Mathematics of Deep Learning, Summer Term 2020

Week 3

Dictionary Learning

Philipp Harms Lars Niemann

University of Freiburg

Overview of Week 3

1. Introduction to Dictionary Learning
2. Approximating Hölder Functions by Splines
3. Approximating Univariate Splines by Multi-Layer Perceptrons
4. Approximating Products by Multi-Layer Perceptrons
5. Approximating Multivariate Splines by Multi-Layer Perceptrons
6. Approximating Hölder Functions by Multi-Layer Perceptrons
7. Wrapup

Acknowledgement of Sources

Sources for this lecture:

- Philipp Christian Petersen (Faculty of Mathematics, University of Vienna): Course on Neural Network Theory.

Mathematics of Deep Learning, Summer Term 2020

Week 3, Video 1

Introduction to Dictionary Learning

Philipp Harms Lars Niemann

University of Freiburg

Signal classes

Definition (Signal class, approximation error)

Let $\mathcal{H}$ be a normed space.

- A signal class is a subset $\mathcal{C}$ of $\mathcal{H}$.
- The approximation error of signal class $\mathcal{C}$ by signal class $\mathcal{A}$ is

$$\sigma(\mathcal{A}, \mathcal{C}) = \sup_{f \in \mathcal{C}} \inf_{g \in \mathcal{A}} \|f - g\|_{\mathcal{H}} .$$

- A function $g \in \mathcal{A}$ which realizes the above infimum is called best approximation of f.

Example:

- $\mathcal{H} = L^2(\Omega)$ for some $\Omega \subseteq \mathbb{R}^d$.
- $\mathcal{C} = C^s(\Omega)$ or $H^s(\Omega)$ for some $s \in \mathbb{R}$
- $\mathcal{A}$ is a set of multi-layer perceptrons, splines, or wavelets

Dictionaries

Definition (Dictionaries)

Let $\mathcal{H}$ be a normed space, and let Λ be a countable index set.

- A dictionary is a collection $\phi = (\phi_\lambda)_{\lambda \in \Lambda}$ of elements in $\mathcal{H}$.
- The set of n-term linear combinations in ϕ is defined for any $n \in \mathbb{N}$ as

$$\Sigma_n(\phi) = \left\{ \sum_{\lambda \in \Lambda} c_\lambda \phi_\lambda : c \in \mathbb{R}^\Lambda, \|c\|_0 \leq n \right\},$$

  where $\|\cdot\|_0$ denotes the number of non-zero entries.
- The n-term approximation error of signal class $\mathcal{C}$ by dictionary ϕ is

$$\sigma(\Sigma_n(\phi), \mathcal{C}) = \sup_{f \in \mathcal{C}} \inf_{g \in \Sigma_n(\phi)} \|f - g\|_{\mathcal{H}}.$$

- A function g which realizes the above infimum is called best n-term approximation of f.

Approximation Rates

Definition (Approximation Rates)

Let $\mathcal{C}$ be a signal class, and let $h \in \mathbb{R}^{\mathbb{N}}$.

- A sequence $(\mathcal{A}_n)_{n\in\mathbb{N}}$ of signal classes achieves an approximation rate of h for $\mathcal{C}$ if
$$\sigma(\mathcal{A}_n, \mathcal{C}) = \mathcal{O}(h_n) \text{ as } n \to \infty\,.$$
- A dictionary ϕ achieves an approximation rate of h for $\mathcal{C}$ if
$$\sigma(\Sigma_n(\phi), \mathcal{C}) = \mathcal{O}(h_n) \text{ as } n \to \infty\,.$$

Remark:

- Bounds on the approximation rate are typically more easily obtained than bounds on the approximation error for finite n.
- A "good" dictionary needs more than just a good approximation rate. Indeed, any dense sequence ϕ in $\mathcal{H}$ achieves any approximation rate for any signal class but is ill-suited for efficient encoding of functions.

Dictionary Learning: Transfer of Approximation

Motivation: show a result of the following type

- If multi-layer perceptrons approximate a dictionary well, and the dictionary approximates a signal class well, then multi-layer perceptrons approximate the signal class well.

