Mathematics of Deep Learning, Summer Term 2020

Week 1

Deep Learning as Statistical Learning

Philipp Harms Lars Niemann

University of Freiburg

Overview of Week 1

1. Motivation for Deep Learning
2. Introduction to Statistical Learning
3. Empirical risk minimization and related algorithms
4. Error decompositions
5. Error trade-offs
6. Error bounds
7. Organizational Issues
8. Wrapup

Acknowledgement of Sources

Sources for this lecture:

- Frank Hutter and Joschka Boedecker (Department of Computer Science, Freiburg): Course on Deep Learning.
- Bousquet, Boucheron, and Lugosi (2003): Introduction to statistical learning theory.
- Vapnik (1999): An overview of statistical learning theory.

Mathematics of Deep Learning, Summer Term 2020

Week 1, Video 1

Motivation for Deep Learning

Philipp Harms Lars Niemann

University of Freiburg

Deep Learning in the News

The New York Times

Science

WORLD | U.S. | N.Y. / REGION | BUSINESS | TECHNOLOGY | SCIENCE | HEALTH | SPORTS | OPINION

ENVIRONMENT | SPACE & COSMOS

Scientists See Promise in Deep-Learning

A voice recognition program translated a speech given by Richard F. Rashid Chinese.

By JOHN MARKOFF
Published: November 23, 2012

Using an artificial intelligence technique inspired by theories al the brain recognizes patterns, technology companies are reportin gains in fields as diverse as computer vision, speech recognitior

THE NEW YORKER

NEWS | CULTURE | BOOKS & FICTION | SCIENCE & TECH | BUSINESS | HUMOR | MAGAZINE | ARCHIVE | SUBSCRIBE

NOVEMBER 25, 2012

IS "DEEP LEARNING" A REVOLUTION IN ARTIFICIAL INTELLIGENCE?

BY GARY MARCUS

Can a new technique known as deep learning revolutionize artificial intelligence, as yesterday's front-page article at the New York *Times* suggests? There is good reason to be excited about deep learning, a sophisticated "machine learning" algorithm that far exceeds many of its predecessors in its abilities to recognize syllables and images. But there's also good reason to be skeptical. While the *Times* reports that "advances in an artificial intelligence technology that can recognize patterns offer

MIT Technology Review

10 BREAKTHROUGH TECHNOLOGIES 2013

Introduction | The 10 Technologies
Past Years

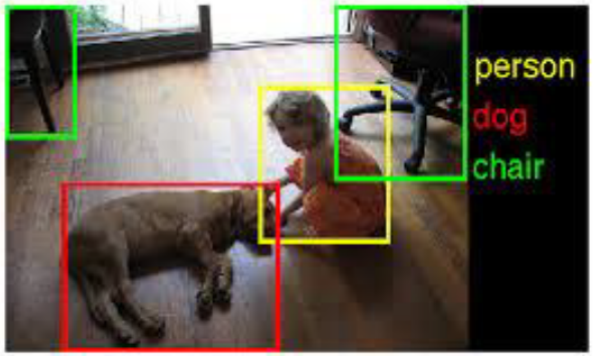
Deep Learning

With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.

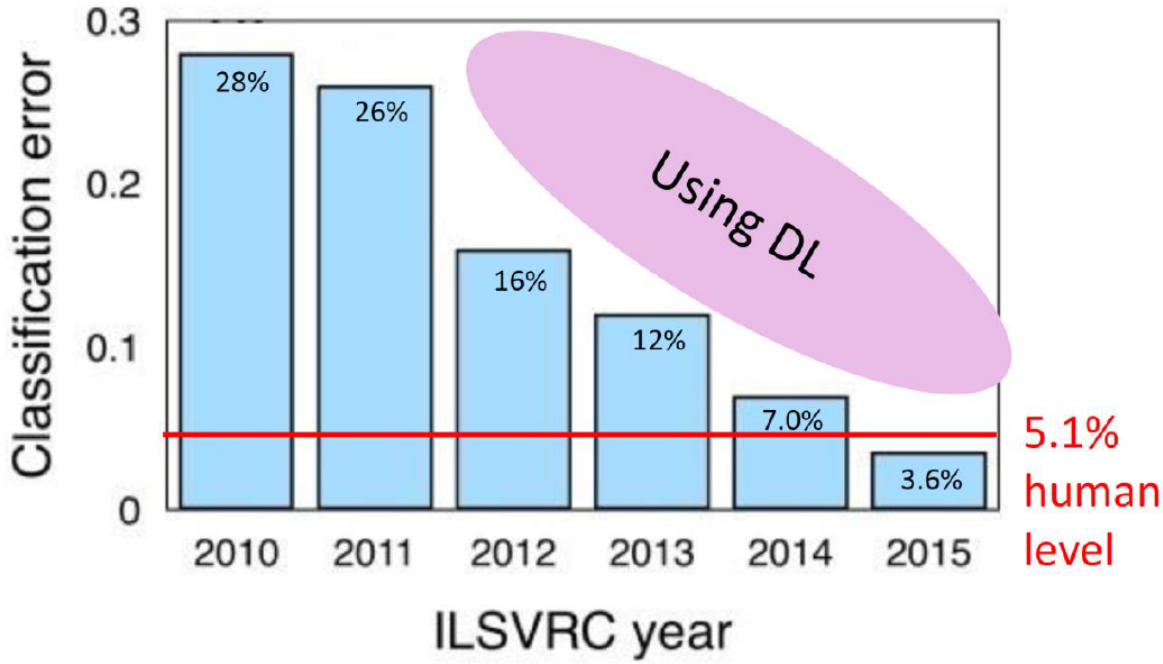
Deep Learning Revolutionized Computer Vision

