UNDERSTANDING DEEP LEARNING REQUIRES RETHINKING GENERALIZATION

Chiyuan Zhang Samy Bengjo Moritz Hardt Benjamin Recht Oriol Vinyals

ICLR 2017 Best Paper

Lee Joonam
slack : 이주남_T1163
email : joonamm@naver.com

Contents

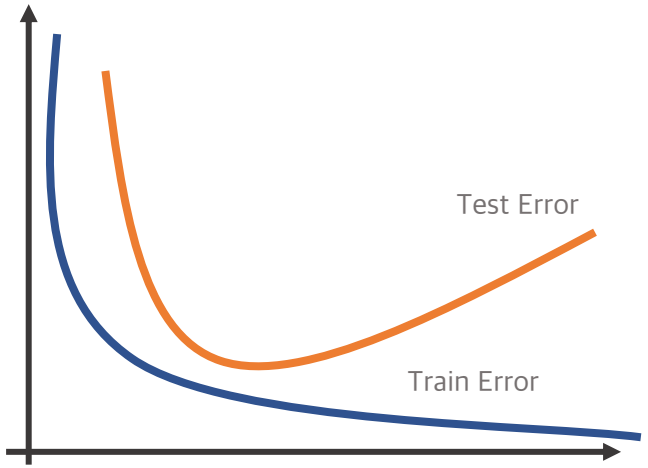
UNDERSTANDING DEEP LEARNING REQUIRES RETHINKING GENERALIZATION

Contents

UNDERSTANDING DEEP LEARNING REQUIRES RETHINKING GENERALIZATION

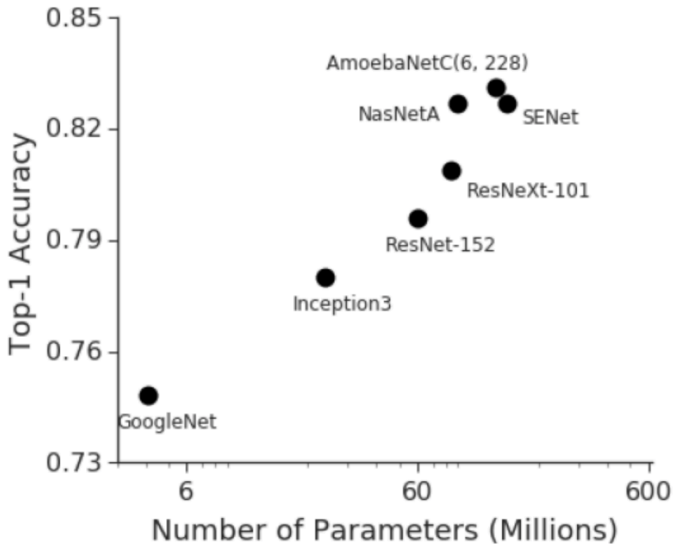
Generalization

Generalization = | train error – test error |



Lower Parameters Higher Complexity

For Generalization performance



연구 접근

UNDERSTANDING DEEP LEARNING REQUIRES
RETHINKING GENERALIZATION

Why large neural networks generalize well in practice

Can the traditional theories on generalization **explain the results we are seeing these days?**

What is it then that **distinguishes** neural networks that generalize well from those that don't?

Randomize Label

Randomize Label?

If well trained Model (Good generalization performance)

I don't know.. → So does not train. Train loss ↑

Else..

Memorize All Data → Memorize All **Train** Data → **Train Loss 0**

Experiment

Model : AlexNet, MLP, Inception Data set : CIFAR 10, IamgeNet1000

Contributions

Deep neural networks easily fit random labels.

Deep learning model 은 이미 충분한 capacity를 가지고 있다.

Explicit regularization may improve generalization performance
but is neither necessary nor by itself sufficient for controlling generalization error.

