{\| x \|} : \| x \| > 0 \right\}.
  \]
Representing with near-identities

### Key ideas

1. Assume $h(0) = 0$ and $Dh(0) = \text{Id}$ (else shift and linearly transform).

2. Construct the $h_i$ so that

   $$h_1(x) = \frac{h(a_1 x)}{a_1}$$

   $$h_2(h_1(x)) = \frac{h(a_2 x)}{a_2}$$

   $$\vdots$$

   $$h_m(\cdots(h_1(x))\cdots) = \frac{h(a_m x)}{a_m},$$

3. Pick $a_m = 1$ so $h_m \circ \cdots \circ h_1 = h$.

4. Ensure that $a_1$ is small enough that $h_1 \approx Dh(0) = \text{Id}$.

5. Ensure that $a_i$ and $a_{i+1}$ are sufficiently close that $h_i \approx \text{Id}$.

6. Show $\|h_i - \text{Id}\|_L$ is small on small and large scales (c.f. $a_i - a_{i-1}$).
The computation of a smooth invertible map $h$ can be spread throughout a deep network,

$$h_m \circ h_{m-1} \circ \cdots \circ h_1 = h,$$

so that all layers compute near-identity functions:

$$\|h_i - \text{Id}\|_L = O\left(\frac{\log m}{m}\right).$$

- Deeper networks allow flatter nonlinear functions at each layer.
Deep residual networks
- Representing with near-identities
- **Global optimality of stationary points**

Optimization in deep linear residual networks

Statistical complexity of deep networks
Stationary points

Result

For \((X, Y)\) with an arbitrary joint distribution, define the squared error,

\[
Q(h) = \frac{1}{2} \mathbb{E} \| h(X) - Y \|^2_2,
\]

define the minimizer \(h^*(x) = \mathbb{E}[Y|X = x]\).

Consider a function \(h = h_m \circ \cdots \circ h_1\), where \(\|h_i - \text{Id}\|_L \leq \epsilon < 1\). Then for all \(i\),

\[
\|D_{h_i} Q(h)\| \geq \frac{(1 - \epsilon)^{m-1}}{\| h - h^* \|} (Q(h) - Q(h^*)).
\]

- e.g., if \((X, Y)\) is uniform on a training sample, then \(Q\) is empirical risk and \(h^*\) an empirical risk minimizer.
- \(D_{h_i} Q\) is a Fréchet derivative; \(\|h\|\) is the induced norm.
Stationary points

What the theorem says

- If the composition $h$ is sub-optimal and each function $h_i$ is a near-identity, then there is a downhill direction in function space: the functional gradient of $Q$ wrt $h_i$ is non-zero.

- Thus every stationary point is a global optimum.

- There are no local minima and no saddle points.
What the theorem says

- The theorem does not say there are no local minima of a deep residual network of ReLUs or sigmoids with a fixed architecture.
- Except at the global minimum, there is a downhill direction in function space. But this direction might be orthogonal to functions that can be computed with this fixed architecture.
- We should expect suboptimal stationary points in the ReLU or sigmoid parameter space, but these cannot arise because of interactions between parameters in different layers; they arise only within a layer.
For \((X, Y)\) with an arbitrary joint distribution, define the squared error,

\[
Q(h) = \frac{1}{2} \mathbb{E} \| h(X) - Y \|_2^2,
\]

define the minimizer \(h^*(x) = \mathbb{E}[Y|X = x]\).

Consider a function \(h = h_m \circ \cdots \circ h_1\), where \(\|h_i - \text{Id}\|_L \leq \epsilon < 1\).

Then for all \(i\),

\[
\| D_{h_i} Q(h) \| \geq \frac{(1 - \epsilon)^{m-1}}{\| h - h^* \|} \left( Q(h) - Q(h^*) \right).
\]

- e.g., if \((X, Y)\) is uniform on a training sample, then \(Q\) is empirical risk and \(h^*\) an empirical risk minimizer.
- \(D_{h_i} Q\) is a Fréchet derivative; \(\| h \|\) is the induced norm.
Stationary points

Proof ideas (1)

If $\|f - \text{Id}\|_L \leq \alpha < 1$ then

1. $f$ is invertible.

