Theory of Deep Convolutional Neural Networks

Ding-Xuan Zhou

School of Data Science
Department of Mathematics
Liu Bie Ju Centre for Mathematical Sciences
City University of Hong Kong
E-mail: mazhou@cityu.edu.hk

Supported in part by Research Grants Council of Hong Kong

Start

November 18, 2020

Outline of the talk

I. Deep learning and deep neural networks

II. Least squares regression and ERM

III. Classical theory of fully connected neural networks

IV. Theory of deep CNNs

V. Superiority of deep CNNs

VI. Theory of deep CNNs induced by 2-D convolutions

I. Deep learning and deep neural networks

I.1. Applications of deep learning

Big data leads to scientific challenges:
storage bottleneck, algorithmic scalability, ...

Applications of deep learning
Speech Recognition: from hidden Markov models and Gaussian mixture models to restricted Boltzmann machines and end-to-end deep learning speech recognition systems, phoneme error rate brought down from 26% to 17.7%

Computer Vision: Google Street View, autonomous driving, ...

Natural Language Processing, Reinforcement Learning, ...

Theory of deep learning: at its infancy

I.2. Deep neural network architectures

Convolutional neural networks (CNNs) mainly for speech and image applications: LeCun-Bottou-Bengio-Haffner, Hinton-Osindero-Teh, Bengio, LeCun, Krizhevsky-Sutskever-Hinton, Simonyan-Zisserman, Mallat, ...

Recurrent neural networks: Szegedy-Liu-Jia-Sermanet-Reed-Anguelov-Erhan-Vanhoucke-Rabinovich, ...

Deep belief networks: no connection within layers: deep Boltzmann machines

Connections to **generative adversarial networks** (GANs)

I.3. Examples of deep CNNs in computer vision

ImageNet Large Scale Visual Recognition Challenge winners:

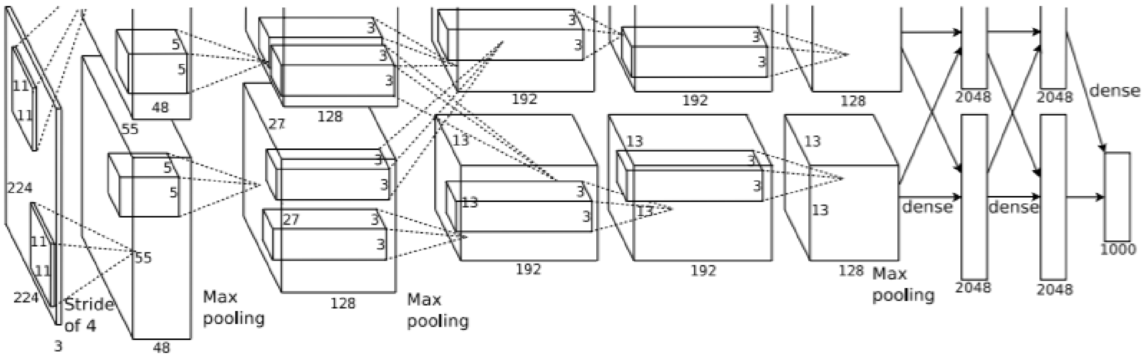
AlexNet (Krizhevsky-Sutskever-Hinton 2012 winner: 16.4% top 5 error):
5 layers of CNNs with $s = 10, 4, 2, 2, 2$ and 3 layers of fully-connected nets with polling and normal layers

ZFNet (Zeiler-Fergus 2013 winner: 11.7% top 5 error):
8 layers improving AlexNet

VGGNet (Simonyan-Zisserman 2014: 7.3% top 5 error):
VGG16 with 12 layers of CNNs with $s = 2$ and 4 layers of fully-connected nets with polling, 138M parameters
VGG19 with 19 layers (slightly better)

GoogLeNet (Szegedy-Liu-Jia-Sermanet-Reed-Anguelov-Erhan-Vanhoucke-Rabinovich 2014 winner: 6.7% top 5 error):
Combination of CNNs and recursive nets (inception module), 22 layers of CNNs, no fully-connected layer, 5M parameters

ResNet (He-Zhang-Ren-Sun 2015 winner: 3.57% top 5 error):
152 layers using residual connections based on many CNNs with $s = 2$ and 2 layers of fully-connected nets


224
224
11
11
11
11
Stride of 4
3
55
5
5
5
5
48
48
55
Max pooling
27
27
128
128
3
3
3
3
3
3
3
3
Max pooling
13
13
192
192
3
3
3
3
13
13
192
192
3
3
3
3
13
13
128
128
Max pooling
dense
dense
dense
2048
2048
2048
2048
1000

II. Least squares regression and ERM

II.1. Model for the least squares regression. Learn $f : \mathcal{X} \to \mathcal{Y}$ from a random sample $\mathbf{z} = \{(x_i, y_i)\}_{i=1}^m$

Take $\mathcal{X}$ to be a compact metric space and $\mathcal{Y} = \mathbf{R}$. $y \approx f(x)$ Due to noises or other uncertainty, we assume a (unknown) probability measure ρ on $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ governs the sampling.

marginal distribution ρ_X on $\mathcal{X}$: $\mathbf{x} = \{x_i\}_{i=1}^m$ drawn from ρ_X

conditional distribution $\rho(\cdot|x)$ at $x \in \mathcal{X}$

Learning the **regression function**: $f_\rho(x) = \int_{\mathcal{Y}} y d\rho(y|x)$