Regularization 이 generalization 성능을 개선 시키긴 하지만…

Finite-Sample expressivity

model capacity 에 대한 증명

Implicit regularization – An appeal to linear models

SDG 가 regularization 역할을 하나

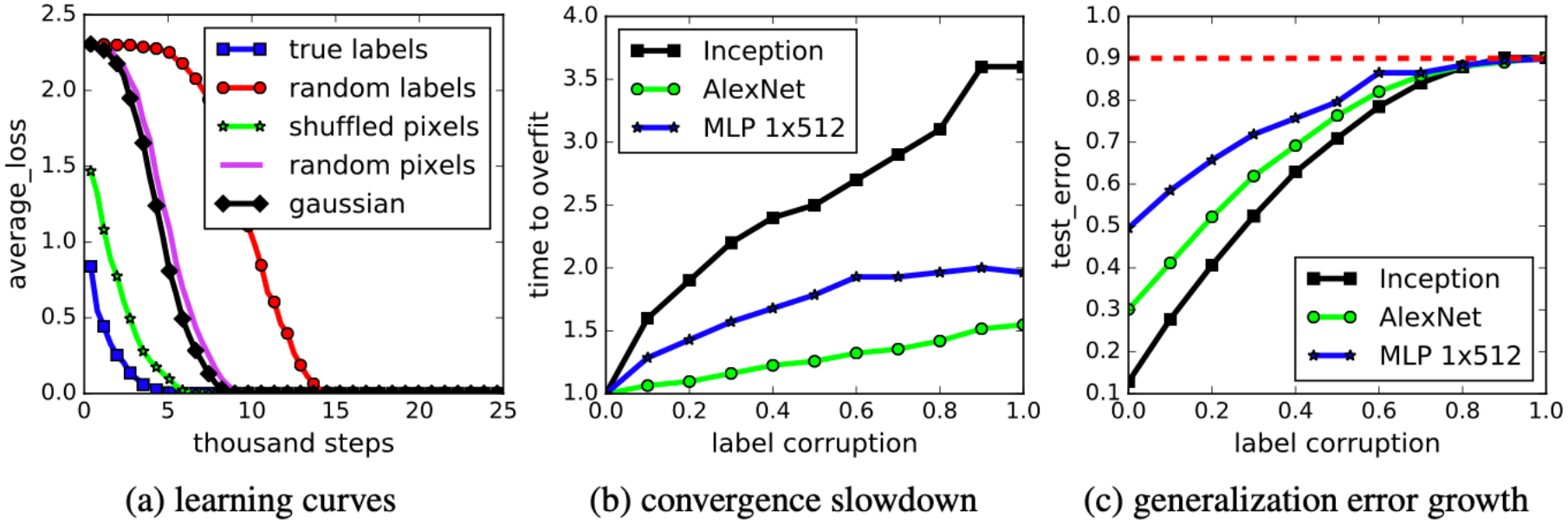
Contents

UNDERSTANDING DEEP LEARNING REQUIRES RETHINKING GENERALIZATION

Effective Capacity of Neural Networks

Data : CIFAR10

corrupted as a uniform random class
Gaussian : A Gaussian distribution (input data)
Shuffled pixels: a random permutation of the pixels is chosen then
the same permutation is applied to all the images in both training and test set.



(a) learning curves (b) convergence slowdown (c) generalization error growth

Model has large capacity, memorized.. All train data..

다양한 모델을 사용하여도 Overffing이 발생

Implications (Traditional Approaches)

Redemacher complexity and VC-dimension (statistical Learning)

$$\hat{\mathfrak{R}}_s(\mathcal{H}) = \frac{1}{n}\mathbb{E}_\sigma\left[\sup_{h\ \mathcal{H}}\sum_{i=1}^{n}\sigma_i h(x_i)\right]$$

$\sigma_1, \sigma_2, \ldots \sigma_n \in \{-1, +1\}$ 균등 분포 random 변수

$x_1, x_2, \ldots x_n$ Datasets

$h \in \mathcal{H}$ 가설 함수 (model)

