Deep Learning: Approximation of Functions by Composition

Zuowei Shen

Department of Mathematics
National University of Singapore

Outline

1. A brief introduction of approximation theory
2. Deep learning: approximation of functions by composition
3. Approximation of CNNs and sparse coding
4. Approximation in feature space

Outline

1. A brief introduction of approximation theory
2. Deep learning: approximation of functions by composition
3. Approximation of CNNs and sparse coding
4. Approximation in feature space

A brief introduction of approximation theory

For a given function $f : \mathbb{R}^d \to \mathbb{R}$ and $\epsilon > 0$, approximation is to find a simple function g such that

$$\|f - g\| < \epsilon.$$

A brief introduction of approximation theory

For a given function $f : \mathbb{R}^d \to \mathbb{R}$ and $\epsilon > 0$, approximation is to find a simple function g such that

$$\|f - g\| < \epsilon.$$

Function $g : \mathbb{R}^n \to \mathbb{R}$ can be as simple as $g(x) = a \cdot x$. To make sense of this approximation, we need to find a map $T : \mathbb{R}^d \mapsto \mathbb{R}^n$, such that

$$\|f - g \circ T\| < \epsilon.$$

A brief introduction of approximation theory

For a given function $f : \mathbb{R}^d \to \mathbb{R}$ and $\epsilon > 0$, approximation is to find a simple function g such that

$$\|f - g\| < \epsilon.$$

Function $g : \mathbb{R}^n \to \mathbb{R}$ can be as simple as $g(x) = a \cdot x$. To make sense of this approximation, we need to find a map $T : \mathbb{R}^d \mapsto \mathbb{R}^n$, such that

$$\|f - g \circ T\| < \epsilon.$$

1. Classical approximation: T is independent of data and n depends on ϵ.
2. Deep learning: T depends on the data and n can be independent of ϵ (T is learned from data).

Classical approximation

- Linear approximation: Given a finite set of generators $\{\phi_1, \ldots, \phi_n\}$, e.g. splines, wavelet frames, finite elements or generators in reproducing kernel Hilbert spaces. Define

  $$T = [\phi_1, \phi_2, \ldots, \phi_n]^\top : \mathbb{R}^d \mapsto \mathbb{R}^n \quad \text{and} \quad g(x) = a \cdot x.$$

  The linear approximation is to find $a \in \mathbb{R}^n$ such that

  $$g \circ T = \sum_{i=1}^{n} a_i \phi_i \sim f$$

  It is linear because $f_1 \sim g_1, f_2 \sim g_2 \Rightarrow f_1 + f_2 \sim g_1 + g_2$.

Classical approximation

- Linear approximation: Given a finite set of generators $\{\phi_1, \ldots, \phi_n\}$, e.g. splines, wavelet frames, finite elements or generators in reproducing kernel Hilbert spaces. Define

  $$T = [\phi_1, \phi_2, \ldots, \phi_n]^\top : \mathbb{R}^d \mapsto \mathbb{R}^n \quad \text{and} \quad g(x) = a \cdot x.$$

  The linear approximation is to find $a \in \mathbb{R}^n$ such that

  $$g \circ T = \sum_{i=1}^{n} a_i \phi_i \sim f$$

  It is linear because $f_1 \sim g_1, f_2 \sim g_2 \Rightarrow f_1 + f_2 \sim g_1 + g_2$.
- Nonlinear approximation: Given infinite generators $\Phi = \{\phi_i\}_{i=1}^{\infty}$ and define

  $$T = [\phi_1, \phi_2, \ldots,]^\top : \mathbb{R}^d \mapsto \in \mathbb{R}^\infty \quad \text{and} \quad g(x) = a \cdot x$$

  The nonlinear approximation of f is to find a finitely supported a such that $g \circ T \sim f$.

  It is nonlinear because $f_1 \sim g_1, f_2 \sim g_2 \not\Rightarrow f_1 + f_2 \sim g_1 + g_2$ as the support of the approximator g of f depends on f.

Examples

Consider a function space $L_2(\mathbb{R}^d)$, let $\{\phi_i\}_{i=1}^{\infty}$ be an orthonormal basis of $L_2(\mathbb{R}^d)$.

Examples

Consider a function space $L_2(\mathbb{R}^d)$, let $\{\phi_i\}_{i=1}^{\infty}$ be an orthonormal basis of $L_2(\mathbb{R}^d)$.

Linear approximation
For a given n, $T = [\phi_1, \ldots, \phi_n]^\top$ and $g = a \cdot x$ where $a_j = \langle f, \phi_j \rangle$. Denote $\mathcal{H} = \operatorname{span}\{\phi_1, \ldots, \phi_n\} \subseteq L_2(\mathbb{R}^d)$.
Then,

$$g \circ T = \sum_{i=1}^{n} \langle f, \phi_i \rangle \phi_i$$

is the orthogonal projection onto the space $\mathcal{H}$ and is the best approximation of f from the space $\mathcal{H}$.

Examples

Consider a function space $L_2(\mathbb{R}^d)$, let $\{\phi_i\}_{i=1}^{\infty}$ be an orthonormal basis of $L_2(\mathbb{R}^d)$.

Linear approximation
For a given n, $T = [\phi_1, \ldots, \phi_n]^\top$ and $g = a \cdot x$ where $a_j = \langle f, \phi_j \rangle$. Denote $\mathcal{H} = \mathrm{span}\{\phi_1, \ldots, \phi_n\} \subseteq L_2(\mathbb{R}^d)$.
Then,

$$g \circ T = \sum_{i=1}^{n} \langle f, \phi_i \rangle \phi_i$$

is the orthogonal projection onto the space $\mathcal{H}$ and is the best approximation of f from the space $\mathcal{H}$.
$g \circ T$ provides a good approximation of f when the sequence $\{\langle f, \phi_j \rangle\}_{j=1}^{\infty}$ decays fast as $j \to +\infty$.

Examples

Consider a function space $L_2(\mathbb{R}^d)$, let $\{\phi_i\}_{i=1}^{\infty}$ be an orthonormal basis of $L_2(\mathbb{R}^d)$.