Theorem (Transfer of approximation)

Let $\mathcal{C}$ be a signal class in a normed space $\mathcal{H}$ of functions $\mathbb{R}^d \to \mathbb{R}$. Assume that multi-layer perceptrons of depth L with activation function ρ and at most M weights approximate any function in a dictionary ϕ to arbitrary accuracy:

$$\forall \epsilon > 0 \; \forall \lambda \in \Lambda \; \exists \Phi : \quad \mathrm{L}(\Phi) = L, \quad \mathrm{M}(\Phi) \leq M, \quad \|\phi_\lambda - \mathrm{R}(\Phi)\|_{\mathcal{H}} \leq \epsilon \,.$$

Then multi-layer perceptrons with Mn weights approximate $\mathcal{C}$ with error

$$\sigma(\{\mathrm{R}(\Phi) : \mathrm{L}(\Phi) = L, \mathrm{M}(\Phi) \leq Mn\}, \mathcal{C}) \leq \sigma(\Sigma_n(\phi), \mathcal{C}).$$

Proof: Transfer of Approximation

Proof:

- Given $f \in \mathcal{C}$ and $\epsilon > 0$, there exists $g \in \Sigma_n(\phi)$ with

$$\|f - g\|_{\mathcal{H}} \leq \sigma(\Sigma_n(\phi), \mathcal{C}) + \epsilon.$$

- After relabeling we may write $g = \sum_{i \leq n} c_i \phi_i$ for some $c_i \in \mathbb{R}$.
- Given $\epsilon > 0$, there exists neural networks Φ_i for $i = 1, \ldots, n$ with

$$\mathrm{L}(\Phi_i) = L, \quad \mathrm{M}(\Phi_i) \leq M, \quad \|\phi_i - \mathrm{R}(\Phi_i)\|_{\mathcal{H}} \leq \frac{\epsilon}{n \cdot \|c\|_\infty}.$$

- By the subsequent lemma on linear combinations of neural networks, there exists a neural network Φ with

$$\mathrm{L}(\Phi) = L, \quad \mathrm{M}(\Phi) \leq Mn, \quad \Big\| \sum_{i \leq n} c_i \phi_i - \mathrm{R}(\Phi) \Big\|_{\mathcal{H}} \leq \epsilon.$$

- Consequently $\mathrm{R}(\Phi)$ approximates f with error

$$\|f - R(\Phi)\|_{\mathcal{H}} \leq \|f - g\|_{\mathcal{H}} + \|g - R(\Phi)\|_{\mathcal{H}} \leq \sigma(\Sigma_n(\phi), \mathcal{C}) + 2\epsilon. \qquad \square$$

Linear combinations of networks

Lemma (Linear combinations of networks)

Let $\Phi_1, \ldots, \Phi_n$ be neural networks with depth L and input dimension d, and let $c_1, \ldots, c_n \in \mathbb{R}$. Then there exists a neural network Φ with depth L and input dimension d such that

$$\mathrm{R}(\Phi) = \sum_{i \leq n} c_i \, \mathrm{R}(\Phi_i), \qquad \mathrm{M}(\Phi) \leq \sum_{i \leq n} \mathrm{M}(\Phi_i).$$

Proof:

- Let c be the row vector $(c_1, \ldots, c_n) \in \mathbb{R}^{1 \times n}$
- Define the neural network Φ by

$$\Phi = ((c, 0)) \bullet \mathrm{P}(\Phi_1, \ldots, \Phi_n)$$

- Count the number of layers and weights □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Recall the definitions of signal classes, dictionaries, and approximation errors.
- Check: Verify that the network Φ in the lemma on linear combinations has indeed depth L and not $L+1$.
- Check: Is the set $\Sigma_n(\phi)$, which consists of n-term linear combinations in the dictionary ϕ, a linear space?
- Transfer: How is the approximation error related to the one defined in statistical learning theory?

Mathematics of Deep Learning, Summer Term 2020

Week 3, Video 2

Approximating Hölder Functions by Splines

Philipp Harms   Lars Niemann

University of Freiburg

Univariate Splines

Definition (Univariate splines)

Let $k \in \mathbb{N}$.