Model capacity dependent on number of **parameters**

Uniform stability

알고리즘의 민감도에 따라서 모델의 Complexity 를 측정 하는 방법론

Single sample 을 넣어서 얼마나 민감 하게 반응하는지를 측정

Didn’t handle parameters or algorithm

Just randomize and noise DATA

Summary

Deep Learning easily fit random label

Capacity 가 모든 데이터를 memorize 할 만큼 충분

Traditional view couldn’t explain complexity of Deep learning

Rademacher complexity & Uniform stability

Parameter, Algorithm(model) 이 complexity 를 충분히 설명 하기엔 힘듦

The Role of Regularization

SO Deep Learning model has large capacity to fit all data.

Use regularization (Dropout, Data augmentation..)
Improve Generalization performance

Really??

The Role of Regularization

Is regularization improve test accuracy(generalization performance)?

model	# params	random crop	weight decay	train accuracy	test accuracy
Inception	1,649,402	yes	yes	100.0	89.05
		yes	no	100.0	89.31
		no	yes	100.0	86.03
		no	no	100.0	85.75
(fitting random labels)		no	no	100.0	9.78
Inception w/o BatchNorm	1,649,402	no	yes	100.0	83.00
		no	no	100.0	82.00
(fitting random labels)		no	no	100.0	10.12
Alexnet	1,387,786	yes	yes	99.90	81.22
		yes	no	99.82	79.66
		no	yes	100.0	77.36
		no	no	100.0	76.07
(fitting random labels)		no	no	99.82	9.86
MLP 3x512	1,735,178	no	yes	100.0	53.35
		no	no	100.0	52.39
(fitting random labels)		no	no	100.0	10.48
MLP 1x512	1,209,866	no	yes	99.80	50.39
		no	no	100.0	50.51
(fitting random labels)		no	no	99.34	10.61

The Role of Regularization

Is regularization improve test accuracy(generalization performance)? **Yes.**

model	# params	random crop	weight decay	train accuracy	test accuracy
Inception	1,649,402	yes	yes	100.0	89.05
		yes	no	100.0	89.31
		no	yes	100.0	86.03
		no	no	100.0	85.75
(fitting random labels)		no	no	100.0	9.78
Inception w/o BatchNorm	1,649,402	no	yes	100.0	83.00
		no	no	100.0	82.00
(fitting random labels)		no	no	100.0	10.12
Alexnet	1,387,786	yes	yes	99.90	81.22
		yes	no	99.82	79.66
		no	yes	100.0	77.36
		no	no	100.0	76.07
(fitting random labels)		no	no	99.82	9.86
MLP 3x512	1,735,178	no	yes	100.0	53.35
		no	no	100.0	52.39
(fitting random labels)		no	no	100.0	10.48
MLP 1x512	1,209,866	no	yes	99.80	50.39
		no	no	100.0	50.51
(fitting random labels)		no	no	99.34	10.61

The Role of Regularization

Is regularization improve test accuracy(generalization performance)?
Is regularization force to generalization?

Inception v3 model

data aug	dropout	weight decay	top-1 train	top-5 train	top-1 test	top-5 test
ImageNet 1000 classes with the original labels						
yes	yes	yes	92.18	99.21	77.84	93.92
yes	no	no	92.33	99.17	72.95	90.43
no	no	yes	90.60	100.0	67.18 (72.57)	86.44 (91.31)
no	no	no	99.53	100.0	59.80 (63.16)	80.38 (84.49)
Alexnet (Krizhevsky et al., 2012)			-	-	-	83.6
ImageNet 1000 classes with random labels						
no	yes	yes	91.18	97.95	0.09	0.49
no	no	yes	87.81	96.15	0.12	0.50
no	no	no	95.20	99.14	0.11	0.56

top-n : softmax 된 결과 값 상위 n 개의 클래스에 target이 포함되어 있는지 에 대한 정확도cc

The Role of Regularization

Is regularization improve test accuracy(generalization performance)? **Yes.**
Is regularization force to generalization? **No.**