2. $\|f\|_L \leq 1 + \alpha$ and $\|f^{-1}\|_L \leq 1/(1 - \alpha)$.

3. For $F(g) = f \circ g$, $\|DF(g) - \text{Id}\| \leq \alpha$.

4. For a linear map $h$ (such as $DF(g) - \text{Id}$), $\|h\| = \|h\|_L$.

- $\|f\|$ denotes the induced norm: $\|g\| := \sup \left\{ \frac{\|g(x)\|}{\|x\|} : \|x\| > 0 \right\}$.
Stationary points

Proof ideas (2)

1. Projection theorem implies

\[ Q(h) = \frac{1}{2} \mathbb{E} \left\| h(X) - h^*(X) \right\|_2^2 + \text{constant}. \]

2. Then

\[ D_{h_i} Q(h) = \mathbb{E} \left[ (h(X) - h^*(X)) \cdot \text{ev}_X \circ D_{h_i} h \right]. \]

3. It is possible to choose a direction \( \Delta \) s.t. \( \|\Delta\| = 1 \) and

\[ D_{h_i} Q(h)(\Delta) = c \mathbb{E} \left\| h(X) - h^*(X) \right\|_2^2. \]

4. Because the \( h_j \)s are near-identities,

\[ c \geq \frac{(1 - \epsilon)^{m-1}}{\| h - h^* \|}. \]

• \( \text{ev}_x \) is the evaluation functional, \( \text{ev}_x(f) = f(x) \).
Stationary points

Result

For \((X, Y)\) with an arbitrary joint distribution, define the squared error,

\[
Q(h) = \frac{1}{2} \mathbb{E} \| h(X) - Y \|_2^2,
\]

define the minimizer \(h^*(x) = \mathbb{E}[Y|X=x]\).

Consider a function \(h = h_m \circ \cdots \circ h_1\), where \(\|h_i - \text{Id}\|_L \leq \epsilon < 1\).

Then for all \(i\),

\[
\| D_{h_i} Q(h) \| \geq \frac{(1 - \epsilon)^{m-1}}{\| h - h^* \|} (Q(h) - Q(h^*)).
\]

- e.g., if \((X, Y)\) is uniform on a training sample, then \(Q\) is empirical risk and \(h^*\) an empirical risk minimizer.
- \(D_{h_i} Q\) is a Fréchet derivative; \(\| h \|\) is the induced norm.
Deep compositions of near-identities

Questions

- If the mapping is not invertible? e.g., $h : \mathbb{R}^d \rightarrow \mathbb{R}$?
  - If $h$ can be extended to a bi-Lipschitz mapping to $\mathbb{R}^d$, it can be represented with flat functions at each layer. What if it cannot?

- Implications for optimization?
  - Related to Polyak-Łojasiewicz function classes; proximal algorithms for these classes converge quickly to stationary points.

- Regularized gradient methods for near-identity maps?
Outline

- Deep residual networks
- **Optimization in deep linear residual networks**
  - Gradient descent
  - Symmetric maps and positivity
  - Regularized gradient descent and positive maps
- Statistical complexity of deep networks

Dave Helmbold
UCSC

Phil Long
Google

arXiv:1802.06093
Optimization in deep linear residual networks

Linear networks

- Consider $f_\Theta : \mathbb{R}^d \to \mathbb{R}^d$ defined by $f_\Theta(x) = \Theta_L \cdots \Theta_1 x$.
- Suppose $(x, y) \sim P$, and consider using gradient methods to choose $\Theta$ to minimize $\ell(\Theta) = \frac{1}{2} \mathbb{E} \| f_\Theta(x) - y \|^2$.