$y_i \approx f_\rho(x_i)$

II.2. Least squares generalization error

$\mathcal{E}(f) = \int_{\mathcal{Z}} (f(x) - y)^2 d\rho$ minimized by f_ρ:

$$\mathcal{E}(f) - \mathcal{E}(f_\rho) = \|f - f_\rho\|^2_{L^2_{\rho_X}} =: \|f - f_\rho\|^2_{\rho_X} \geq 0.$$

Classical Approach of Empirical Risk Minimization (ERM)

Let $\mathcal{H}$ be a compact subset of $C(\mathcal{X})$ called **hypothesis space**. The ERM algorithm is given by

$$f_{\mathbf{z}} = \arg\min_{f \in \mathcal{H}} \mathcal{E}_{\mathbf{z}}(f), \qquad \mathcal{E}_{\mathbf{z}}(f) = \frac{1}{m} \sum_{i=1}^{m} (f(x_i) - y_i)^2.$$

Target function $f_{\mathcal{H}}$: best approximation of f_ρ in $\mathcal{H}$

$$f_{\mathcal{H}} = \arg\min_{f \in \mathcal{H}} \mathcal{E}(f) = \arg\inf_{f \in \mathcal{H}} \int_{\mathcal{Z}} (f(x) - y)^2 d\rho.$$

II.3. Approximation error

Analysis. $\|f_{\mathbf{z}} - f_\rho\|^2_{\rho_X} = \int_{\mathcal{X}} (f_{\mathbf{z}}(x) - f_\rho(x))^2 d\rho_X$ is bounded by $2 \sup_{f \in \mathcal{H}} |\mathcal{E}_{\mathbf{z}}(f) - \mathcal{E}(f)| + \left\{ \mathcal{E}(f_{\mathcal{H}}) - \mathcal{E}(f_\rho) \right\}$.

Approximation Error. Smale-Zhou (2003)

$$\mathcal{E}(f_{\mathcal{H}}) - \mathcal{E}(f_\rho) = \|f_{\mathcal{H}} - f_\rho\|^2_{\rho_X} = \inf_{f \in \mathcal{H}} \int (f(x) - f_\rho(x))^2 d\rho_X$$

$f_{\mathcal{H}} \approx f_\rho$ when $\mathcal{H}$ is rich

Theorem 1. *Let B be a Banach space (such as a Sobolev space or a reproducing kernel Hilbert space). If $B \subset L^2_{\rho_X}$ is dense and $\theta > 0$, then*

$$\inf_{\|f\|_B \le R} \|f - f_\rho\|_{\rho_X} = O(R^{-\theta})$$

if and only if f_ρ lies in the interpolation space $(B, L^2_{\rho_X})_{\frac{\theta}{1+\theta}, \infty}$.

II.4. Sample error

Uniform Law of Large Numbers: Theory of Uniform Convergence for bounding $\sup_{f\in\mathcal{H}}\left|\frac{1}{m}\sum_{i=1}^m f(x_i) - \int_{\mathcal{X}} f(x)d\rho\right|$.

$$\lim_{\ell\to\infty}\sup_{\rho}\mathrm{Prob}\left\{\sup_{m\geq\ell}\sup_{f\in\mathcal{H}}\left|\frac{1}{m}\sum_{i=1}^m f(x_i) - \int_{\mathcal{X}} f(x)d\rho\right| > \epsilon\right\} = 0?$$

Characterizations by VC dimensions: Vapnik-Ch. (1971), V_γ dimensions: Alon, Ben-David, Cesa-Bianchi, Haussler (1997) Quantitative estimates by means of covering numbers when $\mathcal{H}\subset C(\mathcal{X})$: Cucker-Smale, Zhou, Ying-Zhou, ...

Special hypothesis spaces for kernel methods:
Reproducing Kernel Hilbert Spaces $\mathcal{H}_K$

III. Classical theory of fully connected neural networks

III.1. Hypothesis space by deep fully connected networks

Classical **shallow fully connected neural networks** on $\mathbb{R}^d$:

$$f_{T^{(1)},b,c}(x) = \sum_{i=1}^{N} c_i \sigma(\mathbf{w}_i \cdot x + b_i) = c^T \sigma(T^{(1)} x + b),$$

where $x \in \mathbb{R}^d$, activation $\sigma : \mathbb{R} \to \mathbb{R}$ acts componentwise

$$T^{(1)} = [\mathbf{w}_1, \ldots, \mathbf{w}_N]^T = \begin{bmatrix} \mathbf{w}_1^T \\ \vdots \\ \mathbf{w}_N^T \end{bmatrix} \in \mathbb{R}^{N \times d}, \ b, \ c \in \mathbb{R}^N$$

Most commonly used activation function in deep learning: Rectified linear unit **ReLU** given by

$$\sigma(u) = (u)_+ = \max\{u, 0\}$$

Multi-layer neural networks with J hidden layers of neurons $\{h^{(j)} : \mathbb{R}^d \to \mathbb{R}^{d_j}\}_{j=1}^J$ with widths $\{d_j \in \mathbb{N}\}$ and $h^{(0)}(x) = x$

$$h^{(j)}(x) = \sigma\left(T^{(j)} h^{(j-1)}(x) + b^{(j)}\right), \qquad j = 1, \ldots, J$$

with **full connection matrices** $T^{(j)} \in \mathbb{R}^{d_j \times d_{j-1}}$
bias vectors $b^{(j)} \in \mathbb{R}^{d_j}$