Inception v3 model

data aug	dropout	weight decay	top-1 train	top-5 train	top-1 test	top-5 test
ImageNet 1000 classes with the original labels						
yes	yes	yes	92.18	99.21	77.84	93.92
yes	no	no	92.33	99.17	72.95	90.43
no	no	yes	90.60	100.0	67.18 (72.57)	86.44 (91.31)
no	no	no	99.53	100.0	59.80 (63.16)	80.38 (84.49)
Alexnet (Krizhevsky et al., 2012)			-	-	-	83.6
ImageNet 1000 classes with random labels						
no	yes	yes	91.18	97.95	0.09	0.49
no	no	yes	87.81	96.15	0.12	0.50
no	no	no	95.20	99.14	0.11	0.56

top-n : softmax 된 결과 값 상위 n 개의 클래스에 target이 포함되어 있는지 에 대한 정확도cc

Summary

Regularization could help to improve the generalization performance
when properly tuned

Regularizers are *not fundamental reason for generalization*

Because networks continue to perform well after all the regularizers removed.

Memorize 관련해선 막아 주지 않는다.

Experiment Summary Randomize label 을 통해서 실험한 결과

Model capacity

Deep Learning model 은 모든 데이터를 memorize 할만한 large capacity 를 점유
이는 기존의 Traditional 한 관점으로써는 설명 할 수 없는 부분이 존재

Model capacity 를 측정할 방법은?

Regularization

regularization (weight decay, drop out, BN) generalization 성능 향상에 기여
하지만 generalization성능 향상에 근본적인 기여를 한다고 보기엔 어려움
Random labeling 을 한 경우 overfitting이 일어나는 상황이 관측됨

딥러닝 모델에 대한 이해도 부족한 듯 보인다.

Contents

UNDERSTANDING DEEP LEARNING REQUIRES RETHINKING GENERALIZATION

Proof

Finite-Sample Expressivity

Model capacity 를 측정할 방법은?

2-layer neural net 으로 Model capacity 를 측정.

Implicit Regularization

Deep learning 모델에 대한 이해도 부족한 듯 보인다.

그렇다면 간단한 선형 모델에 대한 일반화 측면은 어찌 바라 볼 수 있을까?
Implicit 한 Regularization 인 SGD 를 선형 모델에 적용

Proof

Finite-Sample Expressivity

Model capacity 를 측정할 방법은?

2-layer neural net 으로 Model capacity 를 측정.

Theorem 1. *There exists a two-layer neural network with ReLU activations and $2n + d$ weights that can represent any function on a sample of size n in d dimensions.*

Implicit Regularization

Deep learning 모델에 대한 이해도 부족한 듯 보인다.

그렇다면 간단한 선형 모델에 대한 일반화 측면은 어찌 바라 볼 수 있을까?
Implicit 한 Regularization 인 SGD 를 선형 모델에 적용

Finite-Sample Expressivity

Parameter 가 데이터 수 보다 크면
데이터를 모두 표현 할 수 있는 함수들을 생성할 수 있다.

정리

Theorem 1. *There exists a two-layer neural network with ReLU activations and* $2n+d$ *weights that can represent any function on a sample of size* n *in* d *dimensions.*

"d개의 차원을 가진 n개의 샘플 데이터는
2n+d 개의 weight를 가지고 Activation function 이 ReLU 인 2층의 layer를 가진 network 를 통해
어떠한 함수 든 표현 할 수 있다." (모두 표현이 가능하다)

보조 정리

$x_1, x_2, \ldots x_n$ Datasets

Lemma 1. *For any two interleaving sequences of* n *real numbers* $b_1 < x_1 < b_2 < x_2 \cdots < b_n < x_n$*, the* $n \times n$ *matrix* $A = [\max\{x_i - b_j, 0\}]_{ij}$ *has full rank. Its smallest eigenvalue is* $\min_i x_i - b_i$.