- Excellent empirical results

Object recognition



Self-driving cars

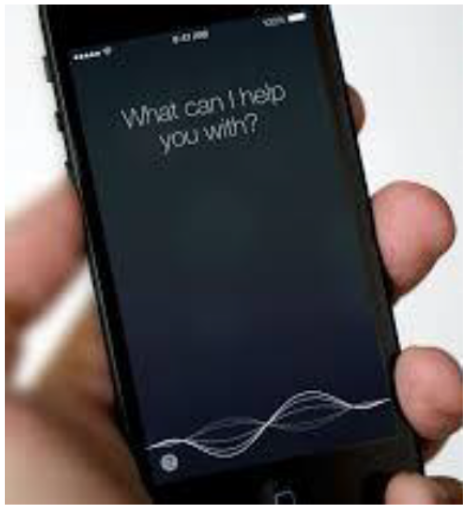


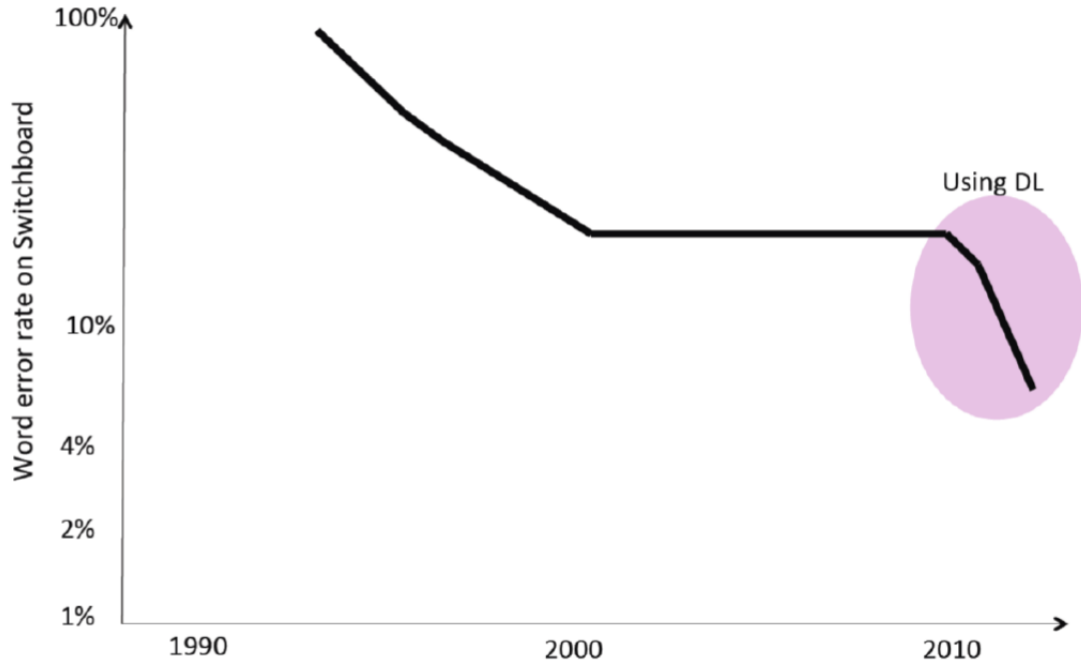
ILSVRC: ImageNet Large-Scale Visual Recognition Challenge

Deep Learning Revolutionized Speech Recognition

- Excellent empirical results

Speech recognition





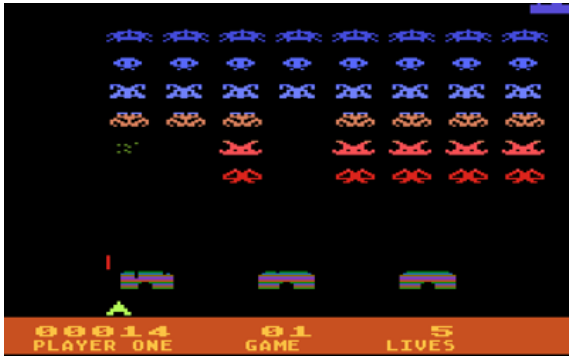
Auto-Translator

Image credit: Yoshua Bengio (data from Microsoft speech group)

Deep Learning Goes Great with Reinforcement Learning

- Excellent empirical results obtained by deep reinforcement learning

  - Superhuman performance in playing Atari games
    [Mnih et al, Nature 2015]



  - Beating the world's best Go player
    [Silver et al, Nature 2016]

(Deep) Learning as A Different Way of Programming

- We don't understand how the human brain solves certain problems
  - Face recognition
  - Playing Atari games
  - Speech recognition
  - Picking the next move in the game of Go
- We can nevertheless learn these tasks from data/experience
- If the task changes, we simply re-train

Deep Learning Allows Many Branches of AI to Converge

- Deep learning is now the principle approach in many different branches of AI:
  - Computer vision
  - Speech recognition
  - Natural language processing
  - (Robotics)

- The same general techniques apply in all of these fields
  - Amazing potential for cross-fertilization
  - Fields that drifted apart for decades have largely converged again
  - E.g., in Freiburg:
    - close collaboration & joint reading group between machine learning, computer vision, robotics, neurorobotics, and robot learning

Further Reasons for the Popularity of Deep Learning

- Very quick to get good results for some problems
  - Deep learning can handle raw data (images, speech, text, etc)
  - Very well-engineered libraries handle the complex underpinnings (Tensorflow, Pytorch, ...)
  - Very little machine learning knowledge is required to get started
- Misconception: "it works like the brain"
- Neural networks are very flexible models – this is the main content of the lecture

Understanding deep learning

- Neural networks are excellent function approximators
  - They are dense in many function spaces; this is often called the universal approximation property [Cybenko, Hornik]
  - Approximation rates are known for many shallow and deep network architectures

- However, this only partially explains their success
  - Generalization capability is needed in addition to approximation capability
  - Deep learning performs better than the theory predicts; this is the oft-quoted unreasonable effectiveness of deep learning in artificial intelligence [Sejnowski]

- Many interesting mathematical questions remain
  - Mathematicians are ideally prepared for appreciating the abstract issues involved in high-dimensional data analysis [Donoho]

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Why is deep learning so popular?
- Discussion:
  What might a mathematical theory of deep learning look like?
- Relation to your interests:
  What would you like to learn from this lecture?

Mathematics of Deep Learning, Summer Term 2020

Week 1, Video 2

Introduction to Statistical Learning

Philipp Harms Lars Niemann

University of Freiburg

Learning

Learning or, more precisely, inductive inference:

- Observe a phenomenon
- Construct a model of that phenomenon
- Make predictions using this model

Goals of learning theory and machine learning:

- Machine learning: automize inference
- Statistical learning theory: formalize inference

> *Nothing is more practical than a good theory.* *[Vapnik, Statistical Learning Theory 1998]*

Main assumption of statistical learning theory:

- Test and training data are iid.
- This distinguishes it from time series analysis (not independent) and transfer learning (not the same distribution).

Formalization

- Input and output spaces: measurable spaces $\mathcal{X}$ and $\mathcal{Y}$.
- Loss function: a measurable function $L\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$.
- Hypothesis class (aka. model class): a set H_0 of measurable functions $f\colon \mathcal{X} \to \mathcal{Y}$.
- Observations: independent random variables (X_1, Y_1), $\ldots$, (X_n, Y_n), defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, distributed according to a probability measure P on $\mathcal{X} \times \mathcal{Y}$.
- Objective: Find a function $f \in H_0$ with low or minimal risk (aka. test or generalization risk)

$$R(f) := \int L(f(x), y) P(dx, dy)$$

in the situation where P is unknown and the only information is contained in the observations.

Remarks

Applications:

- Regression: $\mathcal{Y} = \mathbb{R}$ and $L(y_1, y_2) = (y_1 - y_2)^2$.
- Classification: $\mathcal{Y} = \{0, 1\}$ and $L(y_1, y_2) = \mathbb{1}_{\{y_1 \neq y_2\}}$.