- The univariate cardinal basis spline of order k on $[0, k]$ is defined as

$$\mathcal{N}_k(x) := \frac{1}{(k-1)!} \sum_{l=0}^{k} (-1)^l \binom{k}{l} (x-l)_+^{k-1} \quad \text{for } x \in \mathbb{R}$$

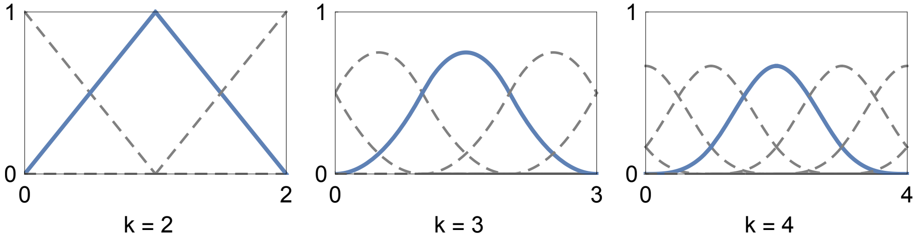
  where $(\cdot)_+ := \max\{0, \cdot\}$.

- For $t \in \mathbb{R}$ and $l \in \mathbb{N}$ we define the univariate basis splines by rescalings and translations:

$$\mathcal{N}_{l,t,k}(x) := \mathcal{N}_k(2^l(x-t)) \quad \text{for } x \in \mathbb{R}\,.$$

Univariate Splines

Plots of the basis spline $\mathcal{N}_k$ (blue) and some translates of it (gray):



Multivariate Splines

Definition (Multivariate splines)

Let $d, k \in \mathbb{N}$.

- For $l \in \mathbb{N}$ and $t \in \mathbb{R}^d$ we define the multivariate basis splines

$$\mathcal{N}^d_{l,t,k}(x) := \prod_{i=1}^{d} \mathcal{N}_{l,t_i,k}(x_i) \quad \text{for } x = (x_1, \dots x_n) \in \mathbb{R}^d \,.$$

- The dictionary of dyadic basis splines of order k is

$$\mathcal{B}^k := (\mathcal{N}^d_{l,t,k})_{l \in \mathbb{N}, t \in 2^{-l}\mathbb{Z}^d} \,.$$

Approximating Hölder Functions by Splines

Theorem

Let $\mathcal{H} = L^p([0,1]^d)$ for some $d \in \mathbb{N}$ and $p \in (0,\infty]$, let $\mathcal{B}^k$ denote the dyadic basis splines of some order $k \in \mathbb{N}$, and let $\mathcal{C}$ be the unit ball in $C^s([0,1]^d)$ for some $s \in (0,k]$. Then for any $r < s/d$, the dictionary $\mathcal{B}^k$ achieves an approximation rate of $(n^{-r})_{n\in\mathbb{N}}$ for the signal class $\mathcal{C}$ in $\mathcal{H}$.

Remark:

- The coefficients c_i in the spline approximation of $f \in \mathcal{C}$ by $\sum_{i\leq n} c_i B_i \in \mathcal{B}^k$ can be chosen such that $\max_i |c_i| \lesssim \|f\|_\infty$.
- More generally, spline approximations of Besov $B^s_{p,q}(\mathbb{R}^d)$ functions converge in Besov $B^{s'}_{p',q'}(\mathbb{R}^d)$ norms at a rate of (nearly) $(n^{-(s-s')/d})_{n\in\mathbb{N}}$. For $p \geq p'$, this follows from the constructive linear theory with non-adaptive grids, whereas for $p < p'$ adaptive grids are needed, and the approximation theory becomes non-constructive and non-linear.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What is the meaning of the parameters l, t, k, d of dyadic basis splines $\mathcal{N}^d_{l,t,k}$?
- Background: Read up on the definition of Hölder functions and splines if needed.
- Transfer: Cubic interpolating splines are the solution of a linear best-approximation problem—which one?

Mathematics of Deep Learning, Summer Term 2020

Week 3, Video 3

Approximating Univariate Splines by Multi-Layer Perceptrons

Philipp Harms   Lars Niemann

University of Freiburg

Sigmoidal Functions of Higher Order

Definition

A function $\rho : \mathbb{R} \to \mathbb{R}$ is called sigmoidal of order $q \in \mathbb{N}$, if $\rho \in C^{q-1}(\mathbb{R})$ and the following three conditions are met:

- $\frac{\rho(x)}{x^q} \to 0$ for $x \to -\infty$.
- $\frac{\rho(x)}{x^q} \to 1$ for $x \to \infty$.
- $|\rho(x)| \lesssim (1+|x|)^q$ for $x \in \mathbb{R}$.