$$A = \begin{pmatrix} x_1 - b_1 & 0 & \ldots & 0 & 0 \\ x_2 - b_1 & x_2 - b_2 & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ x_{n-1} - b_1 & x_{n-1} - b_2 & \ldots & x_{n-1} - b_{n-1} & 0 \\ x_n - b_1 & x_n - b_2 & \ldots & x_n - b_{n-1} & x_n - b_n \end{pmatrix}$$

Finite-Sample Expressivity

Lemma 1. *For any two interleaving sequences of n real numbers $b_1 < x_1 < b_2 < x_2 \cdots < b_n < x_n$, the $n \times n$ matrix $A = [\max\{x_i - b_j, 0\}]_{ij}$ has full rank. Its smallest eigenvalue is $\min_i x_i - b_i$.*

$$b_1 < x_1 < \ b_2 < x_2 < b_3 < x_3 \ \ldots \ < b_n < x_n$$

$$A = [\max\{\ x_i \ - b_i, 0\ \}]_{ij}$$

$$A = \begin{bmatrix} \max\{x_1 - b_1, 0\} & \max\{x_1 - b_2, 0\} & \cdots & \max\{x_1 - b_n, 0\} \\ \max\{x_2 - b_1, 0\} & \max\{x_2 - b_2, 0\} & \cdots & \max\{x_2 - b_n, 0\} \\ \vdots & \ddots & \ddots & \vdots \\ \max\{x_n - b_1, 0\} & \max\{x_n - b_2, 0\} & \cdots & \max\{x_n - b_n, 0\} \end{bmatrix}$$

Finite-Sample Expressivity

Lemma 1. *For any two interleaving sequences of n real numbers $b_1 < x_1 < b_2 < x_2 \cdots < b_n < x_n$, the $n \times n$ matrix $A = [\max\{x_i - b_j, 0\}]_{ij}$ has full rank. Its smallest eigenvalue is $\min_i x_i - b_i$.*

$$b_1 < x_1 < b_2 < x_2 < b_3 < x_3 \ \ldots < b_n < x_n$$

$$A = [\max\{ x_i - b_i, 0 \}]_{ij}$$

$$A = \begin{bmatrix} \max\{x_1 - b_1, 0\} & \max\{x_1 - b_2, 0\} & \cdots & \max\{x_1 - b_n, 0\} \\ \max\{x_2 - b_1, 0\} & \max\{x_2 - b_2, 0\} & \cdots & \max\{x_2 - b_n, 0\} \\ \vdots & \ddots & \ddots & \vdots \\ \max\{x_n - b_1, 0\} & \max\{x_n - b_2, 0\} & \cdots & \max\{x_n - b_n, 0\} \end{bmatrix}$$

$$\overset{(i)}{=} \begin{bmatrix} x_1 - b_1 & 0 & 0 & \cdots & 0 \\ x_2 - b_1 & x_2 - b_2 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ x_{n-1} - b_1 & x_{n-1} - b_2 & \ddots & \cdots & 0 \\ x_n - b_1 & x_n - b_2 & x_n - b_3 & \cdots & x_n - b_n \end{bmatrix}$$

Finite-Sample Expressivity

보조 정리

Lemma 1. *For any two interleaving sequences of n real numbers $b_1 < x_1 < b_2 < x_2 \cdots < b_n < x_n$, the $n \times n$ matrix $A = [\max\{x_i - b_j, 0\}]_{ij}$ has full rank. Its smallest eigenvalue is $\min_i x_i - b_i$.*

A는 Invertible (역행렬 이 존재한다)

$$A = \begin{pmatrix} x_1 - b_1 & 0 & \dots & 0 & 0 \\ x_2 - b_1 & x_2 - b_2 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ x_{n-1} - b_1 & x_{n-1} - b_2 & \dots & x_{n-1} - b_{n-1} & 0 \\ x_n - b_1 & x_n - b_2 & \dots & x_n - b_{n-1} & x_n - b_n \end{pmatrix}$$