Useful hypothesis classes:

- Linear functions, polynomials, C^k functions, splines, or, as in deep learning, multilayer perceptrons.

Main challenge:

- The distribution P of the data and consequently also the risk functional R, which is to be minimized, are unknown.
- Otherwise this would be a standard optimization problem.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Describe the setup and goal of statistical learning theory.
- Discussion: Which aspects of machine learning are well-described by statistical learning theory? Which aren't?

Mathematics of Deep Learning, Summer Term 2020

Week 1, Video 3

Empirical risk minimization and related algorithms

Philipp Harms Lars Niemann

University of Freiburg

Risk versus empirical risk

Risk: Recall that...

- The objective in statistical learning theory is to minimize the risk

$$R(f) := \int L(f(x), y) P(dx, dy)$$

over all f in the hypothesis class H_0.

- The problem is that the distribution P of the data is unknown.

Empirical risk:

- As a substitute, define the empirical risk

$$R_n(f) := \frac{1}{n} \sum_{i=1}^{n} L(f(X_i), Y_i) = \int L(f(x), y) P_n(dx, dy),$$

where $P_n := \frac{1}{n} \sum_{i=1}^{n} \delta_{(X_i, Y_i)}$ is the empirical measure.

Algorithms

Empirical risk minimization (aka. supervised learning):

$$f_n \in \arg\min_{f \in H_0} R_n(f).$$

Structural risk minimization:

$$f_n \in \arg\min_{\substack{k \in \mathbb{N} \\ f \in H_k}} R_n(f) + p(k, n),$$

for some increasing sequence $(H_k)_{k \in \mathbb{N}}$ of hypothesis classes and a penalty $p(k, n)$ for the size or capacity of the class.

Regularization:

$$f_n \in \arg\min_{f \in H_0} R_n(f) + \|f\|^2,$$

$$f_n \in \arg\min_{f \in H_0} R_n(f) + \|f\|^2 = \arg\max_{f \in H_0} e^{-R_n(f) - \|f\|^2},$$

for some suitable norm $\|\cdot\|$ (or some other form of penalty).

Algorithms (cont.)

Maximum likelihood:

$$f_n \in \arg\max_{f \in H_0} e^{-R_n(f)} p(f) = \arg\min_{f \in H_0} R_n(f) - \log p(f),$$

where $p\colon H_0 \to \mathbb{R}_+$ is a probability density with respect to some reference measure π on H_0.

Posterior mean:

$$f_n = \frac{1}{Z_n} \int_{H_0} f e^{-R_n(f)} p(f) \pi(df),$$

where $Z_n := \int_{H_0} e^{-R_n(f)} p(f) \pi(df)$ is a normalizing factor.

Gibbs sampling:

$$f_n \sim \frac{1}{Z_n} e^{-R_n} p\pi.$$

Questions to Answer for Yourself / Discuss with Friends

- Transfer (optimization): What algorithms could be used to solve the empirical risk minimization problem?
- Transfer (statistics): What do the law of large numbers and the central limit theorem say about the convergence of $R_n(f)$ to $R(f)$ for fixed $f \in H_0$?

Mathematics of Deep Learning, Summer Term 2020

Week 1, Video 4

Error decompositions

Philipp Harms Lars Niemann

University of Freiburg

Error decompositions

Notation: $\mathbb{E}$ and E denote expectations w/r to $\mathbb{P}$ and P, respectively, and:

- f^* solves $R(f^*) = \inf_{f\colon \mathcal{X}\to\mathcal{Y}} R(f)$,
- f_0 solves $R(f_0) = \inf_{f\in H_0} R(f)$, and
- f_n is an H_0-valued random variable.

Approximation and estimation error:

$$R(f_n) = \underbrace{R(f^*)}_{\text{statistical risk}} + \underbrace{\big(R(f_0) - R(f^*)\big)}_{\text{approximation error}} + \underbrace{\big(R(f_n) - R(f_0)\big)}_{\text{estimation error}}$$

Empirical risk and generalization error:

$$R(f_n) = \underbrace{R_n(f_n)}_{\text{empirical risk}} + \underbrace{\big(R(f_n) - R_n(f_n)\big)}_{\text{generalization error}}$$

Bias and variance: for $\mathcal{Y} = \mathbb{R}$ and $L(y_1, y_2) = (y_1 - y_2)^2$,

$$\mathbb{E}[R(f_n)] = \underbrace{R(f^*)}_{\text{statistical risk}} + E\Big[\underbrace{\mathbb{E}[f_n(x) - f^*(x)]^2}_{\text{bias}} + \underbrace{\mathbb{V}\mathrm{ar}[f_n(x)]}_{\text{variance}}\Big]$$

Proof of the bias-variance decomposition

Recall:

- $R(f^*) := \inf_{f\colon \mathcal{X}\to\mathcal{Y}} R(f)$.
- $\mathcal{Y} = \mathbb{R}$, $L(y_1, y_2) = (y_1 - y_2)^2$.

Mean-square optimality of the mean: $f^*(x) = E[y|x]$.

Conditional risk of f_n given (x, ω):

$$\begin{aligned} E[(f_n(x) - y)^2 \mid x] &= \mathrm{Var}[f_n(x) - y \mid x] + E[f_n(x) - y \mid x]^2 \\ &= E[(f^*(x) - y)^2 \mid x] + (f_n(x) - f^*(x))^2. \end{aligned}$$

Expected risk of f_n:

$$\begin{aligned} \mathbb{E}[R(f_n)] &= R(f^*) + E\big[\mathbb{E}[(f_n(x) - f^*(x))^2]\big] \\ &= R(f^*) + E\big[\mathbb{E}[f_n(x) - f^*(x)]^2 + \mathbb{V}\mathrm{ar}[f_n(x)]\big]. \end{aligned}$$

□

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Visualize the approximation, estimation, and generalization error in a drawing.
- Discussion: Can you guess which error terms increase or decrease with respect to H_0 and n?

Mathematics of Deep Learning, Summer Term 2020

Week 1, Video 5

Error trade-offs

Philipp Harms Lars Niemann

University of Freiburg

Error trade-offs

Decompositions versus trade-offs

- A trade-off occurs when one term in an error decomposition increases while another term decreases with respect to a parameter.

Trade-offs in the choice of hypothesis class?

