Generalization in Deep Networks

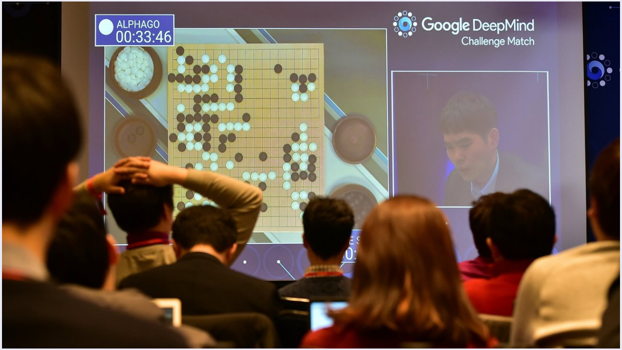
Peter Bartlett

BAIR
UC Berkeley

November 28, 2017

Deep neural networks

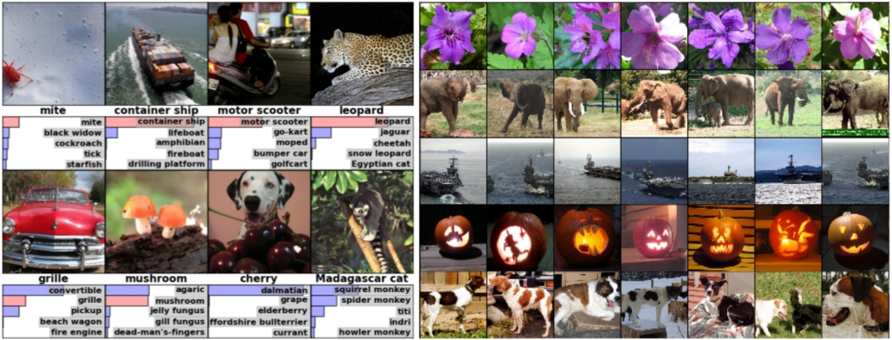
Game playing



(Jung Yeon-Je/AFP/Getty Images)

Deep neural networks

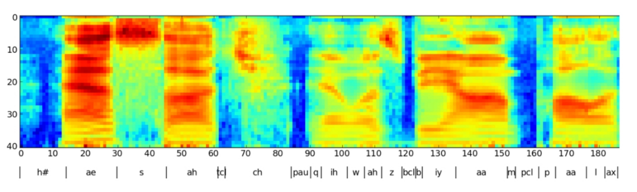
Image recognition



(Krizhevsky et al, 2012)

Deep neural networks

Speech recognition



(Graves et al, 2013)

Outline

- What determines the statistical complexity of a deep network?

Outline

- What determines the statistical complexity of a deep network?
  - VC theory: Number of parameters

Outline

- What determines the statistical complexity of a deep network?
  - VC theory: Number of parameters
  - Margins analysis: Size of parameters

Outline

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

Outline

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

VC Theory

VC Theory

- Assume network maps to $\{-1, 1\}$.
  (Threshold its output)

VC Theory

- Assume network maps to $\{-1, 1\}$.
  (Threshold its output)
- Data generated by a probability distribution P on $\mathcal{X} \times \{-1, 1\}$.

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

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

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

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

- 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.

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

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

- 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:

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

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

- 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

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

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

- 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

VC-Dimension of Neural Networks

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:

VC-Dimension of Neural Networks

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): $\mathrm{VCdim}(\mathcal{F}) = \tilde{O}(p)$.

(Baum and Haussler, 1989)

VC-Dimension of Neural Networks

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): $\mathrm{VCdim}(\mathcal{F}) = \tilde{O}(p)$.

   (Baum and Haussler, 1989)

2. Piecewise linear (ReLUs): $\mathrm{VCdim}(\mathcal{F}) = \tilde{O}(pL)$.

   (B., Harvey, Liaw, Mehrabian, 2017)

VC-Dimension of Neural Networks

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): $\mathrm{VCdim}(\mathcal{F}) = \tilde{O}(p)$.
   (Baum and Haussler, 1989)
2. Piecewise linear (ReLUs): $\mathrm{VCdim}(\mathcal{F}) = \tilde{O}(pL)$.
   (B., Harvey, Liaw, Mehrabian, 2017)
3. Piecewise polynomial: $\mathrm{VCdim}(\mathcal{F}) = \tilde{O}(pL^2)$.
   (B., Maiorov, Meir, 1998)