Example:

- Sigmoidal functions are sigmoidal of order 0.
- The ReLu function $x \mapsto (x)_+$ is sigmoidal of order 1.
- The power unit $x \mapsto (x)_+^q$ is sigmoidal of order $q \in \mathbb{N}$.

Goal:

- Approximation of univariate splines by multi-layer perceptrons with sigmoidal activation functions of order $q \geq 2$.

Approximating Power Units by Multi-Layer Perceptrons

Notation:

- $\lceil x \rceil \in \mathbb{Z}$ denotes the the smallest integer greater than or equal to x.

Theorem

Let $k \in \mathbb{N}$, and let $\rho\colon \mathbb{R} \to \mathbb{R}$ sigmoidal of order $q \geq 2$. Then there exists a constant $C > 0$ such that for every $\epsilon, K > 0$, there is a neural network Φ with $\lceil \max\{\log_q(k), 0\} \rceil + 1$ layers and C weights satisfying

$$\sup_{x \in [-K,K]} \left| \mathrm{R}(\Phi)(x) - (x)_+^k \right| \leq \epsilon \,.$$

Remark:

- Two layers suffice for the approximation of square units.

Proof: Approximating Power Units by MLPs

Proof:

- Let $n := \lceil \max\{\log_q(k), 0\} \rceil$, let $p := q^n \geq k$, and let f_λ be the n-fold composition of ρ, rescaled by $\lambda > 0$:

$$f_\lambda(x) := \lambda^{-p} \rho^n(\lambda x) \qquad \text{for } x \in \mathbb{R}.$$

- Then f_λ converges to the p-th power unit:

$$\forall K > 0 : \qquad \lim_{\lambda \to \infty} \sup_{x \in [-K,K]} \left| f_\lambda(x) - (x)_+^p \right| = 0.$$

- The difference quotient converges to the $(p-1)$-th power unit:

$$\forall K > 0 : \qquad \lim_{\substack{\delta \to 0 \\ \lambda \to \infty}} \sup_{x \in [-K,K]} \left| \frac{f_\lambda(x+\delta) - f_\lambda(x)}{\delta} - p(x)_+^{p-1} \right| = 0,$$

  and similarly for higher-order difference quotients and derivatives.

- These difference quotients are realizations of neural networks Φ with $\lceil \max\{\log_q(k), 0\} \rceil + 1$ layers. □

Approximating Univariate Basis Splines by MLPs

Corollary

Any univariate basis spline of degree $k \in \mathbb{N}$ can be approximated uniformly on compacts by neural networks with sigmoidal activation function of order $q \geq 2$ and architecture depending only on k and q.

Proof:

- Univariate basis splines $\mathcal{N}_{l,t,k}$ are linear combinations of translated and rescaled power units:

$$\mathcal{N}_{l,t,k}(x) = \mathcal{N}_k(2^l(x-t)),$$

$$\mathcal{N}_k(x) = \frac{1}{(k-1)!} \sum_{l=0}^{k} (-1)^l \binom{k}{l} (x-l)_+^{k-1}.$$

- Approximate the power units by multi-layer perceptrons, apply translations and scalings using the subsequent lemma, and take linear combinations. □

Shifting and rescaling neural networks

Lemma (Shifting and rescaling neural networks)

Let Φ be a neural networks of input dimension $d \in \mathbb{N}$.

For any $t \in \mathbb{R}^d$ and $\lambda \in \mathbb{R}$, there exists a neural network Ψ with the same architecture as Φ and at most d additional weights such that

$$R(\Psi)(x) = R(\Phi)(\lambda x + t) \qquad \text{for } x \in \mathbb{R}^d.$$

Proof:

- Define the neural network Ψ as

$$\Psi = \Phi \bullet ((\lambda \operatorname{Id}_{\mathbb{R}^d}, t))$$

- Count the number of layers and weights □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What are power units and how are they related to splines?
- Repetition: What are sigmoidal functions of higher order what are they useful for?
- Check: Verify the claims about uniform convergence on compacts of rescaled sigmoidal functions to power units!