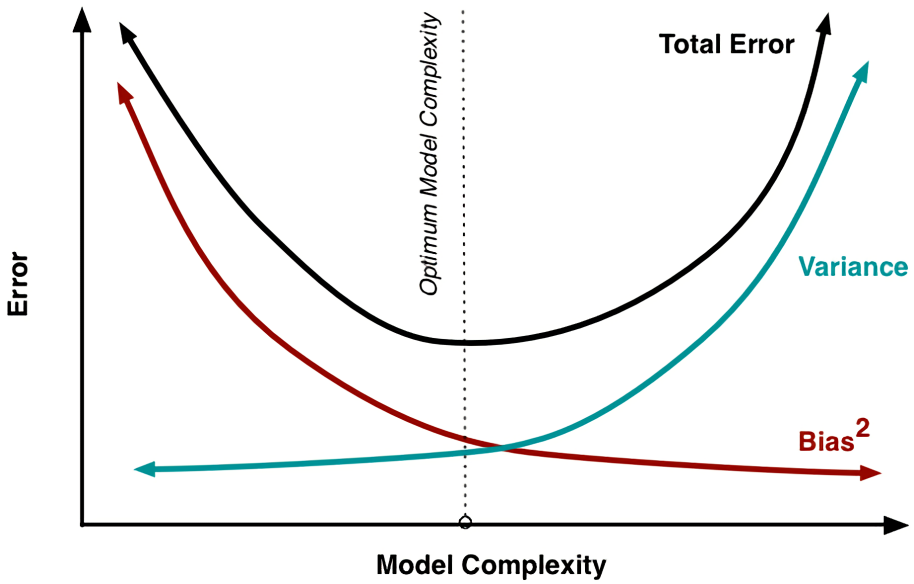
- In general, there is no trade-off in the above error decompositions with respect to H_0.
- However, there may be trade-offs with respect to H_0 in error bounds (as opposed to the error itself).

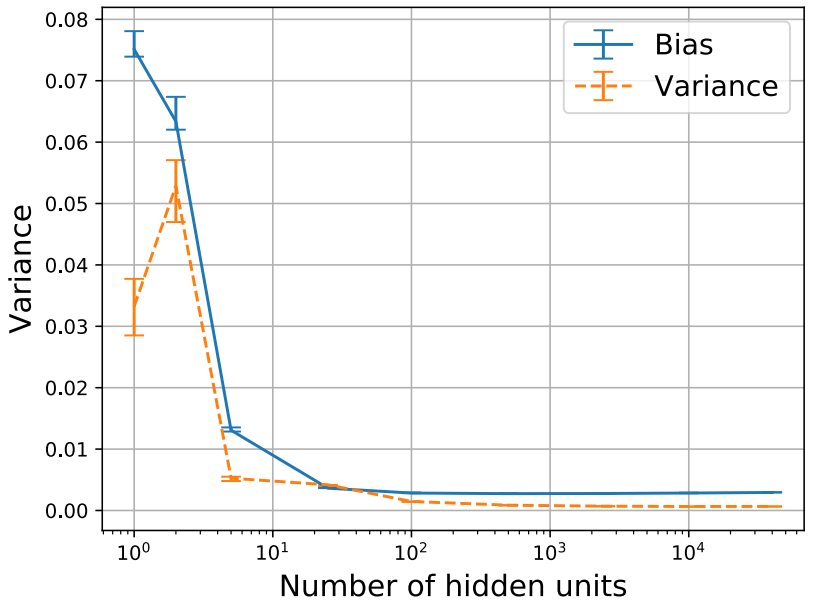
Example: bias-variance decomposition

- Conventional wisdom: The price to pay for achieving low bias is high variance—a trade-off in the choice of H_0. [Geman et al. 1992].
- However, this is false in over-parameterized regimes, which are common in modern machine learning applications (see next slide).

Example: bias-variance decomposition

Traditional view of the bias-variance trade-off (left) versus lack of any trade-off in MNIST character recognition using sufficiently wide ReLu networks (right).

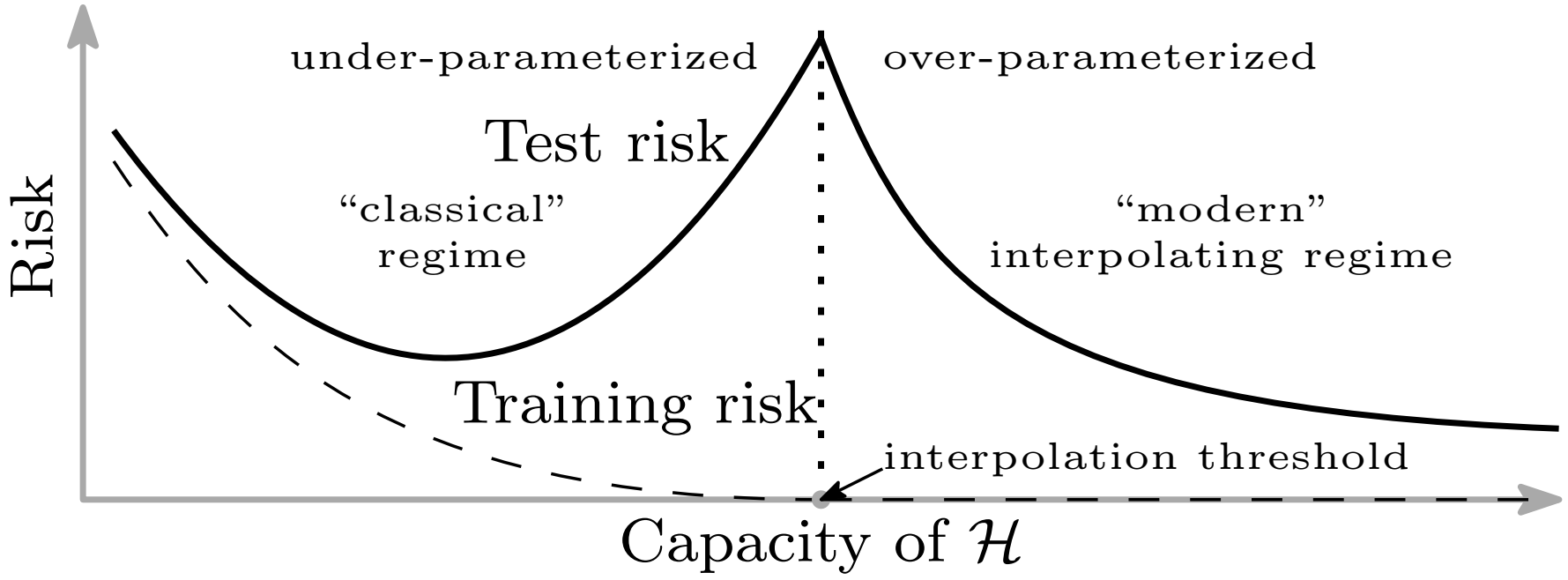




[Figures from Neal 2019]

Example: bias-variance decomposition (cont.)

Conjectured reconciliation: U-shaped risk curve in the underparameterized regime and decreasing risk in the overparameterized regime [Belkin e.a. 2019]



[Figure from Belkin e.a. 2019]

Questions to Answer for Yourself / Discuss with Friends

- Discussion: Can you think of a reason (or an example) why the variance might be decreasing in over-parameterized regimes?