Hypothesis space $\mathcal{H}$:
$\left\{f_{\mathbf{T},\mathbf{b},c}(x) = c^T h^{(J)}(x) : \mathbf{T} = (T^{(j)})_{j=1}^J, \mathbf{b} = (b^{(j)})_{j=1}^J, c \in \mathbb{R}^{d_J}\right\}$

Large literature on local or global convergence of Stochastic Gradient Descent for

$$f_{\mathbf{z}} = \arg\min_{\mathbf{T},\mathbf{b},c} \mathcal{E}_{\mathbf{z}}(f_{\mathbf{T},\mathbf{b},c}) := \frac{1}{m}\sum_{i=1}^m (f_{\mathbf{T},\mathbf{b},c}(x_i) - y_i)^2$$

Total number of **free parameters** to compute:
$\mathcal{N} = \sum_{j=1}^{J} \left(d_j d_{j-1} + d_j \right) + d_J$, increases very fast when the data dimension d or the depth J is large

Sample error estimates with norm bounds for $\mathbf{T}, \mathbf{b}, c$: Kohler-Krzyżak, Schmidt-Hieber, ...

Literature on representational complexity: Goodfellow-Bengio-Courville, Montúfar-Pascanu-Cho-Bengio, Delalleau-Bengio, Poggio-Anselmi-Rosasco, ...

III.2. Approximation theory of fully connected networks

Universality of approximation: If σ is locally bounded, piecewise continuous, and not a polynomial, then for any compact $\Omega \subset \mathbb{R}^d$ and $f \in C(\Omega)$, there holds

$$\lim_{N\to\infty} \inf \left\{ \|f - f_{T^{(1)},b^{(1)},c}\|_{C(\Omega)} : \ T^{(1)} \in \mathbb{R}^{N\times d}, b^{(1)}, c \in \mathbb{R}^N \right\} = 0.$$

Classical theory: Cybenko (1989), Hornik (1991), Barron (1993), Mhaskar (1993), Mhaskar-Micchelli (1994), Chui-Li-Mhaskar, Lin-Pinkus-Schocken, Maiorov, Petrushev, ...

Typical result (Mhaskar 1993): Assume a continuous activation functions σ satisfies with some $b \in \mathbb{R}$, $\sigma^{(k)}(b) \neq 0$ for any $k \in \mathbb{Z}_+$ and with some integer $\ell \neq 1$, $\lim_{u\to-\infty} \sigma(u)/|u|^\ell = 0$ and $\lim_{u\to\infty} \sigma(u)/u^\ell = 1$. Then for $f \in C^r([-1,1]^d)$,

$$\inf \left\{ \|f - f_N\|_{C(\Omega)} : \ \mathbf{w}_i, b, c \right\} = O(N^{-r/d}) = O\left((d+2)^{r/d} N^{-r/d} \right).$$

ReLU with $\ell = 1$ was not included, and recent work of Klusowski-Barron (2018), Shaham-Cloningen-Coifman (2018), Gühring-Kutyniok-Petersen (2020) covers this case.

III.3. Recent work on deep fully connected ReLU nets

Eldan-Shamir (2016), Telgarsky (2016), Yarotsky (2017), Klusowski-Barron (2018), Shaham-Cloningen-Coifman (2018), Bölcskei-Grohs-Kutyniok-Petersen (2019), Petersen-Voigtlaender (2018), McCane-Szymanski (2017), Chui-Lin-Zhou (2019), Grohs-Perekrestenko-Elbrächter-Bölcskei (2019), ...

approximating compositional functions: Mhaskar-Poggio (2016), Poggio-Mhaskar-Rosasco-Miranda-Liao (2016), ...

wavelet and kernel methods for deep CNNs: Bruna-Mallat (2013), Mallat (2016), Steinwart-Thomann-Schmid (2016), ...

Little is known on modelling, approximation or generalization ability of deep **structured** neural networks (not fully connected):

$$\min_{\mathbf{T},\mathbf{b},c} \|f_{\mathbf{T},\mathbf{b},c} - f_\rho\|_{\rho_X}$$

IV. Theory of deep CNNs

IV.1. Convolutions and convolutional matrices

The convolution of two sequences $w = (w_k)_{k=-\infty}^{\infty}$ and $x = (x_k)_{k=-\infty}^{\infty}$ is another sequence $w*x$ given by

$$(w*x)_i = \sum_{k=-\infty}^{\infty} w_{i-k} x_k, \qquad i \in \mathbb{Z}.$$

If w is supported in $\{0, 1, \ldots, s\}$ and x is supported in $\{1, \ldots, d\}$, then $w*x$ is supported in $\{1, \ldots, d+s\}$ and

$$[(w*x)_i]_{i=1}^{d+s} = T^w \, [x_k]_{k=1}^{d}$$

where $T^w = (w_{i-k})_{1 \leq i \leq d+s, 1 \leq k \leq d}$ is a Toeplitz type $(d+s) \times d$ convolutional matrix.