Mathematics of Deep Learning, Summer Term 2020

Week 3, Video 4

Approximating Products by Multi-Layer Perceptrons

Philipp Harms Lars Niemann

University of Freiburg

Representing Products by Square Units

Theorem

Let $d \in \mathbb{N}$, and let ρ be the square unit $x \mapsto (x)_+^2$. Then there exists a neural network Φ with $\lceil \log_2(d) \rceil + 1$ layers such that

$$\mathrm{R}(\Phi)(x) = \prod_{i=1}^{d} x_i \qquad \textit{for } x \in \mathbb{R}^d.$$

Remark:

- Note that this representation is exact; no approximation is needed.
- However, approximation is needed to allow for more general activation functions.

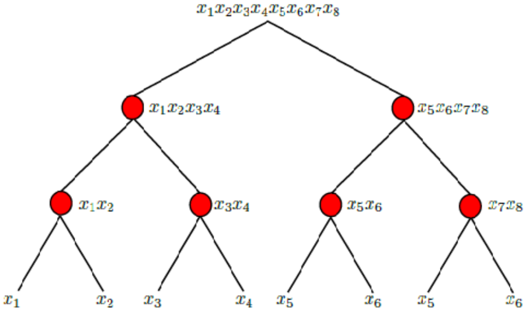
Proof: Representing Products by Square Units

Proof:

- Multiplication of 2 variables can be represented as a network of depth 2 and width 6 thanks to polarization:

$$2x_1x_2 = (x_1+x_2)_+^2 + (-x_1-x_2)_+^2 - (x_1)_+^2 - (-x_1)_+^2 - (x_2)_+^2 - (-x_2)_+^2$$

- Parallelize and concatenate to achieve multiplication of 2^n variables:



[Figure from Petersen]

□

Approximating Products by Multi-Layer Perceptrons

Corollary

Let $d \in \mathbb{N}$, and let ρ be sigmoidal of order $q \geq 2$. Then there exists a constant C such that for every $\epsilon, K > 0$, there exists a neural network Φ with $\lceil \log_2(d) \rceil + 1$ layers and C weights satisfying

$$\sup_{x \in [-K,K]^d} \left| \mathrm{R}(\Phi)(x) - \prod_{i=1}^{d} x_i \right| \leq \epsilon.$$

Proof:

- Represent the product by a network with square-unit activation function as above.
- Approximate each square unit (i.e., each red dot in the previous figure) by a 2-layer network of fixed size and note that this does not increase the overall network depth. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How can the product of two or more variables be represented or approximated by multi-layer perceptrons?
- Check: What does the multiplication network look like when the number of variables is not a power of 2?
- Discussion: Is it possible to build multiplication networks with activation function $x \mapsto x^2$?

Mathematics of Deep Learning, Summer Term 2020

Week 3, Video 5

Approximating Multivariate Splines by Multi-Layer Perceptrons

Philipp Harms   Lars Niemann

University of Freiburg

Approximating Multivariate Basis Splines by MLPs

Theorem

Let $k, d \in \mathbb{N}$*, and let* $\rho \colon \mathbb{R} \to \mathbb{R}$ *be sigmoidal of order* $q \geq 2$*. Then there exists a constant* $C > 0$ *such that for every basis spline* $f \in \mathcal{B}^k$ *and every* $\epsilon, K > 0$ *there is a neural network* Φ *with* $\lceil \log_2(d) \rceil + \left\lceil \max\{\log_q(k-1), 0\} \right\rceil + 1$ *layers and* C *weights satisfying*

$$\|\mathrm{R}(\Phi) - f\|_{L^\infty([-K,K]^d)} \leq \epsilon\,.$$

Proof: Approximating Multivariate Basis Splines by MLPs

Proof: Combine the approximations of power units and multiplication:

- Let $f \in \mathcal{B}^k$ be a dyadic basis spline, i.e.,

$$f(x) = \mathcal{N}^d_{l,t,k}(x) = \prod_{i=1}^{d} \mathcal{N}_k(2^l(x_i - t_i)) \qquad \text{for } x \in \mathbb{R}^d,$$

  where $\mathcal{N}_k$ is the univariate basis spline of order k, i.e.,

$$\mathcal{N}_k(x) := \frac{1}{(k-1)!} \sum_{l=0}^{k} (-1)^l \binom{k}{l} (x-l)_+^{k-1}$$

- Approximate the univariate basis splines $x_i \mapsto \mathcal{N}_k(2^l(x_i - t_i))$ by networks Ψ_i with $\lceil \max\{\log_q(k-1), 0\} \rceil + 1$ layers.
- Approximate multiplication $\mathbb{R}^d \to \mathbb{R}$ by a network Ψ_0 with $\lceil \log_2(d) \rceil + 1$ layers.
- Define $\Phi := \Psi_0 \bullet \mathrm{FP}(\Psi_1, \ldots, \Psi_d)$. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Outline the structure of the proof above: How can multivariate splines be approximated by multi-layer perceptrons?
- Discussion: Where is sigmoidality of higher order used?

Mathematics of Deep Learning, Summer Term 2020

Week 3, Video 6

Approximating Hölder Functions by Multi-Layer Perceptrons

Philipp Harms   Lars Niemann

University of Freiburg

Approximating Hölder Functions by MLPs

Theorem

Let $d \in \mathbb{N}$, $s > 0$, $r < s/d$, *and* $p \in (0, \infty]$. *Moreover, let* $\rho\colon \mathbb{R} \to \mathbb{R}$ *be sigmoidal of order* $q \geq 2$. *Then there exists a constant* $C > 0$ *such that, for every* f *in the unit ball of* $C^s([0,1]^d)$ *and every* $\epsilon \in (0, 1/2)$, *there exists a neural network* Φ *with depth* $L = \lceil \log_2(d) \rceil + \lceil \max\{\log_q(s-1), 0\} \rceil + 1$ *and number of weights* $M \leq C\epsilon^{-r}$ *satisfying*

$$\|f - \mathrm{R}(\Phi)\|_{L^p} \leq \epsilon.$$

- Deep networks are needed to approximate high-dimensional functions using sigmoidal activation functions of low order compared to the regularity of the function.
- The approximation rate is inversely proportional to the dimension d. This is often called the curse of dimensionality.

Proof: Approximating Hölder Functions by MLPs

Proof: Transfer of approximation:

- Let $\mathcal{C}$ be the unit ball in $C^s([0,1]^d)$, let $\mathcal{H} := L^p([0,1]^d)$, and let $\mathcal{B}^k$ be the dictionary of dyadic basis splines.
- Multi-layer perceptrons of depth L with activation function ρ and at most M weights approximate any function in the dictionary $\mathcal{B}^k$ uniformly on compacts and consequently also in $\mathcal{H}$ to arbitrary accuracy.
- The dictionary $\mathcal{B}^k$ approximates the signal class $\mathcal{C}$ at rate $(n^{-r})_{n\in\mathbb{N}}$.
- By the transfer-of-approximation theorem,

$$\sigma(\{\mathrm{R}(\Phi) : \mathrm{L}(\Phi) = L, \mathrm{M}(\Phi) \leq Mn\}, \mathcal{C}) \leq \sigma(\Sigma_n(\mathcal{B}^k), \mathcal{C}) \lesssim n^{-r}.$$

- Equivalently, an error of ϵ can be achieved using networks with $\mathcal{O}(\epsilon^{-1/r})$ weights. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Explain dictionary learning in the context of splines and Hölder functions.
- Discussion: What are strengths and weaknesses of the result when applied to function approximation or encoding?

Mathematics of Deep Learning, Summer Term 2020

Week 3, Video 7

Wrapup

Philipp Harms Lars Niemann

University of Freiburg

Outlook on this week's discussion and reading session

- Reading:
  - Oswald (1990): On the degree of nonlinear spline approximation in Besov-Sobolev spaces
  - DeVore (1998): Nonlinear approximation

Summary by learning goals

Having heard this lecture, you can now ...

- Describe signal classes, dictionaries, and related notions of approximation and transfer of approximation.
- Approximate products and power units by multi-layer perceptrons.
- Establish approximation rates for Hölder functions by multi-layer perceptrons.