Mathematics of Deep Learning, Summer Term 2020

Week 1, Video 6

Error bounds

Philipp Harms Lars Niemann

University of Freiburg

Bounding the approximation error

Notation:

- f^* solves $R(f^*) = \inf_{f\colon \mathcal{X}\to\mathcal{Y}} R(f)$, and
- f_0 solves $R(f_0) = \inf_{f\in H_0} R(f)$.

Approximation error: $R(f_0) - R(f^*)$

- Decreases when H_0 increases.
- Depends on how closely f^* can be approximated by functions in H_0.
- Is the main focus of this lecture.

Bound for quadratic loss functions:

$$\begin{aligned} 0 \le R(f_0) - R(f^*) &= E\big[(f_0(x) - y)^2 - (f^*(x) - y)^2\big] \\ &= E\big[(f_0(x) + f^*(x) - 2y)(f_0(x) - f^*(x))\big] \\ &\le E\big[|f_0(x) + f^*(x) - 2y|\big] \sup_{x\in\mathcal{X}} \big|f_0(x) - f^*(x)\big|. \end{aligned}$$

Bounding the generalization error

Notation:

- $R(f) = \int L(f(x), y) P(dx, dy)$,
- $R_n(f) = \int L(f(x), y) P_n(dx, dy)$, and
- f_n is a random element of H_0.

Generalization error: $R(f_n) - R_n(f_n)$

- Is the difference between a mean and an empirical mean:

$$R(f_n) - R_n(f_n) = \int L(f_n(x), y)(P - P_n)(dx, dy).$$

- Is of order $n^{-1/2}$ by the central limit theorem for fixed $f_n \equiv f$.

Uniform generalization error: $\sup_{f \in H_0} |R(f) - R_n(f)|$

- Increases when H_0 increases.
- Is the main focus of statistical learning theory.

Bounding the estimation error

Notation:

- $R(f) = \int L(f(x), y) P(dx, dy)$,
- $R_n(f) = \int L(f(x), y) P_n(dx, dy)$, and
- f_n is a random element of H_0.

Estimation error: $R(f_n) - R(f_0)$

- Is bounded by twice the uniform generalization error if f_n minimizes the empirical risk:

$$\cdots \leq \underbrace{R(f_n) - R_n(f_n)}_{\text{generalization error}} + \underbrace{R_n(f_n) - R_n(f_0)}_{\leq 0} + \underbrace{R_n(f_0) - R(f_0)}_{\text{generalization error}}.$$

A glimpse into statistical learning theory

Höffding's inequality: for any function $g\colon \mathcal{X} \times \mathcal{Y} \to [a,b]$, one has the Gaussian tail estimate

$$\mathbb{P}\big[|P_n g - Pg| > \epsilon\big] \leq 2\exp\left(-\frac{2n\epsilon^2}{(b-a)^2}\right), \qquad \epsilon > 0.$$

Uniform risk bound: given $H_0 = \{f_1, \ldots, f_N\}$, assume that the losses $g_i := L(f_i(\cdot), \cdot)$ take values in $[a,b]$ and estimate

$$\begin{aligned}\mathbb{P}\left[\max_{f\in H_0} |R_n f - Rf| > \epsilon\right] &= \mathbb{P}\left[\max_{i\in\{1,\ldots,N\}} |P_n g_i - Pg_i| > \epsilon\right] \\ &\leq \sum_{i=1}^{N} \mathbb{P}\big[|P_n g_i - Pg_i| > \epsilon\big] \leq 2N\exp\left(-\frac{2n\epsilon^2}{(b-a)^2}\right).\end{aligned}$$

A glimpse into statistical learning theory

Expected risk: deduce convergence of order $n^{-1/2}$ via

$$\mathbb{E}\left[\max_{f\in H_0}|R_n f - Rf|\right] = \int_0^\infty \mathbb{P}\left[\max_{f\in H_0}|R_n f - Rf| > \epsilon\right] d\epsilon$$
$$\leq N(b-a)\sqrt{\frac{\pi}{2n}}.$$

Note that the right-hand side depends on the size N of H_0.

Extension to infinite sets H_0: Approximate H_0 by finite sets of indicator functions; the error can be controlled by the Vapnik–Cervonenkis (VC) dimension of H_0 or other capacity measures.

Further topics: unbounded loss functions and capacity measures for specific hypothesis classes such as indicator functions or neural networks.

Caveat: deep learning performs better than predicted by this theory—once more, the unreasonable effectiveness of deep learning...

Questions to Answer for Yourself / Discuss with Friends

- Discussion: Can you spot any points where the error analysis of statistical learning theory might leave room for improvements?
- Suggestion: Read up on Höffding's inequality and related large deviations results or concentration inequalities.

Mathematics of Deep Learning, Summer Term 2020

Week 1, Video 7

Organizational Issues

Philipp Harms Lars Niemann

University of Freiburg

Team

- Philipp Harms: Lecturer, main contact for lectures
  `www.stochastik.uni-freiburg.de/professoren/harms/philipp-harms`
- Jakob Stiefel: Teaching Assistant, main contact for exercises
- Lars Niemann: Teaching Assistant
  `www.stochastik.uni-freiburg.de/mitarbeiter/niemann`

Web links

- Lecture homepage for general information: `www.stochastik.uni-freiburg.de/lehre/ss-2020/vorlesung-deep-learning-ss-2020`
- ILIAS for slides, videos, forum, and exercises: `ilias.uni-freiburg.de/goto.php?target=crs_1542865&client_id=unifreiburg`
- BigBlueButton: virtual meeting room `vHarms` with password `vHarms20206` at `www.math.uni-freiburg.de/lehre/virtuelle_veranstaltungen.html`. Supported Browsers include Chrome and Firefox on desktops and Chrome and Safari on mobiles.
- HisInOne for administrative issues

Outlook on the lecture

- Approximation theory for neural networks
  - shallow/deep
  - feed-forward/residual/recurrent
- Using methods from
  - functional analysis
  - harmonic analysis
  - differential geometry
  - probability theory
  - stochastic analysis
- Further topics
  - For example, generalization capability, auto-encoders, variational auto-encoders, adversarial networks, etc.
  - Depending on your interests and how we do time-wise

Relation to other deep learning courses in Freiburg

- This course: mathematical aspects of deep learning
- At the Mathematical Institute:
  - Angelika Rohde's seminar about the mathematical foundations of statistical learning: `www.stochastik.uni-freiburg.de/professoren/rohde/teaching`
  - Next term: Thorsten Schmidt's lecture on Machine Learning
- At the Department of Computer Science: in the groups on
  - Computer Vision
  - Machine Learning
  - Statistical Pattern Recognition
  - Artificial Intelligence

Parts of the course

- Short videos and slides:
  - Available on ILIAS every Tuesday night
- Live discussion and further reading:
  - Wednesdays 14:15-14:45 via BigBlueButton
- Forum:
  - Available on ILIAS for questions of all kinds
  - Please answer a question if you know the answer
- Graded exercises:
  - Mathematical and programming tasks
  - Solutions to be uploaded to ILIAS every two weeks
  - Collaboration in groups of two is allowed and encouraged.
  - Groups cannot be changed during the term.