Linear approximation

For a given n, $T = [\phi_1, \dots, \phi_n]^\top$ and $g = a \cdot x$ where $a_j = \langle f, \phi_j \rangle$. Denote $\mathcal{H} = \mathrm{span}\{\phi_1, \dots, \phi_n\} \subseteq L_2(\mathbb{R}^d)$.

Then,

$$g \circ T = \sum_{i=1}^{n} \langle f, \phi_i \rangle \phi_i$$

is the orthogonal projection onto the space $\mathcal{H}$ and is the best approximation of f from the space $\mathcal{H}$.

$g \circ T$ provides a good approximation of f when the sequence $\{\langle f, \phi_j \rangle\}_{j=1}^{\infty}$ decays fast as $j \to +\infty$.

Therefore,

1. Linear approximation provides a good approximation for smooth functions.
2. When $n = \infty$, it reproduces any function in $L_2(\mathbb{R}^d)$.
3. **Advantage:** It is a good approximation scheme for d is small, domain is simple, function form is complicated but smooth.
4. **Disadvantage:** if d is big and ϵ is small, n is huge.

Examples

Nonlinear approximation

$T = (\phi_j)_{j=1}^{\infty} : \mathbb{R}^d \mapsto \mathbb{R}^{\infty}$ and $g(x) = a \cdot x$ and each a_j is

$$a_j = \begin{cases} \langle f, \phi_j \rangle, & \text{for the largest } n \text{ terms in the sequence } \{|\langle f, \phi_j \rangle|\}_{j=1}^{\infty} \\ 0, & \text{otherwise.} \end{cases}$$

Examples

Nonlinear approximation

$T = (\phi_j)_{j=1}^{\infty} : \mathbb{R}^d \mapsto \mathbb{R}^{\infty}$ and $g(x) = a \cdot x$ and each a_j is

$$a_j = \begin{cases} \langle f, \phi_j \rangle, & \text{for the largest } n \text{ terms in the sequence } \{|\langle f, \phi_j \rangle|\}_{j=1}^{\infty} \\ 0, & \text{otherwise.} \end{cases}$$

The approximation of f by $g \circ T$ depends less on the decay of the sequence $\{|\langle f, \phi_j \rangle|\}_{j=1}^{\infty}$. Therefore,

1. the nonlinear approximation is better than the linear approximation when f is nonsmooth.
2. curse of dimensionality: if d is big and ϵ is small, n is huge.

Both linear and nonlinear approximations are schemes to approximate a class of function where T is fixed and it essentially changes a basis in order to represent or approximate a certain class of functions.

Both linear and nonlinear approximations do not suit for approximating f when f is defined on a complex domain, e.g manifold in a very high dimensional space.

However, in deep learning, T is constructed by given data that is adaptive to the underlying function to be approximated. T changes variables and maps domain of f to a feature domain where approximation become simpler, robust, and efficient.

Deep learning approximation is to find map T that maps the domain of f to a “simple/ better domain” so that simple classical approximation can be applied.

Outline

1. A brief introduction of approximation theory
2. Deep learning: approximation of functions by composition
3. Approximation of CNNs and sparse coding
4. Approximation in feature space

Approximation for deep learning

Given data $\{(x_i, f(x_i))\}_{i=1}^{m}$.

1. The key of deep learning is to construct a T by the given data.
2. T can simplify the domain of f through the change of variables.
3. T maps the key features of the domain of f and f , so that
4. It is easy to find g s.t. $g \circ T$ gives a good approximation of f.

What is the mathematics behind this?

Settings: construct a map $T : \mathbb{R}^d \mapsto \mathbb{R}^n$ and a simple function g (e.g. $g = a \cdot x$) from data such that $g \circ T$ provides a good approximation of f.

Approximation by compositions

Question 1: For arbitrarily given $\epsilon > 0$, is there $T : \mathbb{R}^d \mapsto \mathbb{R}$ such that $\|f - g \circ T\| \leq \epsilon$?

Approximation by compositions

Question 1: For arbitrarily given $\epsilon > 0$, is there $T : \mathbb{R}^d \mapsto \mathbb{R}$ such that $\|f - g \circ T\| \leq \epsilon$?

Answer: Yes!

Theorem

Let $f : \mathbb{R}^d \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}$ and assume $\mathrm{Im}(f) \subseteq \mathrm{Im}(g)$. For an arbitrarily given $\epsilon > 0$, there exists $T : \mathbb{R}^d \mapsto \mathbb{R}^n$ such that

$$\|f - g \circ T\| \leq \epsilon$$

T can be explicitly written out in terms of f and g.

- T can be complex. This leads to

Approximation by compositions

Question 2: can T be a composition of simple maps? That is, can we write $T = T_1 \circ \cdots \circ T_J$ and each T_i, $i = 1, 2, \ldots, J$ is simple, e.g. perturbation of identity.

Answer: Yes!

Theorem

Denote $f : \mathbb{R}^d \to \mathbb{R}$ *and* $g : \mathbb{R}^n \to \mathbb{R}$*. For an arbitrarily given* $\epsilon > 0$*, if* $\mathrm{Im}(f) \subseteq \mathrm{Im}(g)$*, then there exists* J *simple maps* $T_i, i = 1, 2, \ldots, J$ *such that* $T = T_1 \circ T_2 \ldots \circ T_J : \mathbb{R}^d \mapsto \mathbb{R}^n$ *and*

$$\|f - g \circ T_1 \circ \cdots \circ T_J\| \leq \epsilon$$

T_i, $i = 1, 2, \ldots, J$ can be written out explicitly in terms of T.

Approximation by compositions

Question 3: Can $T_i, i = 1, 2, \ldots, J$ be mathematically constructed by some scheme?
Answer: Yes! T_i, $i = 1, \ldots, J$ can be constructed by solving the minimization problem:

$$\min_{T_1, T_2, \ldots, T_J} \|f - g \circ T_1 \circ \cdots \circ T_J\|$$

A constructive proof is given in paper.