Zero-padding: if x is unknown or unavailable outside $\{1,\dots,d\}$ we take zero terms outside: for $= 1,\dots,d+s$,

$$(w{*}x)_i = \sum_{k=0}^{s} w_k x_{i-k} = w_s x_{i-s} + w_{s-1}x_{i-s+1} + \ldots + w_0 x_i.$$

This leads to the convolutional matrix with zero-padding $T^w = (w_{i-k})_{1\le i\le d+s,1\le k\le d}$:

$$\begin{bmatrix} (w{*}x)_1 \\ (w{*}x)_2 \\ \vdots \\ (w{*}x)_{i-1} \\ (\mathbf{w{*}x})_{\mathbf{i}} \\ (w{*}x)_{i+1} \\ \vdots \\ (w{*}x)_d \\ (w{*}x)_{d+1} \\ \vdots \\ (w{*}x)_{d+s} \end{bmatrix} = \begin{bmatrix} w_0 & 0 & 0 & \cdots & \cdots & 0 \\ w_1 & w_0 & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ w_s & \cdots & \cdots & \ddots & \cdots & 0 \\ 0 & w_s & \cdots & w_0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & w_1 & w_0 \\ 0 & \cdots & 0 & w_s & \cdots & w_1 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & w_s \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{i-s-1} \\ \mathbf{x_{i-s}} \\ \vdots \\ \mathbf{x_i} \\ x_{i+1} \\ \vdots \\ x_d \end{bmatrix}.$$

IV.2. Deep CNNs induced by 1-D convolutions: time-delay neural networks (1989) and shift invariant neural networks (1990)

sequence of convolutional filters $\mathbf{w} = \{w^{(j)} = (w_k^{(j)})_{k=-\infty}^{\infty}\}_{j=1}^{J}$
filter length s: Assume $w^{(j)}$ is supported in $\{0, 1, \ldots, s\}$

A **deep CNN of depth** J is a J-layer neural network with widths $\{d_j = d + js\}$ having a **convolutional sparse structure**:

$$h^{(j)}(x) = \sigma\left(T^{(j)} h^{(j-1)}(x) - b^{(j)}\right), \qquad j = 1, 2, \ldots, J, \tag{1}$$

where $T^{(j)} = T^{w^{(j)}}$ is a Toeplitz type $d_j \times d_{j-1}$ matrix generated by the filter mask $w = w^{(j)}$.

Take $b^{(j)}$ of the form $b = [b_1 \ \ldots b_{s-1} \ b_s \ b_s \ \ldots \ b_s \ b_{d_j-s+1} \ \ldots b_{d_j}]^T$

free parameters:

filter sequence $\mathbf{w} = (w^{(j)})_{j=1}^{J}$, bias vector sequence $\mathbf{b} = \{b^{(j)}\}_{j=1}^{J}$

The generated **hypothesis space** of functions:

$$\cup_{\mathbf{w},\mathbf{b}} \mathcal{H}_J^{\mathbf{w},\mathbf{b}} := \left\{ \sum_{k=1}^{d+Js} c_k h_k^{(J)}(x) : c \in \mathbb{R}^{d+Js} \right\}. \qquad (2)$$

Total number of free parameters: $\mathcal{N} = 3s(J-1) + 2d + 2J$

IV.3. Universality of deep CNNs

Theorem 2. *Let* $2 \leq s \leq d$. *For any compact subset* Ω *of* $\mathbb{R}^d$ *and any* $f \in C(\Omega)$, *there exist sequences* $\mathbf{w}$ *of filters,* $\mathbf{b}$ *of bias vectors, and* $f_J^{\mathbf{w},\mathbf{b}} \in \mathcal{H}_J^{\mathbf{w},\mathbf{b}}$ *such that* $\lim_{J\to\infty} \|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} = 0.$

Key observation

If $W = (W_k)_{k=-\infty}^{\infty}$ is supported in $\{0, \ldots, \mathcal{M}\}$, then there exist filters $\{w^{(j)}\}_{j=1}^{p}$ each supported in $\{0, \ldots, s\}$ with $p \leq \lceil \frac{\mathcal{M}}{s-1} \rceil$ satisfying a convolutional factorization

$$W = w^{(p)} * w^{(p-1)} * \ldots * w^{(2)} * w^{(1)}.$$

IV.4. Rates of approximation

Sobolev space $H^r(\mathbb{R}^d)$: F and all its partial derivatives up to order r are square integrable on $\mathbb{R}^d$.
Embedding Theorem: $H^r(\mathbb{R}^d) \subset C(\mathbb{R}^d)$ only when $r > \frac{d}{2}$.
Theorem 3. *If $\Omega \subseteq [-1,1]^d$ and f is the restriction to Ω of some function $F \in H^r(\mathbb{R}^d)$ on $\mathbb{R}^d$ with integer $r > 2 + \frac{d}{2}$ and $J \geq 2d/(s-1)$, then there exist $\mathbf{w}$, $\mathbf{b}$, and $f_J^{\mathbf{w},\mathbf{b}} \in \mathcal{H}_J^{\mathbf{w},\mathbf{b}}$ such that*

$$\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} \leq \tilde{c}\, \|F\|_{H^r} \sqrt{\log J}\, (1/J)^{\frac{1}{2}+\frac{1}{d}},$$

where $\tilde{c}$ is an absolute constant and $\|F\|_{H^r}$ denotes the Sobolev space norm of F.