Requirements and exam

- Requirements:
  - Solid background in probability theory and functional analysis
  - Basic knowledge in differential equations and stochastic analysis.
  - Basic programming skills
- Oral exam:
  - 50% of exercise points required for participation
  - Scope: content covered in the lecture, live discussions, and exercises
  - Focus on conceptual understanding rather than learning by heart

Resources for Python

- Python tutorials
  - Official tutorial:
    `https://docs.python.org/3/tutorial/index.html`
  - For beginners: `www.learnpython.org/`
  - For programmers:
    `http://stephensugden.com/crash_into_python/`
  - Many more: http://docs.python-guide.org/en/latest/intro/learning/
- Python libraries:
  - Numpy: `http://wiki.scipy.org/Tentative_NumPy_Tutorial`
  - SciPy: `http://docs.scipy.org/doc/scipy/reference/tutorial/`
  - Matplotlib: `http://matplotlib.org/users/beginner.html`

Mathematics of Deep Learning, Summer Term 2020

Week 1, Video 8

Wrapup

Philipp Harms Lars Niemann

University of Freiburg

Summary by learning goals

Having heard this lecture, you can now . . .

- Describe why deep learning is so popular
- Formulate the basic principles of statistical learning theory
- Understand deep learning in the context of statistical learning theory

Outlook on this week's discussion and reading session

- Discussion:
  - Questions and feedback, in both directions
  - Administrative and IT issues, if any
- Reading: related original literature
  - Sejnowski (2020): The unreasonable effectiveness of deep learning in artificial intelligence
  - Donoho (2000): High-Dimensional Data Analysis—the Curses and Blessings of Dimensionality
  - Vapnik (1999): An overview of statistical learning theory
- Preparation:
  - Watch the videos of the week

Mathematics of Deep Learning, Summer Term 2020

Week 2

Neural Networks

Philipp Harms Lars Niemann

University of Freiburg

Overview of Week 2

Acknowledgement of Sources

Sources for this lecture:

- Frank Hutter and Joschka Boedecker (Department of Computer Science, Freiburg): Course on Deep Learning.
- Philipp Christian Petersen (Faculty of Mathematics, University of Vienna): Course on Neural Network Theory.

Mathematics of Deep Learning, Summer Term 2020

Week 2, Video 1

Multilayer Perceptrons

Philipp Harms Lars Niemann

University of Freiburg

McCulloch and Pitts Neuron

The first neural network was devised by McCulloch and Pitts (1943) in an attempt to model a biological neuron.

Definition

A McCulloch and Pitts neuron is a function of the form

$$\mathbb{R}^d \ni x \mapsto \rho\left(\sum_{i=1}^{d} w_i x_i - \theta\right) \in \mathbb{R}$$

where $d \in \mathbb{N}$, $\rho = \mathbb{1}_{\mathbb{R}_+} : \mathbb{R} \to \mathbb{R}$, and $w_i, \theta \in \mathbb{R}$.

- ρ is called activation function,
- θ is called threshold,
- w_i are called weights, and
- the neuron fires (i.e., returns 1) if the weighted sum of inputs exceeds the threshold.

Multilayer Perceptron

A multilayer perceptron, as introduced by Rosenblatt (1958), links multiple neurons together in the sense that the output of one neuron forms an input to another.

Definition

Let d, $L \in \mathbb{N}$, $L \geq 2$ and $\rho : \mathbb{R} \to \mathbb{R}$. Then a multilayer perceptron (MLP) with d-dimensional input, L layers, and activation function $\rho\colon \mathbb{R} \to \mathbb{R}$ is a function

$$F\colon \mathbb{R}^d \to \mathbb{R}^{N_L}, \qquad F = T_L \circ \rho \circ T_{L-1} \circ \cdots \circ \rho \circ T_1,$$

where ρ is applied coordinate-wise and $T_l\colon \mathbb{R}^{l-1} \to \mathbb{R}^l$ is affine, for each $l \in \{1, \ldots, L\}$ and $N_l \in \mathbb{N}$ with $N_0 = d$.

Recall that an affine map is of the form $x \mapsto Ax + b$ for a matrix A and vector b.

Multilayer Perceptron (cont.)

- In contrast to the McCulloch and Pitts neuron, we now allow arbitrary activation functions ρ.
- Notice that the MLP does not allow arbitrary connections between neurons, but only between those, that are in adjacent layers, and only from lower layers to higher layers.