Approximation by compositions

Question 3: Can $T_i, i = 1, 2, \ldots, J$ be mathematically constructed by some scheme?
Answer: Yes! T_i, $i = 1, \ldots, J$ can be constructed by solving the minimization problem:

$$\min_{T_1, T_2, \ldots, T_J} \|f - g \circ T_1 \circ \cdots \circ T_J\|$$

A constructive proof is given in paper.
Question 4: Given training data $\{x_i, f(x_i)\}_{i=1}^m$, can we design numerical scheme to find $\tilde{T}_i$, $i = 1, 2 \ldots, J$ and $\tilde{g}$ such that

$$\|f - \tilde{g} \circ \tilde{T}_1 \circ \cdots \circ \tilde{T}_J\| \leq \epsilon, \quad \text{with high probability}$$

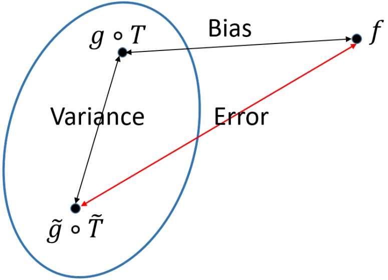
by minimizing

$$\frac{1}{m} \sum_{i=1}^{m} (f(x_i) - \tilde{g} \circ \tilde{T}_1 \circ \cdots \circ \tilde{T}_J(x_i))^2?$$

Answer: Yes! We do have designed deep neural networks for that. Numerical simulations show it performs well.

Ideas

One of the simplest ideas is



$$\begin{aligned} & \|f - \tilde{g} \circ \tilde{T}_1 \circ \cdots \tilde{T}_J\| \\ \leq & \|f - g \circ T_1 \circ \cdots \circ T_J\| + \|g \circ T_1 \circ \cdots \circ T_J - \tilde{g} \circ \tilde{T}_1 \circ \cdots \tilde{T}_J\| \\ =: & \text{ Bias } \qquad + \qquad \text{Variance} \end{aligned}$$

Li, Shen and Tai, Deep learning: approximation of functions by composition, 2018.

This theory is complete, but does not answer all questions! For example, we do not have approximation order in terms of the number of nodes yet.

This theory is complete, but does not answer all questions! For example, we do not have approximation order in terms of the number of nodes yet.

There are many different machine learning architectures, e.g. convolutional neural networks (CNNs) and sparse coding based classifiers, are different from the architecture we designed here.

Next, we will present approximation theory of CNNs and sparse coding based classifiers for image classification.

Outline

1. A brief introduction of approximation theory
2. Deep learning: approximation of functions by composition
3. Approximation of CNNs and sparse coding
4. Approximation in feature space

Approximation for image classification

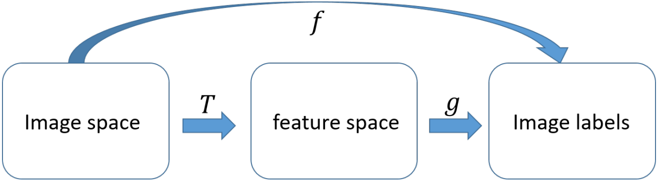
- Binary classification
  - $\Omega = \Omega_0 \cup \Omega_1$ and $\overline{\Omega}_0 \cap \overline{\Omega}_1 = \emptyset$.
  - Let f to be the oracle classifier, i.e. $f : \Omega \subset \mathbb{R}^d \to \{0, 1\}$ and

$$f(x) = \begin{cases} 0, & \text{if } x \in \Omega_0, \\ 1, & \text{if } x \in \Omega_1. \end{cases}$$

- Construct feature map T and the classifier g such that

$$f - g \circ T$$

is small.



Approximation for image classification

- g is normally a fully connected layer followed by softmax defined on feature space.
- Construct a feature map T, so that $g \circ T$ approximates f well, when constructing a fully connected layer to approximate f from the data in image space is hard.

Approximation for image classification

- g is normally a fully connected layer followed by softmax defined on feature space.
- Construct a feature map T, so that $g \circ T$ approximates f well, when constructing a fully connected layer to approximate f from the data in image space is hard.

$g \circ T$ gives a good approximation of f if T satisfies

$$\|T(x) - T(y)\| \leq C\|x - y\|, \quad \forall x, y \in \Omega, \tag{3.1}$$

$$\|T(x) - T(y)\| > c\|x - y\|, \quad \forall x \in \Omega_0, y \in \Omega_1. \tag{3.2}$$

for some $C, c > 0$.

The above inequalities are not easy to prove for both CNNs and sparse coding based classifiers especially when $n \leq d$.

Accuracy of image classification of CNNs

The CNNs achieve desired classification accuracy with high probability!

All the numerical results confirmed it.

Accuracy of image classification of CNNs

The CNNs achieve desired classification accuracy with high probability!

All the numerical results confirmed it.

Settings:

Given J sets of convolution kernels $\{\mathbf{w}_i\}_{i=1}^J$ and bias $\{\mathbf{b_i}\}_{i=1}^J$.
A J layer convolutional neural network is a nonlinear function

$$g \circ T$$

where

$$T(x) = \sigma(\mathbf{w}_J \circledast \sigma(\mathbf{w}_{J-1} \circledast \cdots (\sigma(\mathbf{w}_1 \circledast x + \mathbf{b}_1)) \cdots + \mathbf{b}_{J-1}) + \mathbf{b}_J)$$

$$g(x) = \sum_{i=1}^{n} a_i \sigma(w_i^\top x + b_i), \quad \text{and} \quad \sigma(x) = \max(0, x).$$

Normally, the convolutional kernels $\{\mathbf{w}_i\}_{i=1}^J$ have small size.

Accuracy of image classification of CNNs

The CNNs achieve desired classification accuracy with high probability!

All the numerical results confirmed it.

Settings:

Given J sets of convolution kernels $\{\mathbf{w}_i\}_{i=1}^J$ and bias $\{\mathbf{b_i}\}_{i=1}^J$.
A J layer convolutional neural network is a nonlinear function

$$g \circ T$$

where

$$T(x) = \sigma(\mathbf{w}_J \circledast \sigma(\mathbf{w}_{J-1} \circledast \cdots (\sigma(\mathbf{w}_1 \circledast x + \mathbf{b}_1)) \cdots + \mathbf{b}_{J-1}) + \mathbf{b}_J)$$

$$g(x) = \sum_{i=1}^n a_i \sigma(w_i^\top x + b_i), \quad \text{and} \quad \sigma(x) = \max(0, x).$$

Normally, the convolutional kernels $\{\mathbf{w}_i\}_{i=1}^J$ have small size.
Given m training samples $\{(x_i, y_i)\}_{i=1}^m$, $\tilde{T}$ and a $\tilde{g}$ are learned from

$$\min_{g,T} \frac{1}{m} \sum_{i=1}^m (y_i - g \circ T(x_i))^2.$$

Accuracy of image classification of CNNs

Question: Whether can we have a rigorous proof for this statement?