Assumptions

1. $\mathbb{E} x x^\top = I$
2. $y = \Phi x$ for some matrix $\Phi$ (wlog, because of projection theorem)
Define $\Phi$ as the minimizer of $\mathbb{E}\|\Phi x - y\|^2$ (the least squares map). Then the projection theorem implies

$$
\mathbb{E}\|\Theta x - y\|^2 = \mathbb{E}\|\Theta x - \Phi x\|^2 + 2\mathbb{E}(\Theta x - \Phi x)^\top (\Phi x - y) + \mathbb{E}\|\Phi x - y\|^2
$$

$$
= \mathbb{E}\|\Theta x - \Phi x\|^2 + \mathbb{E}\|\Phi x - y\|^2,
$$

so wlog we can assume $y = \Phi x$ and define, for linear $f_\Theta$,

$$
\ell(\Theta) = \frac{1}{2}\mathbb{E}\|f_\Theta(x) - \Phi x\|^2.
$$
Optimization in deep linear residual networks

Recall $f_{\Theta}(x) = \Theta_L \cdots \Theta_1 x = \Theta_{1:L} x$, where we use the notation $\Theta_{i:j} = \Theta_j \Theta_{j-1} \cdots \Theta_i$.

**Gradient descent**

$$
\Theta^{(0)} = \left( \Theta_1^{(0)}, \Theta_2^{(0)}, \ldots, \Theta_L^{(0)} \right) := (I, I, \ldots, I)
$$

$$
\Theta_i^{(t+1)} := \Theta_i^{(t)} - \eta (\Theta_{i+1:L})^\top \left( \Theta_{1:L}^{(t)} - \Phi \right) (\Theta_{1:i-1}^{(t)})^\top,
$$

where $\eta$ is a step-size.
Gradient descent in deep linear residual networks

Theorem

There is a positive constant $c_0$ and polynomials $p_1$ and $p_2$ such that if $\ell(\Theta^{(0)}) \leq c_0$ and $\eta \leq 1/p_1(d, L)$, after $p_2(d, L, 1/\eta) \log(1/\epsilon)$ iterations, gradient descent achieves $\ell(\Theta^{(t)}) \leq \epsilon$. 
Lemma [Hardt and Ma] (Gradient is big when loss is big)

If, for all layers $i$, $\sigma_{\min}(\Theta_i) \geq 1 - a$, then $\|\nabla_{\Theta} \ell(\Theta)\|^2 \geq 4\ell(\Theta)L(1 - a)^{2L}$.

Lemma (Hessian is small for near-identities)

For $\Theta$ with $\|\Theta_i\|_2 \leq 1 + z$ for all $i$,

$$\|\nabla^2_{\Theta} \ell(\Theta)\|_F \leq 3Ld^5(1 + z)^{2L}.$$

Lemma (Stay close to the identity)

$$R(t + 1) \leq R(t) + \eta(1 + R(t))^L \sqrt{2\ell(t)},$$

where $R(t) := \max_i \|\Theta_i^{(t)} - I\|_2$ and $\ell(t) := \frac{1}{2}\|\Theta_{1:L}^{(t)} - \Phi\|_F^2$.

Then for sufficiently small step-size $\eta$, the gradient update ensures that $\ell(t)$ decreases exponentially.
Deep residual networks
Optimization in deep linear residual networks
  - Gradient descent
  - **Symmetric maps and positivity**
    - Regularized gradient descent and positive maps
Statistical complexity of deep networks
Definition (positive margin matrix)

A matrix $A$ has margin $\gamma > 0$ if, for all unit length $u$, we have $u^\top Au > \gamma$.

Theorem

Suppose $\Phi$ is symmetric.

(a) There is an absolute positive constant $c_3$ such that if $\Phi$ has margin $0 < \gamma < 1$, $L \geq c_3 \ln (\|\Phi\|_2 / \gamma)$, and $\eta \leq \frac{1}{L(1+\|\Phi\|_2^2)}$, after $t = \text{poly}(L, \|\Phi\|_2 / \gamma, 1/\eta) \log(d/\epsilon)$ iterations, gradient descent achieves $\ell(f_{\Theta(t)}) \leq \epsilon$.