VC-Dimension of Neural Networks

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): $\mathrm{VCdim}(\mathcal{F}) = \tilde{O}(p)$.
   (Baum and Haussler, 1989)
2. Piecewise linear (ReLUs): $\mathrm{VCdim}(\mathcal{F}) = \tilde{O}(pL)$.
   (B., Harvey, Liaw, Mehrabian, 2017)
3. Piecewise polynomial: $\mathrm{VCdim}(\mathcal{F}) = \tilde{O}(pL^2)$.
   (B., Maiorov, Meir, 1998)
4. Sigmoid: $\mathrm{VCdim}(\mathcal{F}) = \tilde{O}(p^2 k^2)$.
   (Karpinsky and MacIntyre, 1994)

Generalization in Neural Networks: Number of Parameters

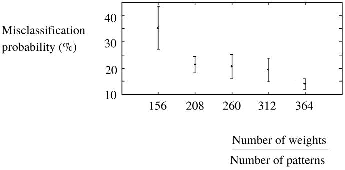
NIPS 1996

Experimental Results

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

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

$m = 50$ training patterns.



Outline

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

Large-Margin Classifiers

- Consider a real-valued function $f : \mathcal{X} \to \mathbb{R}$ used for classification.

Large-Margin Classifiers

- Consider a real-valued function $f : \mathcal{X} \to \mathbb{R}$ used for classification.
- The prediction on $x \in \mathcal{X}$ is $\text{sign}(f(x)) \in \{-1, 1\}$.

Large-Margin Classifiers

- Consider a real-valued function $f : \mathcal{X} \to \mathbb{R}$ used for classification.
- The prediction on $x \in \mathcal{X}$ is $\mathrm{sign}(f(x)) \in \{-1, 1\}$.
- For a pattern-label pair $(x, y) \in \mathcal{X} \times \{-1, 1\}$,
  if $yf(x) > 0$ then f classifies x correctly.

Large-Margin Classifiers

- Consider a real-valued function $f : \mathcal{X} \to \mathbb{R}$ used for classification.
- The prediction on $x \in \mathcal{X}$ is $\mathrm{sign}(f(x)) \in \{-1, 1\}$.
- 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.

Large-Margin Classifiers

- Consider a real-valued function $f : \mathcal{X} \to \mathbb{R}$ used for classification.
- The prediction on $x \in \mathcal{X}$ is $\mathrm{sign}(f(x)) \in \{-1, 1\}$.
- 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.

Large-Margin Classifiers

- Consider a real-valued function $f : \mathcal{X} \to \mathbb{R}$ used for classification.
- The prediction on $x \in \mathcal{X}$ is $\mathrm{sign}(f(x)) \in \{-1, 1\}$.
- 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.

Large-Margin Classifiers

- Consider a real-valued function $f : \mathcal{X} \to \mathbb{R}$ used for classification.
- The prediction on $x \in \mathcal{X}$ is $\mathrm{sign}(f(x)) \in \{-1, 1\}$.
- 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)

Generalization: Margins and Size of Parameters

Theorem (B., 1996)

n training examples

$(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathcal{X} \times \{\pm 1\}$

Generalization: Margins and Size of Parameters

Theorem (B., 1996)

n training examples

$(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathcal{X} \times \{\pm 1\}$ $\qquad f \in \mathcal{F} \subset \mathbb{R}^{\mathcal{X}}$

Generalization: Margins and Size of Parameters

Theorem (B., 1996)

n training examples

$(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathcal{X} \times \{\pm 1\}$ $\quad f \in \mathcal{F} \subset \mathbb{R}^{\mathcal{X}}$

$$\Pr(\mathrm{sign}(f(X)) \neq Y)$$

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

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

- The bound depends on the margin loss plus an error term.

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

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

- The bound depends on the margin loss plus an error term.
- Minimizing quadratic loss or cross-entropy loss leads to large margins.