Answer: Yes!

Theorem

For any given $\epsilon > 0$ and sample data $\mathcal{Z}$ with sample size m, there exists a CNN classifier whose filter size can be as small as 3, such that the classifier accuracy $\mathcal{A}$ satisfies

$$\mathbb{P}(\mathcal{A} \geq 1 - \epsilon) \geq 1 - \eta(\epsilon, m),$$

$\eta(\epsilon, m) \to 0$ *as* $m \to +\infty$*.*

The difficult part is to prove the inequalities (3.1) and (3.2) for $\tilde{T}$.

Bao, Shen, Tai, Wu and Xiang, Approximation and scaling analysis of convolutional neural networks, 2017.

Accuracy of image classification of sparse coding

Given m training samples $\{(x_i, y_i)\}_{i=1}^m$, the sparse coding based classifier learn $\tilde{D}$ and $\tilde{W}$ via solving the problem

$$\min_{\|d_k\|=1, \{c_i\}_{i=1}^m, W} \frac{1}{m} \sum_{i=1}^m \left\{ \|x_i - Dc_i\|^2 + \lambda \|c_i\|_0 + \gamma \|y_i - Wc_i\|^2 \right\}$$

There are numerical algorithms with global convergence property to solve the above minimization.

Accuracy of image classification of sparse coding

Given m training samples $\{(x_i, y_i)\}_{i=1}^m$, the sparse coding based classifier learn $\tilde{D}$ and $\tilde{W}$ via solving the problem

$$\min_{\|d_k\|=1, \{c_i\}_{i=1}^m, W} \frac{1}{m} \sum_{i=1}^m \left\{ \|x_i - Dc_i\|^2 + \lambda \|c_i\|_0 + \gamma \|y_i - Wc_i\|^2 \right\}$$

There are numerical algorithms with global convergence property to solve the above minimization.

The sparse coding based classifier is $\tilde{g} \circ \tilde{T}$, where

$$\tilde{g}(x) = \tilde{W}x \quad \text{and} \quad \tilde{T}(x) \in \arg\min_c \|x - \tilde{D}c\|^2 + \lambda \|c\|_0.$$

There is no mathematical analysis of classification accuracy of $\tilde{g} \circ \tilde{T}$, i.e. $\|f - \tilde{g} \circ \tilde{T}\|$.

Bao, Ji, Quan, Shen, Dictionary learning for sparse coding: Algorithms and convergence analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(7), (2016), 1356-1369.

Bao, Ji, Quan, Shen, L_0 norm based dictionary learning by proximal methods with global convergence, IEEE Conference Computer Vision and Pattern Recognition (CVPR), Columbus, (2014).

Accuracy of image classification of sparse coding

Consider an orthogonal dictionary learning (ODL) scheme

$$\min_{D^\top D=I,\{c_i\}_{i=1}^m,g} \frac{1}{m}\sum_{i=1}^{m}\left\{\|x_i - Dc_i\|^2 + \lambda\|c_i\|_1 + \gamma\|y_i - g(c_i)\|^2\right\} \quad (3.3)$$

where g is a fully connected layer.
The numerical algorithm to solve the above problem has global convergence property.

The classification accuracy is similar to the previous models.

Classification accuracies (%)

Dataset	K-SVD	D-KSVD	IDL	ODL
Face: Extended Yale B	93.10	94.10	95.72	96.12
Face: AR face	86.50	88.80	96.18	96.37
Object: Caltech101	68.70	68.60	72.29	72.54

Accuracy of image classification of sparse coding

Consider an orthogonal dictionary learning (ODL) scheme

$$\min_{D^\top D=I,\{c_i\}_{i=1}^m,g} \frac{1}{m}\sum_{i=1}^{m}\left\{\|x_i - Dc_i\|^2 + \lambda\|c_i\|_1 + \gamma\|y_i - g(c_i)\|^2\right\} \quad (3.3)$$

where g is a fully connected layer.
The numerical algorithm to solve the above problem has global convergence property.

The classification accuracy is similar to the previous models.

Classification accuracies (%)

Dataset	K-SVD	D-KSVD	IDL	ODL
Face: Extended Yale B	93.10	94.10	95.72	96.12
Face: AR face	86.50	88.80	96.18	96.37
Object: Caltech101	68.70	68.60	72.29	72.54

Accuracy of image classification of sparse coding

Consider an orthogonal dictionary learning (ODL) scheme

$$\min_{D^\top D=I,\{c_i\}_{i=1}^m,g} \frac{1}{m}\sum_{i=1}^m \left\{\|x_i - Dc_i\|^2 + \lambda\|c_i\|_1 + \gamma\|y_i - g(c_i)\|^2\right\} \quad (3.3)$$

where g is a fully connected layer.
The numerical algorithm to solve the above problem has global convergence property.

The classification accuracy is similar to the previous models.

Classification accuracies (%)

Dataset	K-SVD	D-KSVD	IDL	ODL
Face: Extended Yale B	93.10	94.10	95.72	96.12
Face: AR face	86.50	88.80	96.18	96.37
Object: Caltech101	68.70	68.60	72.29	72.54

For this model, we have mathematical analysis of accuracy.

Sparse coding approximation of image classification

Let $\tilde{g}$ to be the fully connected layer and $\tilde{D}$ is the dictionary from solving (3.3). Define

$$\tilde{T}(x) = \arg\min_{c} \|x - \tilde{D}c\|^2 + \lambda\|c\|_1.$$

The sparse coding based classifier from ODL model is $\tilde{g} \circ \tilde{T}$.