(b) If $\Phi$ has a negative eigenvalue $-\lambda$ and $L$ is even, then gradient descent satisfies $\ell(\Theta(t)) \geq \lambda^2 / 2$ (as does any penalty-regularized version of gradient descent).
Symmetric linear functions

Proof idea

(a) A set of symmetric matrices $\mathcal{A}$ is *commuting normal* if there is a single unitary matrix $U$ such that for all $A \in \mathcal{A}$, $U^\top AU$ is diagonal. Clearly, $\{\Phi, \Theta_1^{(0)}, \Theta_2^{(0)}, \ldots, \Theta_L^{(0)}\} = \{\Phi, I\}$ is commuting normal. The gradient update keeps $\bigcup_{i,t} \{\Phi, \Theta_i^{(t)}\}$ commuting normal. So the dynamics decomposes:

$$\hat{\lambda}^{(t+1)} = \hat{\lambda}^{(t)} + \eta(\hat{\lambda}^{(t)})^{L-1}(\lambda^L - (\hat{\lambda}^{(t)})^L).$$

(b) The eigenvalues stay positive.
Outline

- Deep residual networks
- Optimization in deep linear residual networks
  - Gradient descent
  - Symmetric maps and positivity
    - Regularized gradient descent and positive maps
- Statistical complexity of deep networks
Positive (not necessarily symmetric) linear functions

**Theorem**

For any $\Phi$ with margin $\gamma$, there is an algorithm (power projection) that, after $t = \text{poly}(d, \|\Phi\|_F, \frac{1}{\gamma}) \log(1/\epsilon)$ iterations, produces $\Theta^{(t)}$ with $\ell(\Theta^{(t)}) \leq \epsilon$.

**Power projection algorithm idea**

1. Take a gradient step for each $\Theta_i$.
2. Project $\Theta_{1:L}$ onto the set of linear maps with margin $\gamma$.
3. Set $\Theta_1^{(t+1)}, \ldots, \Theta_L^{(t+1)}$ as the balanced factorization of $\Theta_{1:L}$.
Positive (not necessarily symmetric) linear functions

Balanced factorization

We can write any matrix $A$, with singular values $\sigma_1, \ldots, \sigma_d$, as $A = A_L \cdots A_1$, where the singular values of each $A_i$ are $\sigma_1^{1/L}, \ldots, \sigma_d^{1/L}$.

(Idea: Write the polar decomposition $A = RP$ (i.e., $R$ unitary, $P$ psd); set $A_i = R^{1/L}P_i$, with $P_i = R^{(i-1)/L}P^{1/L}R^{-(i-1)/L}$.)
Gradient descent
- converges if \( \ell(0) \) sufficiently small,
- converges for a positive symmetric map,
- cannot converge for a symmetric map with a negative eigenvalue.

Regularized gradient descent converges for a positive map.

Convergence is linear in all cases.

Deep nonlinear residual networks?
Outline

- Deep residual networks
- Optimization in deep linear residual networks
- **Statistical complexity of deep networks**
  - VC theory: Number of parameters
  - Margins analysis: Size of parameters
  - Understanding generalization failures
VC Theory

- Assume network maps to \([-1, 1]\).
  (Threshold its output)
- Data generated by a probability distribution \(P\) on \(\mathcal{X} \times \{-1, 1\}\).
- Want to choose a function \(f\) such that \(P(f(x) \neq y)\) is small (near optimal).
VC Theory

**Theorem** (Vapnik and Chervonenkis)

Suppose $\mathcal{F} \subseteq \{-1, 1\}^\mathcal{X}$.

For every prob distribution $P$ on $\mathcal{X} \times \{-1, 1\}$, with probability $1 - \delta$ over $n$ iid examples $(x_1, y_1), \ldots, (x_n, y_n)$, every $f$ in $\mathcal{F}$ satisfies

$$P(f(x) \neq y) \leq \frac{1}{n} \left| \{i : f(x_i) \neq y_i\} \right| + \left( \frac{c}{n} (\text{VCdim}(\mathcal{F}) + \log(1/\delta)) \right)^{1/2}.$$

- For uniform bounds (that is, for all distributions and all $f \in \mathcal{F}$, proportions are close to probabilities), this inequality is tight within a constant factor.
- For neural networks, VC-dimension:
  - increases with number of parameters
  - depends on nonlinearity and depth
Theorem

Consider the class $\mathcal{F}$ of $\{-1, 1\}$-valued functions computed by a network with $L$ layers, $p$ parameters, and $k$ computation units with the following nonlinearities:

1. **Piecewise constant (linear threshold units):**
   
   \[
   \text{VCdim}(\mathcal{F}) = \tilde{O}(p).
   \]
   
   (Baum and Haussler, 1989)

2. **Piecewise linear (ReLUs):**
   
   \[
   \text{VCdim}(\mathcal{F}) = \tilde{O}(pL).
   \]
   
   (B., Harvey, Liaw, Mehrabian, 2017)

3. **Piecewise polynomial:**
   
   \[
   \text{VCdim}(\mathcal{F}) = \tilde{O}(pL^2).
   \]
   
   (B., Maiorov, Meir, 1998)

4. **Sigmoid:**
   
   \[
   \text{VCdim}(\mathcal{F}) = \tilde{O}(p^2k^2).
   \]
   
   (Karpinsky and Maclntyre, 1994)
Generalization in Neural Networks: Number of Parameters

NIPS 1996

Experimental Results

Neural networks with many parameters, trained on small data sets, sometimes generalize well.

**Eg:** *Face recognition* (Lawrence *et al.*, 1996)

\[ m = 50 \text{ training patterns.} \]

![Graph showing error rate vs. number of weights]

- **Number of weights**
- **Number of patterns**
Deep residual networks
- Representing with near-identities
- Global optimality of stationary points

What determines the statistical complexity of a deep network?
- VC theory: Number of parameters
- **Margins analysis: Size of parameters**
- Understanding generalization failures
Consider a real-valued function $f : \mathcal{X} \rightarrow \mathbb{R}$ used for classification. The prediction on $x \in \mathcal{X}$ is $\text{sign}(f(x)) \in \{-1, 1\}$. For a pattern-label pair $(x, y) \in \mathcal{X} \times \{-1, 1\}$, if $yf(x) > 0$ then $f$ classifies $x$ correctly. We call $yf(x)$ the margin of $f$ on $x$. We can view a larger margin as a more confident correct classification. Minimizing a continuous loss, such as

$$\sum_{i=1}^{n} (f(X_i) - Y_i)^2,$$

encourages large margins. For large-margin classifiers, we should expect the fine-grained details of $f$ to be less important.
Generalization: Margins and Size of Parameters

Theorem (B., 1996)

1. With high probability over \( n \) training examples \((X_1, Y_1), \ldots, (X_n, Y_n) \in \mathcal{X} \times \{\pm 1\}\), every \( f \in \mathcal{F} \subset \mathbb{R}^\mathcal{X} \) has

\[
Pr(\text{sign}(f(X)) \neq Y) \leq \frac{1}{n} \sum_{i=1}^{n} 1[Y_i f(X_i) \leq \gamma] + \tilde{O}\left(\sqrt{\frac{\text{fat}_{\mathcal{F}}(\gamma)}{n}}\right).
\]

2. If functions in \( \mathcal{F} \) are computed by two-layer sigmoid networks with each unit’s weights bounded in 1-norm, that is, \( \|w\|_1 \leq B \), then

\[
\text{fat}_{\mathcal{F}}(\gamma) = \tilde{O}((B/\gamma)^2).
\]

- The bound depends on the margin loss plus a complexity term.
- Minimizing quadratic loss or cross-entropy loss leads to large margins.
- \( \text{fat}_{\mathcal{F}}(\gamma) \) is a scale-sensitive version of VC-dimension. Unlike the VC-dimension, it need not grow with the number of parameters.
Generalization: Margins and Size of Parameters

1996: Sigmoid networks

- Qualitative behavior consistent with small weights theorem.

2017: Deep ReLU networks

- How to measure the complexity of a ReLU network?

Simons.berkeley.edu
Deep residual networks
- Representing with near-identities
- Global optimality of stationary points

What determines the statistical complexity of a deep network?
- VC theory: Number of parameters
- Margins analysis: Size of parameters
- **Understanding generalization failures**
Explaining Generalization Failures

CIFAR10

http://corochann.com/
Explaining Generalization Failures

Stochastic Gradient Training Error on CIFAR10

(Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals, 2017)
Explaining Generalization Failures

<table>
<thead>
<tr>
<th>Training margins on CIFAR10 with true and random labels</th>
</tr>
</thead>
</table>

- How does this match the large margin explanation?
- Need to account for the *scale* of the neural network functions.
- What is the appropriate notion of the size of these functions?