Generalization: Margins and Size of Parameters

Theorem (B., 1996)

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

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

- The bound depends on the margin loss plus an error term.
- Minimizing quadratic loss or cross-entropy loss leads to large margins.
- $\mathrm{fat}_{\mathcal{F}}(\gamma)$ 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

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

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

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

$$\mathrm{fat}_{\mathcal{F}}(\gamma) = \tilde{O}((B/\gamma)^2).$$

- The bound depends on the margin loss plus an error term.
- Minimizing quadratic loss or cross-entropy loss leads to large margins.
- $\mathrm{fat}_{\mathcal{F}}(\gamma)$ 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

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

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

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

$$\mathrm{fat}_{\mathcal{F}}(\gamma) = \tilde{O}((B/\gamma)^2).$$

- Same ideas used to give rigorous dimension-independent generalization bounds for SVMs (B. and Shawe-Taylor, 1999)

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

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

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

$$\mathrm{fat}_{\mathcal{F}}(\gamma) = \tilde{O}((B/\gamma)^2).$$

- Same ideas used to give rigorous dimension-independent generalization bounds for SVMs (B. and Shawe-Taylor, 1999)
- ... and margins analysis of AdaBoost. (Schapire, Freund, B., Lee, 1998)

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

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

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

$$\mathrm{fat}_{\mathcal{F}}(\gamma) = \tilde{O}((B/\gamma)^2).$$

- The scale of functions $f \in \mathcal{F}$ is 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

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

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

$$\operatorname{fat}_{\mathcal{F}}(\gamma) = \tilde{O}((B/\gamma)^2).$$

- The scale of functions $f \in \mathcal{F}$ is important.
- Bigger fs give bigger margins, so $\operatorname{fat}_{\mathcal{F}}(\gamma)$ should be bigger.

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

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

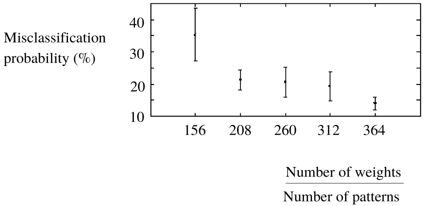
2. If functions in $\mathcal{F}$ are computed by two-layer sigmoid networks with each unit's weights bounded in 1-norm, that is, $\|w\|_1 \leq B$, then

$$\mathrm{fat}_{\mathcal{F}}(\gamma) = \tilde{O}((B/\gamma)^2).$$

- The scale of functions $f \in \mathcal{F}$ is important.
- Bigger fs give bigger margins, so $\mathrm{fat}_{\mathcal{F}}(\gamma)$ should be bigger.
- The output y of a sigmoid layer has $\|y\|_\infty \leq 1$, so $\|w\|_1 \leq B$ controls the scale of f.

Generalization: Margins and Size of Parameters

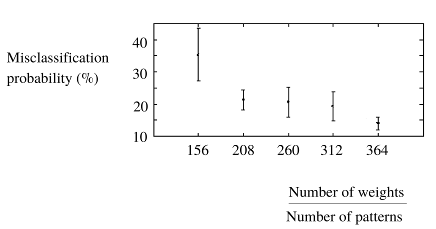
1996: Sigmoid networks



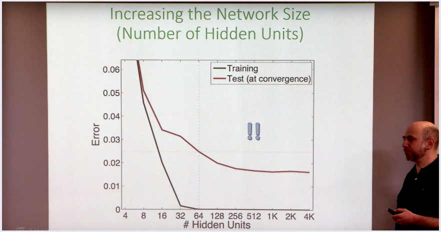
- Qualitative behavior explained by small weights theorem.

Generalization: Margins and Size of Parameters

1996: Sigmoid networks



2017: Deep ReLU networks

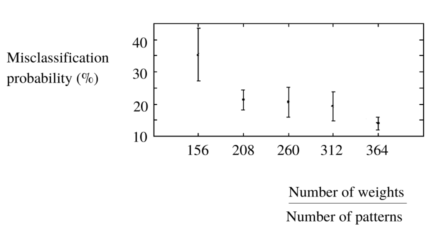


simons.berkeley.edu

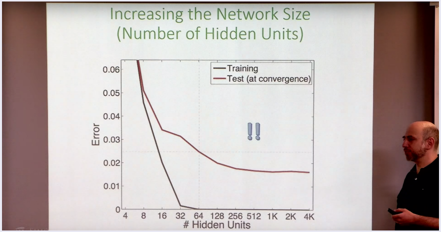
- Qualitative behavior explained by small weights theorem.