Sparse coding approximation of image classification

Let $\tilde{g}$ to be the fully connected layer and $\tilde{D}$ is the dictionary from solving (3.3). Define

$$\tilde{T}(x) = \arg\min_{c} \|x - \tilde{D}c\|^2 + \lambda\|c\|_1.$$

The sparse coding based classifier from ODL model is $\tilde{g} \circ \tilde{T}$.

Theorem

Consider the ODL model. For any given $\epsilon > 0$ and sample data $\mathcal{Z}$ with sample size m, there exists a sparse coding based classifier, such that the classifier accuracy $\mathcal{A}$ satisfies

$$\mathbb{P}(\mathcal{A} \geq 1 - \epsilon) \geq 1 - \eta(\epsilon, m),$$

$\eta(\epsilon, m) \to 0$ *as* $m \to +\infty$*.*

To prove the two inequalities (3.1) and (3.2) of $\tilde{T}$ is not easy.

Bao, Ji and Shen, Classification accuracy of sparse coding based classifier, 2018.

Data-driven tight frame is convolutional sparse coding

Given an image g, the data driven tight frame model solves

$$\begin{aligned} &\min_{\boldsymbol{c},\mathcal{W}} \ \|\mathcal{W}^T\boldsymbol{c}-\boldsymbol{g}\|_2^2+\|(\mathcal{I}-\mathcal{W}\mathcal{W}^T)\boldsymbol{c}\|_2^2+\lambda^2\|\boldsymbol{c}\|_0 \\ &\text{s.t.} \ \ \mathcal{W}^T\mathcal{W}=\mathcal{I}. \end{aligned} \tag{3.4}$$

- When $\mathcal{W}^\top\mathcal{W}=\mathcal{I}$, the rows of $\mathcal{W}$ form a tight frame.

Data-driven tight frame is convolutional sparse coding

Given an image g, the data driven tight frame model solves

$$\begin{aligned} &\min_{\boldsymbol{c},\mathcal{W}} \ \|\mathcal{W}^T\boldsymbol{c}-\boldsymbol{g}\|_2^2 + \|(\mathcal{I}-\mathcal{W}\mathcal{W}^T)\boldsymbol{c}\|_2^2 + \lambda^2\|\boldsymbol{c}\|_0 \\ &\text{s.t. } \ \mathcal{W}^T\mathcal{W} = \mathcal{I}. \end{aligned} \tag{3.4}$$

- When $\mathcal{W}^\top\mathcal{W} = \mathcal{I}$, the rows of $\mathcal{W}$ form a tight frame.
- The minimization model (3.4) is equivalent to

$$\min_{c,\mathcal{W}} \ \|\mathcal{W}\boldsymbol{g}-\boldsymbol{c}\|_2^2 + \lambda^2\|\boldsymbol{c}\|_0, \quad \text{s.t.} \quad \mathcal{W}^\top\mathcal{W} = \mathcal{I}. \tag{3.5}$$

- By the structure of $\mathcal{W}$, each channel of $\mathcal{W}$ corresponds to a convolutional kernel.

Data-driven tight frame is convolutional sparse coding

Given an image g, the data driven tight frame model solves

$$\begin{aligned}&\min_{\boldsymbol{c},\mathcal{W}} \|\mathcal{W}^T\boldsymbol{c}-\boldsymbol{g}\|_2^2+\|(\mathcal{I}-\mathcal{W}\mathcal{W}^T)\boldsymbol{c}\|_2^2+\lambda^2\|\boldsymbol{c}\|_0\\ &\text{s.t. } \mathcal{W}^T\mathcal{W}=\mathcal{I}.\end{aligned} \tag{3.4}$$

- When $\mathcal{W}^\top\mathcal{W}=\mathcal{I}$, the rows of $\mathcal{W}$ form a tight frame.
- The minimization model (3.4) is equivalent to

$$\min_{c,\mathcal{W}} \|\mathcal{W}\boldsymbol{g}-\boldsymbol{c}\|_2^2+\lambda^2\|\boldsymbol{c}\|_0, \quad \text{s.t.} \quad \mathcal{W}^\top\mathcal{W}=\mathcal{I}. \tag{3.5}$$

- By the structure of $\mathcal{W}$, each channel of $\mathcal{W}$ corresponds to a convolutional kernel.
- To solve (3.5), we use ADM. For fixed $\mathcal{W}$, $\boldsymbol{c}$ can be solved by hard thresholding; For fixed $\boldsymbol{c}$, $\mathcal{W}$ has an analytical solution and easy to compute. Thanks the convolution structure of $\mathcal{W}$ and the tight frame property.
- The iteration algorithm converges.