$\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} = O\left(1/\sqrt{J}\right)$ where J is the depth

Specialty of zero-padding: linearly increasing widths $\{d_j = d + js\}$

Corollary. *Take* $s = \lceil 1 + d^\tau/2 \rceil$ *and* $J = \lceil 4d^{1-\tau} \rceil L$ *with* $0 \leq \tau \leq 1$ *and* $L \in \mathbb{N}$. *Under the assumption of the theorem, we have*

$$\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} \leq c\, \|F\|_{H^r} \sqrt{\frac{(1-\tau)\log d + \log L + \log 5}{4d^{1-\tau}L}},$$

while the widths of the deep CNN are bounded by $12Ld$ *and the total number of free parameters by*

$$5sJ + 2d - 2s + 1 \leq (65L + 2)d.$$

We can even take $L = 1$, $\tau = 1/2$, *filter length* $s = \lceil 1 + \sqrt{d}/2 \rceil$, *and depth* $\lceil 4\sqrt{d} \rceil$ *to get a bound for the relative error*

$$\frac{\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)}}{\|F\|_{H^r}} \leq c \sqrt{\frac{\log(5\sqrt{d})}{4\sqrt{d}}},$$

achieved by a deep CNN of at most $67d$ *free parameters, which decreases as the dimension* d *increases.*

IV.5. Comparisons

Take $s = \lceil 1 + d/2 \rceil$. An accuracy $\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} \leq \epsilon$ with $0 < \epsilon \leq c\|F\|_{H^r}$ can be achieved by the deep CNN of depth $J = 4\lceil \frac{1}{\epsilon^2} \log \frac{1}{\epsilon^2} \rceil$ having at most $\lceil \frac{75}{\epsilon^2} \log \frac{1}{\epsilon^2} \rceil d$ free parameters.

Telgarsky (2016), Yarotsky (2017): an accuracy $\epsilon \in (0,1)$ can be achieved by a ReLU deep fully-connected net with at most $c(\log(1/\epsilon) + 1)$ layers and at most $c\epsilon^{-d/r}(\log(1/\epsilon) + 1)$ free parameters with $c = c(d, r)$. But the net needs at least $\frac{C_0 d}{4}(\log(1/\epsilon) + d)$ layers and at least $2^d \epsilon^{-d/r}$ free parameters.

Bölcskei-Grohs-Kutyniok-Petersen (2019), Petersen-Voigtlaender (2018), ...: the rate of approximation of some function classes may be achieved by networks with sparse connection matrices, but the locations of the sparse connections are unknown. This sparsity of unknown pattern is totally different from that of deep CNNs.

V. Superiority of deep CNNs

The downsampling operator $\mathcal{D}_m : \mathbb{R}^D \to \mathbb{R}^{[D/m]}$ with a scaling parameter $m \leq D$ is defined by

$$\mathcal{D}_m(v) = (v_{im})_{i=1}^{[D/m]}, \qquad v \in \mathbb{R}^D.$$

A **downsampled** deep CNN with ℓ downsamplings at layers $\mathcal{J} := \{J_k\}_{k=1}^{\ell}$ with $1 < J_1 \leq J_2 \leq \ldots \leq J_\ell = J$ and filter lengths $\{s^{[k]}\}_{k=1}^{\ell}$ has for $k = 1, \ldots, \ell$ widths $\{d_j\}_{j=0}^{J}$ as

$$d_j = \begin{cases} d_{j-1} + s^{[k]}, & \text{if } J_{k-1} < j < J_k, \\ \left[(d_{j-1} + s^{[k]})/d_{J_{k-1}}\right], & \text{if } j = J_k, \end{cases}$$

and $\left\{h^{(j)}(x) : \mathbb{R}^d \to \mathbb{R}^{d_j}\right\}_{j=1}^{J}$ given iteratively by (1) for $J_{k-1} < j < J_k$ and for $j = J_k$ by

$$h^{(j)}(x) = \mathcal{D}_{d_{J_{k-1}}}\left(\sigma\left(T^{(j)} h^{(j-1)}(x) - b^{(j)}\right)\right).$$

V.1. Achieving fully connected nets by deep CNNs

Theorem 4. *Let $\{H^{(k)} : \mathbb{R}^d \to \mathbb{R}^{n_k}\}_{k=1}^{\ell}$ be an ℓ-layer fully connected neural network such that $n_k n_{k-1} > 1$ for each $k \in \{1, \ldots, \ell\}$. Let $s^{[k]} \in [2, n_k n_{k-1}]$ for each k. Then there is a downsampled deep CNN $\left\{h^{(j)}(x) : \mathbb{R}^d \to \mathbb{R}^{d_j}\right\}_{j=1}^{J}$ with ℓ downsamplings at layers $\{J_k = \sum_{j=1}^{k} \Delta_j\}$ with $\Delta_j \leq \lceil \frac{n_j n_{j-1} - 1}{s^{[j]} - 1} \rceil$ for each j such that*

$$h^{(J_k)}(x) = H^{(k)}(x), \qquad \forall k \in \{1, \ldots, \ell\}, x \in \Omega. \tag{3}$$

The total number of free parameters in the above net is at most $8 \sum_{k=1}^{\ell} (n_k n_{k-1})$ and is at most 8 times of that of the fully-connected net.