Generalization: Margins and Size of Parameters

1996: Sigmoid networks



2017: Deep ReLU networks



simons.berkeley.edu

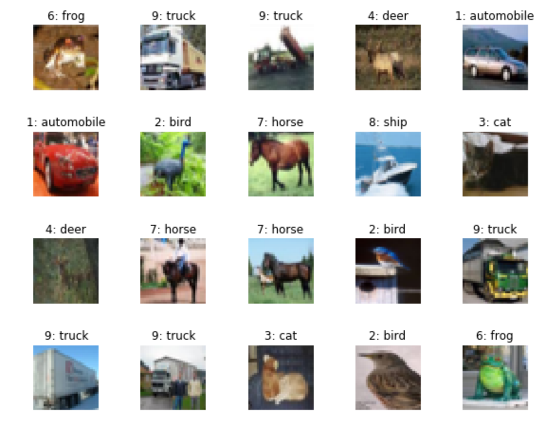
- Qualitative behavior explained by small weights theorem.
- How to measure the complexity of a ReLU network?

Outline

- 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

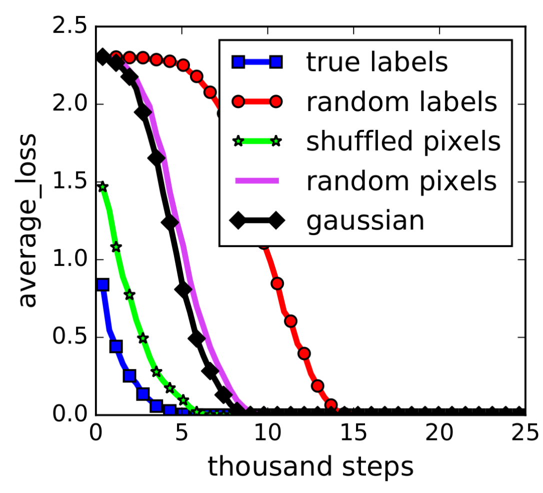
CIFAR10



http://corochann.com/

Explaining Generalization Failures

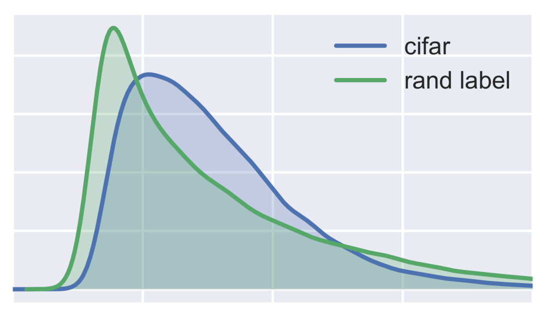
Stochastic Gradient Training Error on CIFAR10



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

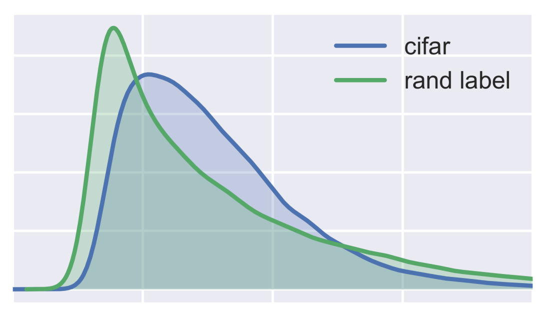
Explaining Generalization Failures

Training margins on CIFAR10 with true and random labels



Explaining Generalization Failures

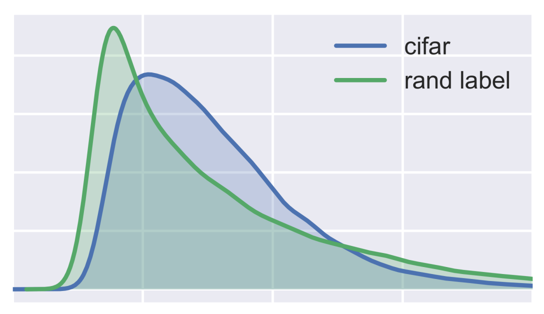
Training margins on CIFAR10 with true and random labels



- How does this match the large margin explanation?

Explaining Generalization Failures

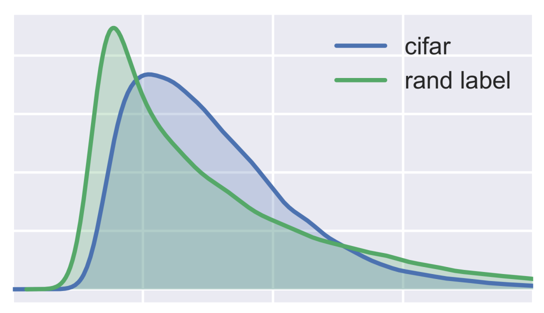
Training margins on CIFAR10 with true and random labels



- How does this match the large margin explanation?
- Need to account for the *scale* of the neural network functions.