Data-driven tight frame for image denoising

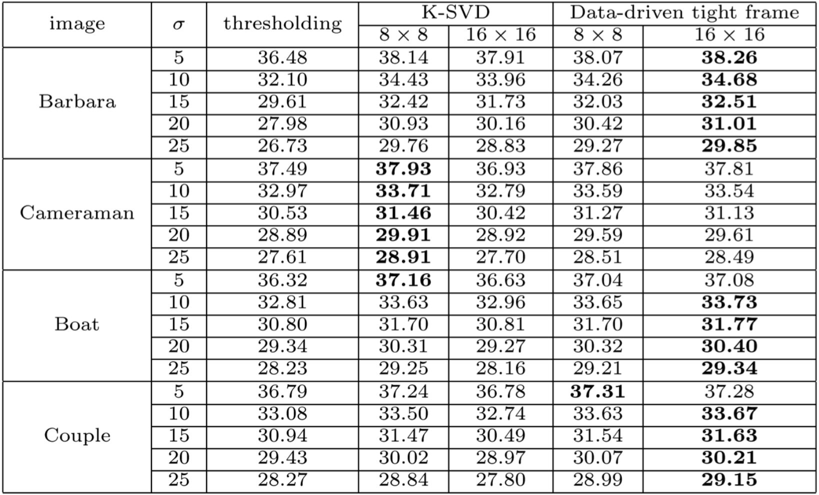


image	σ	thresholding	K-SVD		Data-driven tight frame	
			8×8	16×16	8×8	16×16
Barbara	5	36.48	38.14	37.91	38.07	**38.26**
	10	32.10	34.43	33.96	34.26	**34.68**
	15	29.61	32.42	31.73	32.03	**32.51**
	20	27.98	30.93	30.16	30.42	**31.01**
	25	26.73	29.76	28.83	29.27	**29.85**
Cameraman	5	37.49	**37.93**	36.93	37.86	37.81
	10	32.97	**33.71**	32.79	33.59	33.54
	15	30.53	**31.46**	30.42	31.27	31.13
	20	28.89	**29.91**	28.92	29.59	29.61
	25	27.61	**28.91**	27.70	28.51	28.49
Boat	5	36.32	**37.16**	36.63	37.04	37.08
	10	32.81	33.63	32.96	33.65	**33.73**
	15	30.80	31.70	30.81	31.70	**31.77**
	20	29.34	30.31	29.27	30.32	**30.40**
	25	28.23	29.25	28.16	29.21	**29.34**
Couple	5	36.79	37.24	36.78	**37.31**	37.28
	10	33.08	33.50	32.74	33.63	**33.67**
	15	30.94	31.47	30.49	31.54	**31.63**
	20	29.43	30.02	28.97	30.07	**30.21**
	25	28.27	28.84	27.80	28.99	**29.15**

[1] Cai, Huang, Ji, Shen and Ye, Data-driven tight frame construction and image denoising, Applied and Computational Harmonic Analysis, 37(1), (2014), 89-105.

[2] Bao, Ji and Shen, Convergence analysis for iterative data-driven tight frame construction scheme, Applied and Computational Harmonic Analysis, 38(3), (2015), 510-523.

Outline

1. A brief introduction of approximation theory

2. Deep learning: approximation of functions by composition

3. Approximation of CNNs and sparse coding

4. Approximation in feature space

Back to classical approximation on feature domain

Classical approximation is still useful for approximation in feature space.

Back to classical approximation on feature domain

Classical approximation is still useful for approximation in feature space.

- Given noisy data $\{x_i, y_i\}_{i=1}^m$ with

$$y_i = (\mathcal{S}f)(x_i) + n_i,$$

where $\{y_i\}_{i=1}^m$ are samples of f with noise n_i.

Back to classical approximation on feature domain

Classical approximation is still useful for approximation in feature space.

- Given noisy data $\{x_i, y_i\}_{i=1}^m$ with

$$y_i = (\mathcal{S}f)(x_i) + n_i,$$

  where $\{y_i\}_{i=1}^m$ are samples of f with noise n_i.
- By applying some data fitting scheme, e.g., the wavelet frame scheme, one obtains the denoised result

$$\{y_i^*\}_{i=1}^m.$$

Back to classical approximation on feature domain

Classical approximation is still useful for approximation in feature space.

- Given noisy data $\{x_i, y_i\}_{i=1}^m$ with

$$y_i = (\mathcal{S}f)(x_i) + n_i,$$

  where $\{y_i\}_{i=1}^m$ are samples of f with noise n_i.
- By applying some data fitting scheme, e.g., the wavelet frame scheme, one obtains the denoised result

$$\{y_i^*\}_{i=1}^m.$$

- Let g be the function reconstructed by y_i^* through some approximation scheme.

Back to classical approximation on feature domain

Classical approximation is still useful for approximation in feature space.

- Given noisy data $\{x_i, y_i\}_{i=1}^m$ with

$$y_i = (\mathcal{S}f)(x_i) + n_i,$$

  where $\{y_i\}_{i=1}^m$ are samples of f with noise n_i.
- By applying some data fitting scheme, e.g., the wavelet frame scheme, one obtains the denoised result

$$\{y_i^*\}_{i=1}^m.$$

- Let g be the function reconstructed by y_i^* through some approximation scheme.
- **Question:**
  1. What's the error between g and f?
  2. Can we have $g \longrightarrow f$ when the sampling data is sufficiently dense?

Data fitting

Let $\Omega := [0,1] \times [0,1]$ and $f \in L_2(\Omega)$. Let ϕ be the tensor product of B-spline functions and denote the scaled functions by $\phi_{n,\alpha} := 2^n \phi(2^n \cdot - \alpha)$. Let $(\mathcal{S}f)[\alpha] = 2^n \langle f, \phi_{n,\alpha} \rangle$.

Data fitting

Let $\Omega := [0,1] \times [0,1]$ and $f \in L_2(\Omega)$. Let ϕ be the tensor product of B-spline functions and denote the scaled functions by $\phi_{n,\alpha} := 2^n \phi(2^n \cdot - \alpha)$. Let $(\mathcal{S}f)[\alpha] = 2^n \langle f, \phi_{n,\alpha} \rangle$.

Given noisy observations

$$y[\alpha] = (\mathcal{S}f)[\alpha] + n_\alpha, \alpha = (\alpha_1, \alpha_2),\ 0 \leq \alpha_1, \alpha_2 \leq 2^n - 1.$$

The data fitting problem is to recover f on Ω from $\boldsymbol{y}$.

Wavelet frame

- Let ϕ be a refinable function and $\Psi := \{\psi_i, i = 1, \ldots, r\}$ be the wavelet functions associated with ϕ in $L_2(\mathbb{R}^2)$.

Wavelet frame

- Let ϕ be a refinable function and $\Psi := \{\psi_i, i = 1, \ldots, r\}$ be the wavelet functions associated with ϕ in $L_2(\mathbb{R}^2)$.
- Denote the scaled functions by

$$\phi_{n,\alpha} := 2^n \phi(2^n \cdot -\alpha) \quad \text{and} \quad \psi_{i,n,\alpha} := 2^n \psi_i(2^n \cdot -\alpha).$$

Wavelet frame

- Let ϕ be a refinable function and $\Psi := \{\psi_i, i = 1, \ldots, r\}$ be the wavelet functions associated with ϕ in $L_2(\mathbb{R}^2)$.
- Denote the scaled functions by

$$\phi_{n,\alpha} := 2^n \phi(2^n \cdot -\alpha) \quad \text{and} \quad \psi_{i,n,\alpha} := 2^n \psi_i(2^n \cdot -\alpha).$$

- $X(\Psi) := \{\psi_{i,n,\alpha}\}$ is a tight frame if

$$\|f\|_2^2 = \sum_{i,n,\alpha} |\langle f, \psi_{i,n,\alpha}\rangle|^2, \forall f \in L_2(\mathbb{R}).$$

- Unitary extension principle (UEP): Assume the masks h_i satisfy the following equalities

$$2\sum_{i=0}^{r} \sum_{k\in\mathbb{Z}} \overline{h_i(m+2k+\ell)} h_i(2k+\ell) = \delta_m, \text{ for any } m, \ell \in \mathbb{Z},$$

$X(\Psi) := \{\psi_{i,n,\alpha}\}$ is a tight frame.