Dylan Foster
Cornell

Matus Telgarsky
UIUC
New results for generalization in deep ReLU networks

- Measuring the size of functions computed by a network of ReLUs. (c.f. sigmoid networks: the output $y$ of a layer has $\|y\|_\infty \leq 1$, so $\|w\|_1 \leq B$ keeps the scale under control.)
- Large multiclass versus binary classification.

Definitions

- Consider operator norms: For a matrix $A_i$,
  $$\|A_i\|_* := \sup_{\|x\| \leq 1} \|A_i x\|.$$

- Multiclass margin function for $f : \mathcal{X} \rightarrow \mathbb{R}^m$, $y \in \{1, \ldots, m\}$:
  $$M(f(x), y) = f(x)_y - \max_{i \neq y} f(x)_i.$$
Theorem

With high probability, every $f_A$ with $R_A \leq r$ satisfies

$$\Pr(M(f_A(X), Y) \leq 0) \leq \frac{1}{n} \sum_{i=1}^{n} 1[M(f_A(X_i), Y_i) \leq \gamma] + \tilde{O}\left(\frac{rL}{\gamma \sqrt{n}}\right).$$

Definitions

Network with $L$ layers, parameters $A_1, \ldots, A_L$:

$$f_A(x) := \sigma_L(A_L \sigma_{L-1}(A_{L-1} \cdots \sigma_1(A_1 x) \cdots)).$$

Scale of $f_A$: $R_A := \prod_{i=1}^{L} \|A_i\|_* \left(\sum_{i=1}^{L} \frac{\|A_i\|^{2/3}}{\|A_i\|_*^{2/3}}\right)^{3/2}$.

(Assume $\sigma_i$ is 1-Lipschitz, inputs normalized.)
Explaining Generalization Failures

Stochastic Gradient Training Error on CIFAR10

(Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals, 2017)
How does this match the large margin explanation?
Explaining Generalization Failures

If we rescale the margins by $R_A$ (the scale parameter):

**Rescaled margins on CIFAR10**

- **cifar**
- **rand label**
Explaining Generalization Failures

If we rescale the margins by $R_A$ (the scale parameter):

Rescaled cumulative margins on MNIST
Theorem

With high probability, every $f_A$ with $R_A \leq r$ satisfies

$$\Pr(M(f_A(X), Y) \leq 0) \leq \frac{1}{n} \sum_{i=1}^{n} 1[M(f_A(X_i), Y_i) \leq \gamma] + \tilde{O} \left( \frac{rL}{\gamma \sqrt{n}} \right).$$

Network with $L$ layers, parameters $A_1, \ldots, A_L$:

$$f_A(x) := \sigma(A_L \sigma_{L-1}(A_{L-1} \cdots \sigma_1(A_1 x) \cdots)).$$

Scale of $f_A$: $R_A := \prod_{i=1}^{L} \|A_i\|_* \left( \frac{\sum_{i=1}^{L} \|A_i\|^{2/3}_{2,1}}{\|A_i\|^{2/3}_*} \right)^{3/2}$. 
Explaining Generalization Failures
Explaining Generalization Failures

- Excess risk 0.3
- Excess risk 0.9
- Cifar excess risk
- Cifar Lipschitz
- Cifar [random] excess risk
- Cifar [random] Lipschitz

Epochs: 10, 100
With appropriate normalization, the margins analysis is qualitatively consistent with the generalization performance.

Margin bounds extend to residual networks.

Recent work by Golowich, Rakhlin, and Shamir give bounds with improved dependence on depth.

Lower bounds?

Regularization: explicit control of operator norms?

Role of depth?
Deep residual networks
  - Representing with near-identities
  - Global optimality of stationary points

Optimization in deep linear residual networks
  - Gradient descent
  - Symmetric maps and positivity
  - Regularized gradient descent and positive maps

Statistical complexity of deep networks
  - VC theory: Number of parameters
  - Margins analysis: Size of parameters
  - Understanding generalization failures