A는 Lower-triangular matrix, full Rank N

A의 모든 eigenvalue 는 diagonal elements, 가장 작은 eigenvalue가 diagonal element 중 가장 작은것

Finite-Sample Expressivity

2-layer neural net

"d개의 차원을 가진 n개의 샘플 데이터는
2n+d 개의 weight를 가지고 Activation function 이 ReLU 인 2층의(depth-2) layer를 가진 network 를 통해
어떠한 함수 든 표현 할 수 있다."

Proof of Theorem 1. For weight vectors $w, b \in \mathbb{R}^n$ and $a \in \mathbb{R}^d$, consider the function $c\colon \mathbb{R}^d \to \mathbb{R}$

$$c(x) = \sum_{j=1} w_j \max\{\langle a, x\rangle - b_j, 0\}$$

$$c(X) = \max\left(\underbrace{\begin{bmatrix} --- & x_1 & --- \\ \vdots & \vdots & \vdots \\ --- & x_n & --- \end{bmatrix}}_{n\times d} \underbrace{\begin{bmatrix} | & & | \\ a & \cdots & a \\ | & & | \end{bmatrix}}_{d\times n} - \underbrace{[b_1 \quad \cdots \quad b_n]}_{1\times n}, \underbrace{[0 \quad \cdots \quad 0]}_{1\times n} \right) \cdot \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix}$$

(d, n, n)

Finite-Sample Expressivity

2-layer neural net 에서

$$c(x) = \sum_{j=1} w_j \max\{\langle a, x\rangle - b_j, 0\}$$

$$c(X) = \max\left(\underbrace{\begin{bmatrix} --- & x_1 & --- \\ \vdots & \vdots & \vdots \\ --- & x_n & --- \end{bmatrix}}_{n\times d} \underbrace{\begin{bmatrix} | & & | \\ a & \cdots & a \\ | & & | \end{bmatrix}}_{d\times n} - \underbrace{[b_1 \quad \cdots \quad b_n]}_{1\times n}, \underbrace{[0 \quad \cdots \quad 0]}_{1\times n} \right) \cdot \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix}$$

$z_1, z_2, \ldots z_n$ Sample vector(input data)

$y \in \mathbb{R}^n$ Target vector (예측 변수)

$y_i = c(z_i)$ 를 모두 만족 시키는 w, a, b 를 찾을 수 있다면

이는 모든 샘플 데이터 Z 와 타겟 Y 에 대한 일대일 함수 c를 찾을 수 있다는 것을 말한다.

Finite-Sample Expressivity

$$c(x) = \sum_{j=1} w_j \max\{\langle a, x\rangle - b_j, 0\}$$

$z_1, z_2, \ldots z_n$ Sample vector(input data) $y \in \mathbb{R}^n$ Target vector (예측 변수)

$$y_i = c(z_i)$$

Finite-Sample Expressivity

$$c(x) = \sum_{j=1} w_j \max\{\langle a, x\rangle - b_j, 0\}$$

$z_1, z_2, \ldots z_n$ Sample vector(input data) $y \in \mathbb{R}^n$ Target vector (예측 변수)

$$y_i = c(z_i)$$

$$\langle a, z_i\rangle = x_i$$

$$y = \sum_{j=1} \mathrm{w_j} \max\{x_i - b_j, 0\}$$

Finite-Sample Expressivity

$$c(x) = \sum_{j=1} w_j \max\{\langle a, x\rangle - b_j, 0\}$$

$z_1, z_2, \ldots z_n$ Sample vector(input data)

$y \in \mathbb{R}^n$ Target vector (예측 변수)

$$y_i = c(z_i)$$

$$\langle a, z_i\rangle = x_i \qquad z_i^T a = x_i$$

순서 처리가 가능하다.