Wavelet frame

- Let ϕ be a refinable function and $\Psi := \{\psi_i, i = 1, \ldots, r\}$ be the wavelet functions associated with ϕ in $L_2(\mathbb{R}^2)$.
- Denote the scaled functions by

$$\phi_{n,\alpha} := 2^n \phi(2^n \cdot -\alpha) \quad \text{and} \quad \psi_{i,n,\alpha} := 2^n \psi_i(2^n \cdot -\alpha).$$

- $X(\Psi) := \{\psi_{i,n,\alpha}\}$ is a tight frame if

$$\|f\|_2^2 = \sum_{i,n,\alpha} |\langle f, \psi_{i,n,\alpha}\rangle|^2, \forall f \in L_2(\mathbb{R}).$$

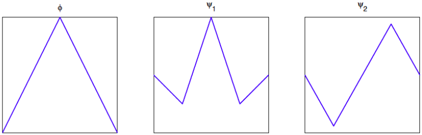
- Unitary extension principle (UEP): Assume the masks h_i satisfy the following equalities

$$2 \sum_{i=0}^{r} \sum_{k \in \mathbb{Z}} \overline{h_i(m + 2k + \ell)} h_i(2k + \ell) = \delta_m, \text{ for any } m, \ell \in \mathbb{Z},$$

$X(\Psi) := \{\psi_{i,n,\alpha}\}$ is a tight frame.

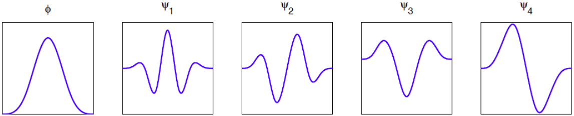
Ron and Shen, Affine systems in $L_2(\mathbb{R}^d)$: the analysis of the analysis operator, Journal of Functional Analysis, 148(2), (1997), 408-447.
Daubechies, Han, Ron and Shen, Framelets: MRA-based constructions of wavelet frames, Applied and Computational Harmonic Analysis, 14(1), (2003), 1-46.

- Examples of spline wavelets from UEP



Piecewise linear refinable B-spline and the corresponding framelets.

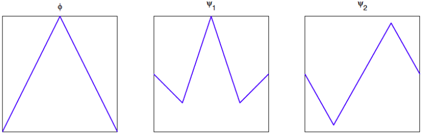
Refinement mask $h_0 = [\frac{1}{4}, \frac{1}{2}, \frac{1}{4}]$. High pass filters $h_1 = [-\frac{1}{4}, \frac{1}{2}, -\frac{1}{4}]$ and $h_2 = [\frac{\sqrt{2}}{4}, 0, -\frac{\sqrt{2}}{4}]$.



Piecewise cubic refinable B-spline and the corresponding framelets.

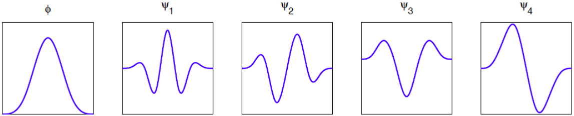
Refinement mask $h_0 = [\frac{1}{16}, \frac{1}{4}, \frac{3}{8}, \frac{1}{4}, \frac{1}{16}]$. High pass filters $h_1 = [\frac{1}{16}, -\frac{1}{4}, \frac{3}{8}, -\frac{1}{4}, \frac{1}{16}]$, $h_2 = [-\frac{1}{8}, \frac{1}{4}, 0, -\frac{1}{4}, \frac{1}{8}]$, $h_3 = [\frac{\sqrt{6}}{16}, 0, -\frac{\sqrt{6}}{8}, 0, \frac{\sqrt{6}}{16}]$ and $h_4 = [-\frac{1}{8}, -\frac{1}{4}, 0, \frac{1}{4}, \frac{1}{8}]$.

- Examples of spline wavelets from UEP



Piecewise linear refinable B-spline and the corresponding framelets.

Refinement mask $h_0 = [\frac{1}{4}, \frac{1}{2}, \frac{1}{4}]$. High pass filters $h_1 = [-\frac{1}{4}, \frac{1}{2}, -\frac{1}{4}]$ and $h_2 = [\frac{\sqrt{2}}{4}, 0, -\frac{\sqrt{2}}{4}]$.



Piecewise cubic refinable B-spline and the corresponding framelets.

Refinement mask $h_0 = [\frac{1}{16}, \frac{1}{4}, \frac{3}{8}, \frac{1}{4}, \frac{1}{16}]$. High pass filters $h_1 = [\frac{1}{16}, -\frac{1}{4}, \frac{3}{8}, -\frac{1}{4}, \frac{1}{16}]$, $h_2 = [-\frac{1}{8}, \frac{1}{4}, 0, -\frac{1}{4}, \frac{1}{8}]$, $h_3 = [\frac{\sqrt{6}}{16}, 0, -\frac{\sqrt{6}}{8}, 0, \frac{\sqrt{6}}{16}]$ and $h_4 = [-\frac{1}{8}, -\frac{1}{4}, 0, \frac{1}{4}, \frac{1}{8}]$.

- Discrete wavelet transform $\mathcal{W}$:

$$\{a_{n,\alpha} := \langle f, \phi_{n,\alpha}\rangle\} \xrightarrow{\mathcal{W}} \{\langle f, \psi_{i,n,\alpha}\rangle\}_{i=0}^{r} = \{h_i * a_{n,\alpha}\}_{i=0}^{r},$$

where $\{h_i\}$ are the wavelet frame filters.

Wavelet frame based data fitting scheme

Example: analysis-based wavelet approach for data fitting

- Let $\boldsymbol{f}_n^*$ be a minimizer of the model

$$E_n(\boldsymbol{f}) := \|\boldsymbol{f} - \boldsymbol{y}\|_2^2 + \|\mathrm{diag}(\boldsymbol{\lambda}_n)\mathcal{W}_n\boldsymbol{f}\|_1,$$

  where $\boldsymbol{\lambda}$ is a vector which scales the different wavelet channels.