V.2. Superiority in approximating ridge functions

Ridge function $g(\xi \cdot x)$ with unknown $\xi \in \mathbb{R}^d$ and $g : \mathbb{R} \to \mathbb{R}$.

Theorem 5. *Let* $\Omega \subseteq \mathbb{B} := \{x \in \mathbb{R}^d : |x| \leq 1\}$ *and* $\xi \in \mathbb{B}$, $2 \leq s \leq d$, *and* $N \in \mathbb{N}$. *If* $g : [-1, 1] \to \mathbb{R}$ *is Lipschitz-*α *for some* $0 < \alpha \leq 1$, *then there exists a downsampled deep CNN* $\left\{h^{(j)}(x)\right\}_{j=1}^{J}$ *at layers* $\mathcal{J} = \{J_1 \leq \lceil \frac{d-1}{s-1} \rceil, J = J_1 + 1\}$, *of widths* $d_j = d + js$ *for* $j = 0, 1, \ldots, J_1 - 1$, $d_j = 1$ *or* 2 *for* $j = J_1$ *and* $d_j = 2N + 4$ *for* $j = J$, *of filter lengths* $\mathcal{S} = \{s, 4N + 6\}$ *with* $\{w_i^{(J)} \equiv 1\}_{i=0}^{4N+6}$, *and* $b^{(J)}$ *given in terms of* N *and a parameter* $B^{(J-1)}$, *and coefficients* $\{c_i\}_{i=1}^{2N+3}$ *such that*

$$\left\| \sum_{i=1}^{2N+3} c_i \left(h^{(J)}(x)\right)_i - g(\xi \cdot x) \right\|_{\infty} \leq 2|g|_{Lip\ \alpha} N^{-\alpha}. \qquad (4)$$

To achieve the approximation accuracy $\epsilon \in (0, 1)$, *we take* $N = \lceil \left(2|g|_{Lip\ \alpha}/\epsilon\right)^{1/\alpha} \rceil$ *and require at most* $9d + 2\left(2|g|_{Lip\ \alpha}/\epsilon\right)^{1/\alpha} + 5$ *free parameters.*

V.3. Superiority in approximating radial functions

Consider $\mathbb{B} := \{x \in \mathbb{R}^d : |x| \leq 1\}$ and the space $C^{0,1}(\mathbb{B})$ of Lipschitz functions with norm $\|g\|_{C^{0,1}(\mathbb{B})} = \sup_{x \neq y \in \mathbb{B}} \frac{|g(x)-g(y)|}{|x-y|} + \sup_{x \in \mathbb{B}} |g(x)|$. Consider the set of radial functions

$$\mathcal{B}\left(C^{0,1}_{|\cdot|}\right) := \left\{f(|\cdot|^2) : \|f(|\cdot|^2)\|_{C^{0,1}(\mathbb{B})} \leq 1\right\}.$$

Denote the span of N ridge functions as

$$\mathcal{S}_N = \left\{\sum_{k=1}^{N} c_k \sigma_k(a_k \cdot x - b_k) : \ \sigma_k \in C(\mathbb{R}), \ a_k \in \mathbb{R}^d, \ c_k, b_k \in \mathbb{R}\right\}.$$

The hypothesis space generated by a shallow neural network is a subset of $\mathcal{S}_N$ (with $\sigma_1 = \ldots = \sigma_N$).

The efficiency of a hypothesis space V in approximating a function set U is measured by the quantity

$$\mathrm{dist}(U, V) := \sup_{f \in U} \inf_{g \in V} \|f - g\|_{L_\infty(\mathbb{B})}.$$

Theorem 6. *Let $2 \le s \le d$. We have*

$$\text{dist}\left(\mathcal{B}\left(C^{0,1}_{|\cdot|}\right), \mathcal{S}_N\right) \ge c_d N^{-\frac{1}{d-1}}, \qquad \forall N \in \mathbb{N}$$

with a constant c_d independent of N; while

$$\text{dist}\left(\mathcal{B}\left(C^{0,1}_{|\cdot|}\right), \mathcal{H}_N\right) \le 3\sqrt{1+4d}N^{-\frac{1}{2}}, \qquad \forall N \in \mathbb{N}$$

achieved by the hypothesis space $\mathcal{H}_N$ generated by a deep CNN $\{h^{(j)} : \mathbb{R}^d \to \mathbb{R}^{d_j}\}_{j=1}^J$ of depth $J = \left\lceil \frac{(2N+3)d}{s-1} \right\rceil$ and widths $\{d_j = d + js\}_{j=1}^J$ followed by a fully connected layer of widths $2N+3$.

By Theorem 6, the total number of free parameters in our deep CNN network for achieving an accuracy $\epsilon > 0$ in approximating functions from the class $\mathcal{B}\left(C^{0,1}_{|\cdot|}\right)$ is $O(\epsilon^{-2})$ while that of a fully connected shallow network is $O(\epsilon^{-(d-1)})$. This shows that deep neural networks are much more efficient than shallow networks in approximating radial functions when the dimension $d > 3$ is large.