$b_1 < x_1 < b_2 < x_2 < b_3 < x_3 \ \ldots \ < b_n < x_n$

$$y = \sum_{j=1} \mathrm{w_j} \max\{x_i - b_j, 0\}$$

Finite-Sample Expressivity

$$y = \sum_{j=1} \mathrm{w_j} \max\{ x_i - b_j, 0 \}$$

$$y = \left[\max\{ x_i - b_j, 0 \}\right]_{ij} w$$

Finite-Sample Expressivity

$$y = \left[\max\{x_i - b_j, 0\}\right]_{ij} w$$

보조 정리 Lemma1.

$$b_1 < x_1 < b_2 < x_2 < b_3 < x_3 \ \ldots < b_n < x_n$$

$$A = [\max\{x_i - b_i, 0\}]_{ij}$$

$$A = \begin{pmatrix} x_1 - b_1 & 0 & \ldots & 0 & 0 \\ x_2 - b_1 & x_2 - b_2 & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ x_{n-1} - b_1 & x_{n-1} - b_2 & \ldots & x_{n-1} - b_{n-1} & 0 \\ x_n - b_1 & x_n - b_2 & \ldots & x_n - b_{n-1} & x_n - b_n \end{pmatrix}$$

$$y = Aw$$

A 는 Invertible

$$A^{-1}y = w$$

Finite-Sample Expressivity

$$A^{-1}y = w$$

모든 데이터 포인트와 target에서
임의의 a, b 에 해당 하는 w 를 항상 찾을 수 있다.

이는 모든 샘플 데이터 $\mathbb{Z}$ 와 타겟 Y 에 대한
일대일 함수 c를 찾을 수 있다는 것을 말한다.

"d개의 차원을 가진 n개의 샘플 데이터는2n+d 개의 weight를 가지고
Activation function 이 ReLU 인 2층의 layer를 가진 network 를 통해
어떠한 함수 든 표현 할 수 있다."

$$c(x) = \sum_{j=1} w_j \max\{\langle a, x\rangle - b_j, 0\}$$

$z_1, z_2, \ldots z_n$: input data

$y \in \mathbb{R}^n$: target

$$y = \sum_{j=1} \mathrm{w_j} \max\{ x_i - b_j, 0 \}$$

$\mathrm{L}emma1$

$$y = Aw$$

Inverse

$$A^{-1}y = w$$

Finite-Sample Expressivity

Universal approximation Theory…

일반적인 통념과는 달리 Universal approximation theorem이 Generalization 을 보장해주는 것은 아니다.

특정 2 layer NN 에 대한 Capability에 대한 증명을 직관적으로 보이도록 발전.

서정훈

Proof

Finite-Sample Expressivity

Model capacity 를 측정할 방법은?
2-layer neural net 으로 Model capacity 를 측정.

Implicit Regularization

Deep learning 모델에 대한 이해도 부족한 듯 보인다.

그렇다면 간단한 선형 모델에 대한 일반화 측면은 어찌 바라 볼 수 있을까?
SGD 를 선형 모델에 적용

Implicit Regularization

SGD 를 선형 모델에 적용 한다면 Regularization 의 효과를 볼 수 있을까

Nonnegative loss function을 가정, 일반적인 선형 모델의 오차 최소화 접근

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \text{loss}(w^T x_i, y_i)$$

X 는 d차원의 feature vector, y는 label

$$Xw = y$$ X 는 n × d 행렬, Rank n

d가 n 보다 크다면, **underdetermined** 다수의 global minima 를 갖게 된다.

이 다수의 global minima 중에 어떤 것이 가장 좋은 **global minima 인지 결정하기가 어렵다.**
또한 위 선형 모델은 Curvature가 모든 data point 마다 같음

Minima 의 quality를 결정 할 수 있는 인덱스가 있다면,

이를 SGD가 찾아 줄 수 있다면?