Wavelet frame based data fitting scheme

Example: analysis-based wavelet approach for data fitting

- Let $\boldsymbol{f}_n^*$ be a minimizer of the model

$$E_n(\boldsymbol{f}) := \|\boldsymbol{f} - \boldsymbol{y}\|_2^2 + \|\mathrm{diag}(\boldsymbol{\lambda}_n)\mathcal{W}_n\boldsymbol{f}\|_1,$$

  where $\boldsymbol{\lambda}$ is a vector which scales the different wavelet channels.

- Let $g_n^* := \sum_{\alpha\in\mathbb{I}_n} \boldsymbol{f}_n^*(\alpha)\phi(2^n \cdot -\alpha)$.

Wavelet frame based data fitting scheme

Example: analysis-based wavelet approach for data fitting

- Let $\boldsymbol{f}_n^*$ be a minimizer of the model

$$E_n(\boldsymbol{f}) := \|\boldsymbol{f} - \boldsymbol{y}\|_2^2 + \|\mathrm{diag}(\boldsymbol{\lambda}_n)\mathcal{W}_n\boldsymbol{f}\|_1,$$

  where $\boldsymbol{\lambda}$ is a vector which scales the different wavelet channels.

- Let $g_n^* := \sum_{\alpha \in \mathbb{I}_n} \boldsymbol{f}_n^*(\alpha)\phi(2^n \cdot -\alpha)$.
- What's the bound of $\|g_n^* - f\|_{L_2(\Omega)}$?

Wavelet frame based data fitting scheme

Example: analysis-based wavelet approach for data fitting

- Let $\boldsymbol{f}_n^*$ be a minimizer of the model

$$E_n(\boldsymbol{f}) := \|\boldsymbol{f} - \boldsymbol{y}\|_2^2 + \|\mathrm{diag}(\boldsymbol{\lambda}_n)\mathcal{W}_n\boldsymbol{f}\|_1,$$

  where $\boldsymbol{\lambda}$ is a vector which scales the different wavelet channels.
- Let $g_n^* := \sum_{\alpha\in\mathbb{I}_n} \boldsymbol{f}_n^*(\alpha)\phi(2^n\cdot-\alpha)$.
- What's the bound of $\|g_n^* - f\|_{L_2(\Omega)}$?
- Does g_n^* converge to f in $L_2(\Omega)$ as $n \to \infty$?

- Regularity assumption of f: There exits $\beta > -1$ such that

$$\sum_{\alpha} |\langle f, \phi_{0,\alpha} \rangle| + \sum_{n \geq 0} 2^{\beta n} \sum_{i,\alpha} |\langle f, \psi_{i,n,\alpha} \rangle| < \infty.$$

- Regularity assumption of f: There exits $\beta > -1$ such that

$$\sum_{\alpha} |\langle f, \phi_{0,\alpha}\rangle| + \sum_{n\geq 0} 2^{\beta n} \sum_{i,\alpha} |\langle f, \psi_{i,n,\alpha}\rangle| < \infty.$$

- Then for an arbitrary given $0 < \delta < 1$, the following inequality

$$\|g_n^* - f\|_{L_2(\Omega)} \leq C_1 2^{-n \min\{\frac{1+\beta}{2}, \frac{1}{2}\}} \log \frac{1}{\delta} + C_2 \sigma^2$$

holds with confidence $1 - \delta$.

- Regularity assumption of f: There exits $\beta > -1$ such that

$$\sum_{\alpha} |\langle f, \phi_{0,\alpha} \rangle| + \sum_{n \geq 0} 2^{\beta n} \sum_{i,\alpha} |\langle f, \psi_{i,n,\alpha} \rangle| < \infty.$$

- Then for an arbitrary given $0 < \delta < 1$, the following inequality

$$\|g_n^* - f\|_{L_2(\Omega)} \leq C_1 2^{-n \min\{\frac{1+\beta}{2}, \frac{1}{2}\}} \log \frac{1}{\delta} + C_2 \sigma^2$$

  holds with confidence $1 - \delta$.

- When $n \to \infty$, one can design a data fitting scheme such that

$$\lim_{n \to \infty} \mathcal{E}\big(\|g_n^* - f\|_{L_2(\Omega)}\big) = 0.$$

- Regularity assumption of f: There exits $\beta > -1$ such that

$$\sum_{\alpha} |\langle f, \phi_{0,\alpha}\rangle| + \sum_{n\geq 0} 2^{\beta n} \sum_{i,\alpha} |\langle f, \psi_{i,n,\alpha}\rangle| < \infty.$$

- Then for an arbitrary given $0 < \delta < 1$, the following inequality

$$\|g_n^* - f\|_{L_2(\Omega)} \leq C_1 2^{-n \min\{\frac{1+\beta}{2}, \frac{1}{2}\}} \log \frac{1}{\delta} + C_2\sigma^2$$

  holds with confidence $1 - \delta$.

- When $n \to \infty$, one can design a data fitting scheme such that

$$\lim_{n\to\infty} \mathcal{E}\big(\|g_n^* - f\|_{L_2(\Omega)}\big) = 0.$$

- In this case, data is given on uniform grids.

- When the noisy data are obtained from nonuniform grids or obtained by random sampling, can we have a similar result?

- When the noisy data are obtained from nonuniform grids or obtained by random sampling, can we have a similar result?
- Yes. It is technical but it has been carefully studied.

- When the noisy data are obtained from nonuniform grids or obtained by random sampling, can we have a similar result?
- Yes. It is technical but it has been carefully studied.

Cai, Shen and Ye, Approximation of frame based missing data recovery, Applied and Computational Harmonic Analysis, 31(2), (2011), 185-204.

Yang, Stahl and Shen, An analysis of wavelet frame based scattered data reconstruction, Applied and Computational Harmonic Analysis, 42(3), (2017), 480-507.

Yang, Dong and Shen, Approximation of analog signals from noisy data, manuscript, 2018.

Thank you!

http://www.math.nus.edu.sg/∼matzuows/