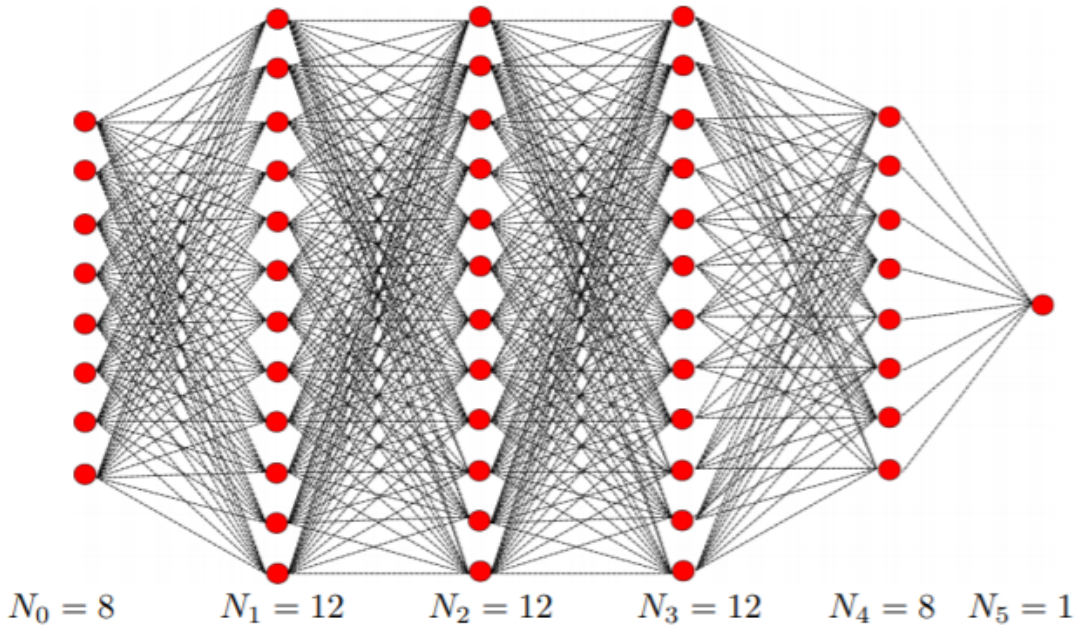


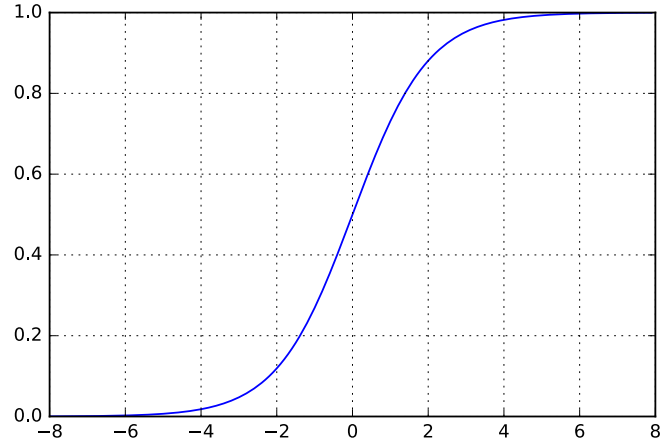
Illustration of a multi-layer perceptron with 5 layers. The red dots correspond to the neurons.

[figure from Petersen, Ch. 1]

Activation Functions - Examples

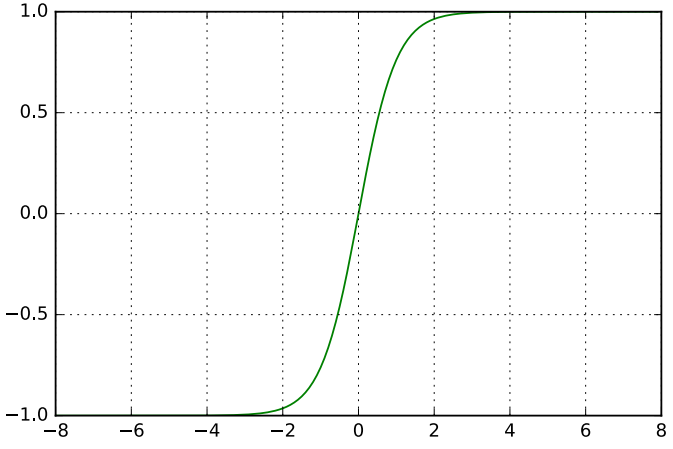
Logistic sigmoid activation function:

$$g_{logistic}(z) = \frac{1}{1 + \exp(-z)}$$



Logistic hyperbolic tangent activation function:

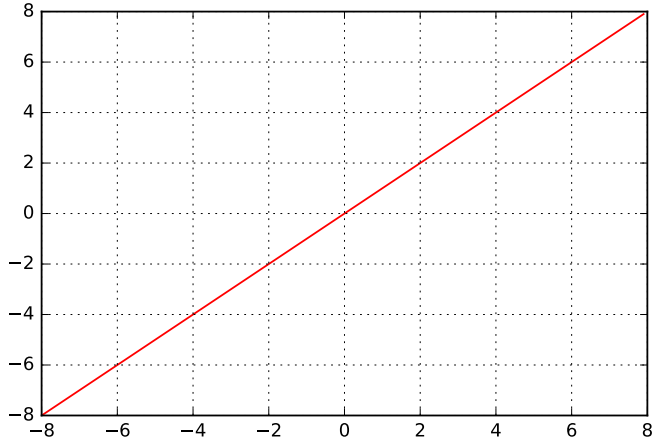
$$g_{tanh}(z) = \tanh(z)$$
$$= \frac{\exp(z) - \exp(-z)}{\exp(z) + \exp(-z)}$$



Activation Functions - Examples (cont.)

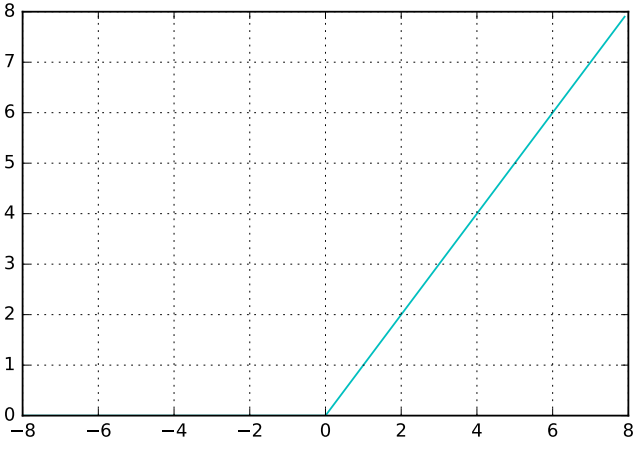
Linear activation function:

$$g_{linear}(z) = z$$



Rectified Linear (ReLU) activation function:

$$g_{relu}(z) = \max(0, z)$$



Deep learning

Definition

Deep learning is the use of multilayer perceptrons in learning tasks.

For example, supervised learning, i.e., empirical risk minimization:

- Given observations $(x_1, y_1), \ldots, (x_n, y_n)$,
- Find a multilayer perceptron f such that $f(x_i) \approx y_i$.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What is a multi-layer perceptron?
- Application of what you just learned:
  What class of functions is represented by multi-layer perceptrons with linear, polynomial, or ReLu activation functions?
- Transfer: How do multi-layer perceptrons differ from spline or finite element discretizations?

Mathematics of Deep Learning, Summer Term 2020

Week 2, Video 2

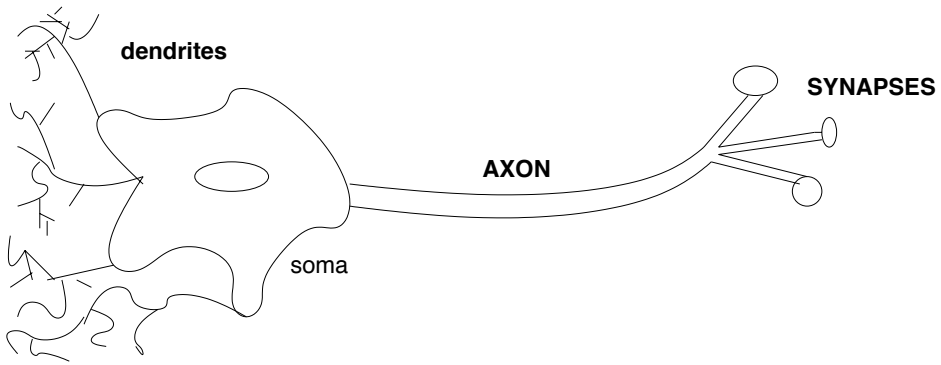
A Brief History of Deep Learning

Philipp Harms Lars Niemann

University of Freiburg

Biological Inspiration of Artificial Neural Networks

- Dendrites input information to the cell
- Neuron fires (has action potential) if a certain threshold for the voltage is exceeded
- Output of information by axon
- The axon is connected to dentrites of other cells via synapses
- Learning: adaptation of the synapse's efficiency, its synaptical weight