Explaining Generalization Failures

Training margins on CIFAR10 with true and random labels



- 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?

Generalization in Deep Networks

Spectrally-normalized margin bounds for neural networks.
B., Dylan J. Foster, Matus Telgarsky, 2017.
arXiv:1706.08498

Matus Telgarsky
UIUC

Dylan Foster
Cornell

Generalization in Deep Networks

New results for generalization in deep ReLU networks

- Measuring the size of functions computed by a network of ReLUs.

Generalization in Deep Networks

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.)

Generalization in Deep Networks

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.

Generalization in Deep Networks

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,

$$\|A_i\|_* := \sup_{\|x\| \leq 1} \|A_i x\|.$$

Generalization in Deep Networks

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 $f : \mathcal{X} \to \mathbb{R}^m$, $y \in \{1, \ldots, m\}$:

$$M(f(x), y) = f(x)_y - \max_{i \neq y} f(x)_i.$$

Generalization in Deep Networks

Theorem

With high probability, every f_A

Generalization in Deep Networks

Theorem

With high probability, every f_A

Definitions

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

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

Generalization in Deep Networks

Theorem

With high probability, every f_A satisfies

$$\Pr(M(f_A(X), Y) \leq 0) \leq$$

Definitions

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

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

Generalization in Deep Networks

Theorem

With high probability, every f_A satisfies

$$\Pr(M(f_A(X), Y) \leq 0) \leq \frac{1}{n}\sum_{i=1}^{n} 1[M(f_A(X_i), Y_i) \leq \gamma]$$

Definitions

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

$$f_A(x) := \sigma_L(A_L\sigma_{L-1}(A_{L-1}\cdots\sigma_1(A_1x)\cdots)).$$

Generalization in Deep Networks

Theorem

With high probability, every f_A satisfies

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

Definitions

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

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

Generalization in Deep Networks

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: $R_A := \prod_{i=1}^{L} \|A_i\|_* \sqrt{\sum_{i=1}^{L} \frac{\|A_i\|_F}{\|A_i\|_*}}$.

Generalization in Deep Networks

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 $A_1, \ldots, A_L$:

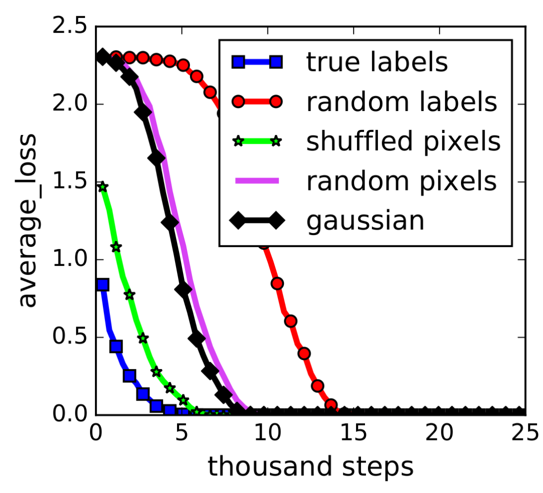
$$f_A(x) := \sigma_L(A_L\sigma_{L-1}(A_{L-1}\cdots\sigma_1(A_1 x)\cdots)).$$

Scale of f_A: $R_A := \prod_{i=1}^{L} \|A_i\|_* \sqrt{\sum_{i=1}^{L} \frac{\|A_i\|_F}{\|A_i\|_*}}$.

(Assume σ_i is 1-Lipschitz, inputs normalized.)

Explaining Generalization Failures

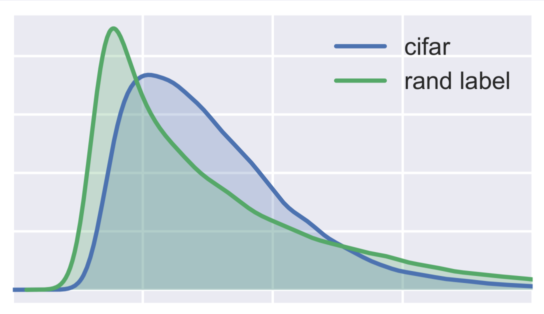
Stochastic Gradient Training Error on CIFAR10



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

Explaining Generalization Failures

Training margins on CIFAR10 with true and random labels

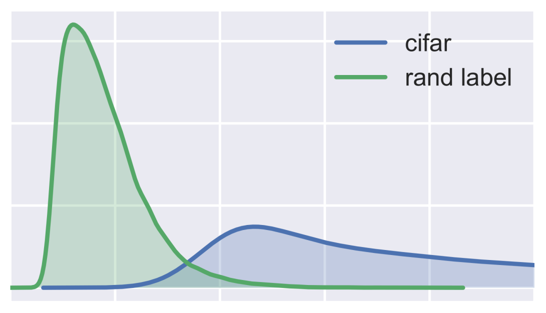


- How does this match the large margin explanation?