Implicit Regularization

SGD formula

$$w_{t+1} = w_t - \eta_t e_t x_{it}$$

$$w_0 = 0$$

$$w_1 = 0 - \eta_0 e_0 x_{i0}$$
$$w_2 = w_1 - \eta_1 e_1 x_{i1} = -\eta_0 e_0 x_{i0} - \eta_1 e_1 x_{i1}$$
$$\vdots$$
$$w = \sum_{i=1}^{n} a_i x_i$$

SGD 에서 찾아진 weight (w) 는 data point의 vector 공간에 span 하게 된다.

Implicit Regularization

앞서 가정한 $Xw = y$ 에 $w = \sum_{i=1}^{n} a_i x_i = X^T a$ 를 대입하게 되면

$XX^T a = y$ 이는 SGD에 의하면 a 는 **unique하게 하나로 결정이 된다.**

또한 XX^T 를 일종의 **Kernel matrix** 처럼 사용 할 수 있다.

kernel solution has an appealing interpretation in terms of implicit regularization.
Simple algebra reveals that it is equivalent to the *minimum* l2-*norm* solution of Xw = y

exactly fit the data, SGD will often converge to the solution with **minimum norm**

$Xw = y$ 식에 L2-norm 을 최소화 하는 것과 같다.

Implicit Regularization

$XX^T a = y$ 식을 CIFAR10, MNIST 에 적용

data set	pre-processing	test error
MNIST	none	1.2%
MNIST	gabor filters	0.6%
CIFAR10	none	46%
CIFAR10	random conv-net	17%

L2 Regularization 을 암시적으로 보여 주기 때문에
SGD 는 그 자체로 일반화 능력을 어느정도 내포 하고 있다?

Implicit Regularization

$XX^T a = y$ 식을 CIFAR10, MNIST 에 적용

data set	pre-processing	test error
MNIST	none	1.2%
MNIST	gabor filters	0.6%
CIFAR10	none	46%
CIFAR10	random conv-net	17%

L2 norm : 220

L2 norm : 390

L2 Regularization 을 암시적으로 보여 주기 때문에
SGD 는 그 자체로 일반화 능력을 어느정도 내포 하고 있다?

확실 하게 말 할 수는 없다. 두 데이터 셋을 비교한 경우 L2 norm 이 차이가 있다.

Summary

Finite-Sample Expressivity

2-layer neural net 으로 Model capacity 를 측정.

"d개의 차원을 가진 n개의 샘플 데이터는
2n+d 개의 weight를 가지고 Activation function 이 ReLU 인 2층의 layer를 가진 network 를 통해
어떠한 함수 든 표현 할 수 있다." (모두 표현이 가능하다)

Deep learning 의 large한 capacity 에 대해 입증

Implicit Regularization

Implicit 한 Regularization 인 SGD 를 선형 모델에 적용 generalization을 할 수 있게 도와 준다.

하지만 완전히 맞지는 않다.

Contents

UNDERSTANDING DEEP LEARNING REQUIRES RETHINKING GENERALIZATION

Can the traditional theories on generalization **explain the results we are seeing these days?** **Finite-Sample Expressivity, Deep Neural Net optimize easy..**

What is it then that **distinguishes** neural networks that generalize well from those that don't? **Not sure..**

Why large neural networks generalize well in practice?

RETHINKING GENERALIZATION

Reference

출처

https://arxiv.org/pdf/1611.03530.pdf (Paper)
https://www.slideshare.net/JungHoonSeo2/understanding-deep-learning-requires-rethinking-generalization-2017-12 (Slide share , JungHoon Seo)
https://www.slideshare.net/thinkingfactory/pr12-understanding-deep-learning-requires-rethinking-generalization. (Silde share , JaeJun Yoo)
https://www.youtube.com/watch?v=UxJNG7ENRNg&t=784s (Youtube, JaeHun Yoo)
https://danieltakeshi.github.io/2017/05/19/understanding-deep-learning-requires-rethinking-generalization-my-thoughts-and-notes (Blogs,