VI. Theory of deep CNNs induced by 2-D convolutions

VI.1. Structure of 2-D convolutions

1-D convolution: when x outside $[1, d]$ is unavailable or unknown, take those terms $(w*x)_i = \sum_{k\in\mathbb{Z}} w_{i-k}x_k$ involving $\{x_k\}_{k=1}^{d}$ only, restricting $i \in \{s+1, \ldots, d\}$.

Convolutional matrix without zero-padding:

$$\mathcal{C}^w = [w_{i-k}]_{s+1\le i\le d,\ 1\le k\le d} = \begin{bmatrix} w_s & \cdots & \cdots w_0 & 0 \cdots & 0 \\ 0 & \ddots\ \ddots & \ddots & \ddots\ \ddots & \vdots \\ \vdots & 0 \cdots & w_s & \cdots & w_0 \end{bmatrix}.$$

If a 2-D filter $W = (W_\alpha)_{\alpha\in\mathbb{Z}^2}$ is supported in $\{0, \ldots, s\}^2$ and a digital image X is supported in $\{0, \ldots, d\}^2$, then the 2-D convolution without zero-padding induces a tensor $\mathcal{T}^W$, mapping a digital image $X : \{0, \ldots, d\}^2 \to \mathbb{R}$ to another $\mathcal{T}^W X : \{s, \ldots, d\}^2 \to \mathbb{R}$ given by

$$\left(\mathcal{T}^W X\right)_\alpha = (W*X)_\alpha = \sum_{\beta\in[0,s]^2} W_\beta X_{\alpha-\beta}, \qquad \alpha \in \{s, \ldots, d\}^2.$$

VI.2. Deep tensor product convolutional neural networks

When the 2-D filter W is given by the tensor product of two 1-D filters u and v supported in $\{0, 1, \ldots, s\}$ as

$$W = u \otimes v, \text{that is, } W_{(\beta_1,\beta_2)} = u_{\beta_1} v_{\beta_2} \quad \text{for } \beta = (\beta_1, \beta_2) \in \mathbb{Z}^2,$$

then $\mathcal{T}^W X = \mathcal{C}^{u^{(j)}} X (\mathcal{C}^{v^{(j)}})^T$.

A **deep tensor product convolutional neural network** in 2-D of depth $J = d/s$ is a neural network $\{h^{(j)} : \mathbb{R}^{\{0,\ldots,d\}^2} \to \mathbb{R}^{\{js,\ldots,d\}^2}\}_{j=1}^J$ defined iteratively by $h^{(0)}(X) = X \in \mathbb{R}^{\{0,\ldots,d\}^2}$ and

$$h^{(j)}(X) = \sigma\left(\mathcal{L}^{(j)} h^{(j-1)}(X) \mathcal{R}^{(j)} + B^{(j)}\right), \qquad j = 1, \ldots, J,$$

with a bias matrix $B^{(j)} \in \mathbb{R}^{\{js,\ldots,d\}^2}$ and $\mathcal{L}^{(j)} = \mathcal{C}^{u^{(j)}}, \mathcal{R}^{(j)} = (\mathcal{C}^{v^{(j)}})^T$ induced by 1-D convolutional matrices associated with pairs $\{u^{(j)}, v^{(j)}\}$ of filters on $\mathbb{Z}$ supported in $\{0, \ldots, s\}$.

VI.3. Theory of deep CNNs induced by 2-D convolutions

Assume that s is even, d/s is an integer and take $J = d/s$.

Theorem 7. *For any compact subset Ω of $\mathbb{R}^{(d+1)\times(d+1)}$ and any $U, V \in \mathbb{R}^{d+1}, b \in \mathbb{R}$, there exist sequences $\mathbf{u} = \{u^{(j)}\}_{j=1}^{J}, \mathbf{v} = \{v^{(j)}\}_{j=1}^{J}$ of filters on $\mathbb{Z}$ supported in $\{0, 1, \ldots, s\}$ and numbers $\mathbf{b} = \{b^{(j)} \in \mathbb{R}\}_{j=1}^{J}$ such that the deep tensor product convolutional neural network in 2-D of depth J generated with $\{B^{(j)} = b^{(j)}\mathbf{1}_{(d-js+1)\times(d-js+1)}\}$ has the output layer*

$$h^{(J)}(X) = \sigma(U^T X V + b), \qquad \forall X \in \Omega.$$

More theorems for image classification: deep tensor product convolutional neural networks in 2-D with m channels followed by one fully-connected layer $H : \mathbb{R}^m \to \mathbb{R}$ of width N can approximate functions of type $f\left(\left(\left(U^{[i]}\right)^T X V^{[i]}\right)_{i=1}^{m}\right)$ with vectors $U^{[i]}, V^{[i]} \in \mathbb{R}^{d+1}$ and a function $f : \mathbb{R}^m \to \mathbb{R}$ by orders $O(1/\sqrt{N})$.