Explaining Generalization Failures

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

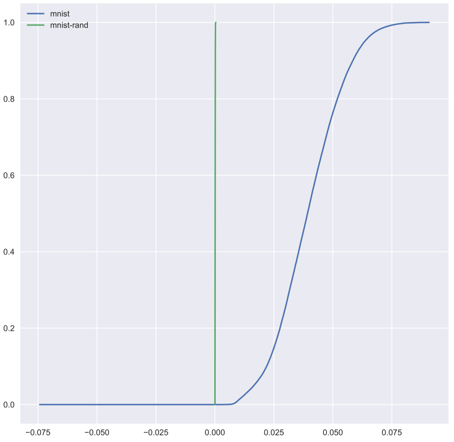
Rescaled margins on CIFAR10



Explaining Generalization Failures

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

Rescaled cumulative margins on MNIST



Generalization in Deep Networks

Theorem

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

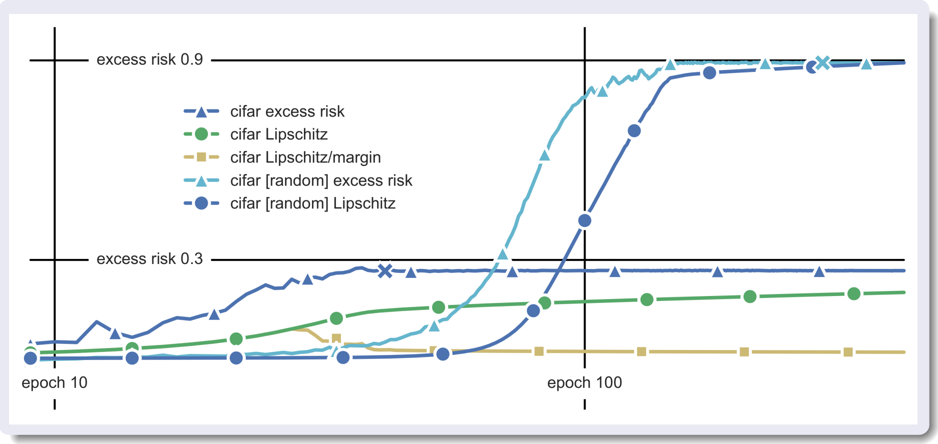
$$\Pr(M(f_A(X), Y) \le 0) \le \frac{1}{n}\sum_{i=1}^{n} 1[M(f_A(X_i), Y_i) \le \gamma] + \tilde{O}\left(\frac{rL}{\gamma\sqrt{n}}\right).$$

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: $R_A := \prod_{i=1}^{L} \|A_i\|_* \sqrt{\sum_{i=1}^{L} \frac{\|A_i\|_F}{\|A_i\|_*}}$.

Explaining Generalization Failures



Generalization in Neural Networks

- With appropriate normalization, the margins analysis is qualitatively consistent with the generalization performance.

Generalization in Neural Networks

- With appropriate normalization, the margins analysis is qualitatively consistent with the generalization performance.
- Margin bounds extend to residual networks.

Generalization in Neural Networks

- With appropriate normalization, the margins analysis is qualitatively consistent with the generalization performance.
- Margin bounds extend to residual networks.
- Lower bounds?

Generalization in Neural Networks

- With appropriate normalization, the margins analysis is qualitatively consistent with the generalization performance.
- Margin bounds extend to residual networks.
- Lower bounds?
- Regularization: explicit control of operator norms?

Generalization in Neural Networks

- With appropriate normalization, the margins analysis is qualitatively consistent with the generalization performance.
- Margin bounds extend to residual networks.
- Lower bounds?
- Regularization: explicit control of operator norms?
- Role of depth?

Generalization in Neural Networks

- With appropriate normalization, the margins analysis is qualitatively consistent with the generalization performance.
- Margin bounds extend to residual networks.
- Lower bounds?
- Regularization: explicit control of operator norms?
- Role of depth?
- Interplay with optimization?

Outline

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