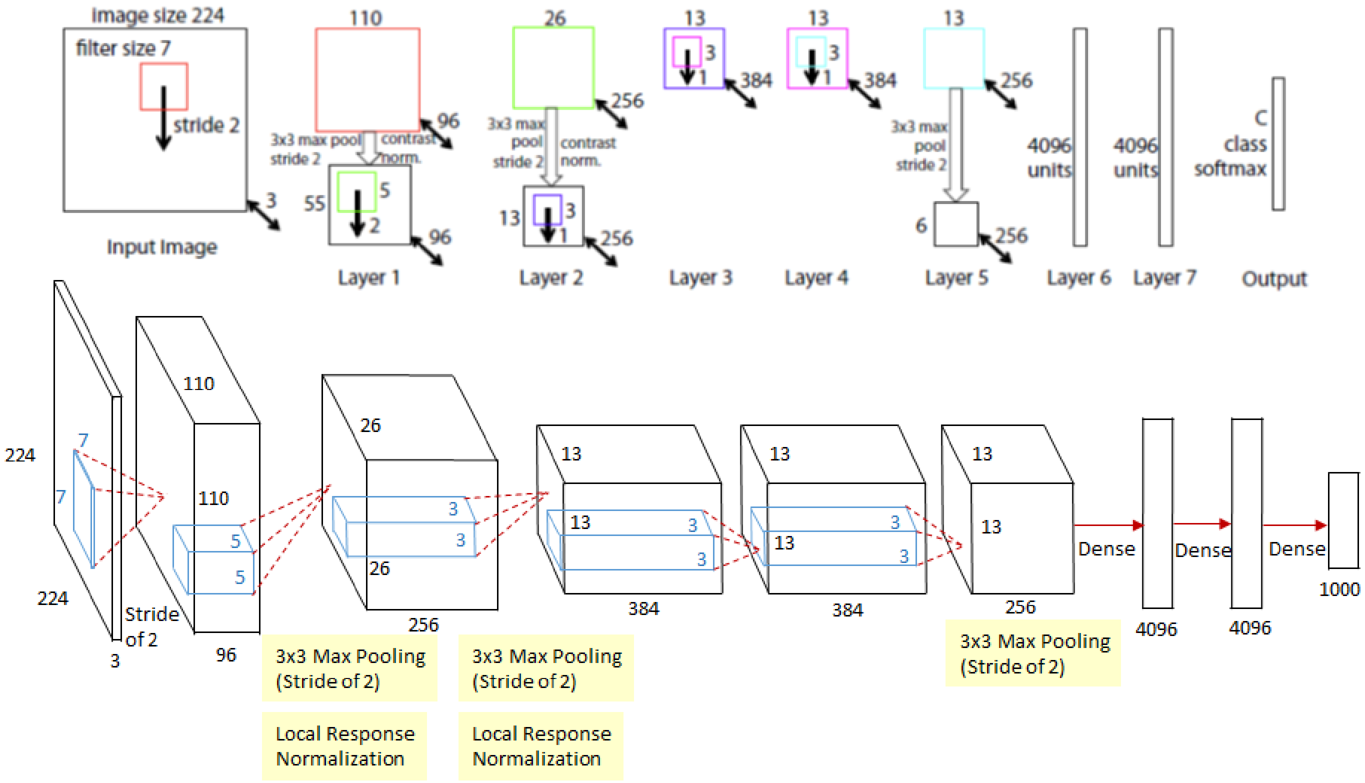


image size 224
filter size 7
stride 2
3
Input Image
110
3x3 max pool stride 2
contrast norm.
96
55
5
2
96
Layer 1
26
3x3 max pool stride 2
contrast norm.
256
13
3
1
256
Layer 2
13
3
1
384
Layer 3
13
3
1
384
Layer 4
13
3x3 max pool stride 2
256
6
256
Layer 5
4096 units
Layer 6
4096 units
Layer 7
C class softmax
Output
224
7
7
224
3
Stride of 2
110
110
5
5
96
26
3
3
26
256
13
3
13
3
384
13
3
13
3
384
13
13
256
Dense
4096
Dense
4096
Dense
1000
3x3 Max Pooling (Stride of 2)
Local Response Normalization
3x3 Max Pooling (Stride of 2)
Local Response Normalization
3x3 Max Pooling (Stride of 2)

Selected papers related to deep CNNs:

[1] D. X. Zhou, Deep distributed convolutional neural networks: Universality, Anal. Appl. **16** (2018), 895–919.

[2] D. X. Zhou, Universality of deep convolutional neural networks, Appl. Comput. Harmonic Anal. **48** (2020), 787-794.

[3] D. X. Zhou, Theory of deep convolutional neural networks: Downsampling, Neural Networks **124** (2020), 319-327.

[4] Z. Y. Fang, H. Feng, S. Huang, and D. X. Zhou, Theory of deep convolutional neural networks II: Spherical analysis, Neural Networks **131** (2020), 154-162.

[5] T. Y. Zhou and D. X. Zhou, Theory of deep CNNs induced by 2D convolutions, submitted, 2020.

[6] T. Mao, Z. J. Shi, and D. X. Zhou, Theory of Deep Convolutional Neural Networks III: Approximating radial functions, submitted, 2020.

[7] D. X. Zhou, Deep convolutional neural networks, invited survey for the Wiley Encyclopedia of Electrical and Electronics Engineering, 2020.

[8] C. K. Chui, S. B. Lin, B. Zhang, and D. X. Zhou, Realization of spatial sparseness by deep ReLU nets with massive data, IEEE Transactions on Neural Networks and Learning Systems, in press.

[9] Z. Han, S. Q. Yu, S. B. Lin, and D. X. Zhou, Depth selection for deep ReLU nets in feature extraction and generalization, IEEE Trans. Pattern Anal. Machine Intelligence, in press.

Thank you very much!