History of Deep Learning

Deep Learning has developed in several waves

The early days, under the name of artificial neural networks/cybernetics

- 1942 Artificial neurons as a model of brain function [McCulloch/Pitts]
- 1949 Hebbian learning [Hebb]
- 1958 Rosenblatt perceptron [Rosenblatt]
- 1960 Adaline $\rightarrow$ stochastic gradient descent [Widrow/Hoff]

The first time the popularity of NNs declined

- Negative result: linear models cannot represent the XOR function
- Backlash against biologically inspired learning [Minsky/Papert, 1969]

History of Deep Learning

1980 - early 2000s (under the name of connectionism)

- 1980 Neocognitron [Fukushima]
- 1986 Multilayer Perceptrons and backpropagation [Rumelhart et al.]
- 1989 Autoencoders [Baldi and Hornik],
  Convolutional neural networks [LeCun]
- 1997 LSTMs [Hochreiter and Schmidhuber]

The second time the popularity of NNs declined

- Ventures based on NNs made unrealistically ambitious claims
  - AI research could not fulfill these unreasonable expectations
- Other fields of machine learning made advances
  - E.g., SVMs and graphical models
  - SVMs were the state of the art on many datasets (data was small), specialized ConvNets held state of the art on MNIST but didn't scale

History of Deep Learning and ANNs (cont.)

Mid 2000s, the field got re-invigorated:

- Greedy layer-wise pretraining [Hinton, 2006]
  - It was now possible to train much deeper networks
- Several groups "resurrected" the idea of training large neural networks supervisedly using large amounts of data.
  - Most prominently [Krizhevsky et al., 2012] improved results on Imagenet benchmark by large margin
- Since then: exponential growth
  - NeurIPS attendance has grown exponentially
    - In 2018, it sold out in 12 minutes; lottery system since then
  - Some people are raising unrealistic expectations
  - Let's see how long this current wave persists

Questions to Answer for Yourself / Discuss with Friends

- Discussion: How long will the current deep learning wave persist?
  - What are reasons that it will continue?
  - What are reasons that it will end?

Mathematics of Deep Learning, Summer Term 2020

Week 2, Video 3

Deep Learning as Representation Learning

Philipp Harms Lars Niemann

University of Freiburg

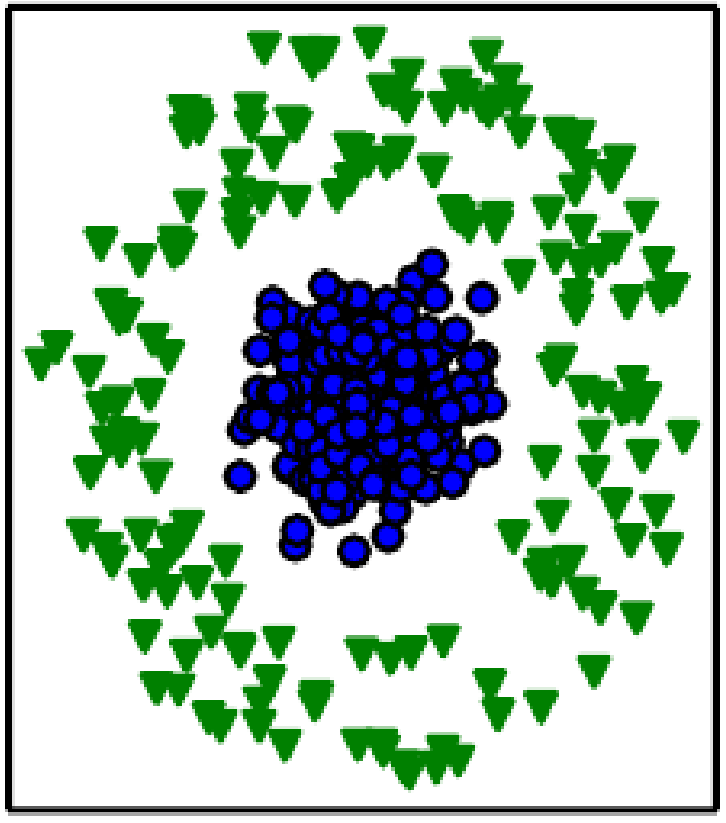
Some terminology

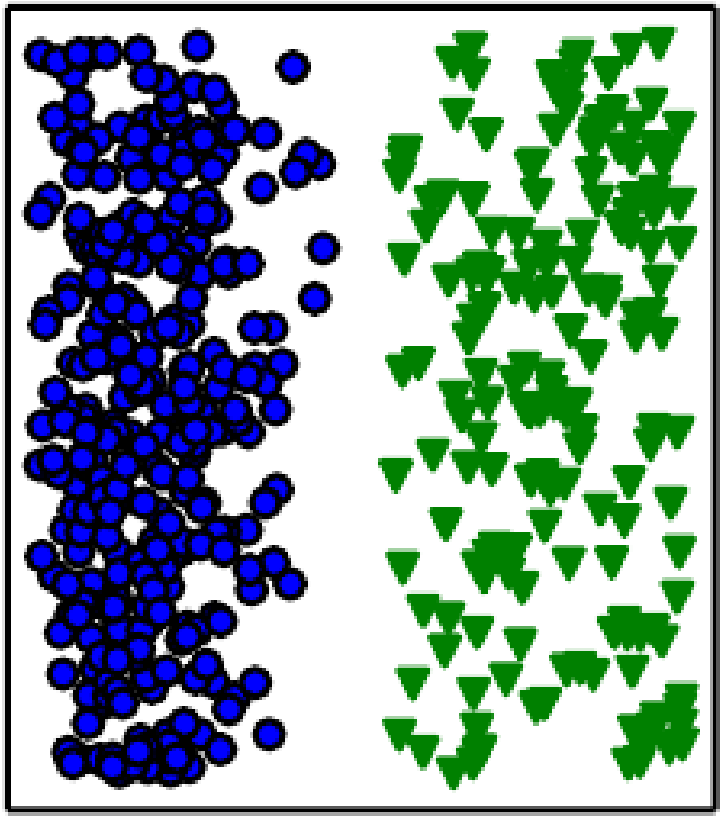
- Supervised learning: given data (x_i, y_i), find a function f such that $f(x_i) \approx y_i$
- Classification: special case where f is an indicator function (aka. classifier) and y_i belong to $\{0, 1\}$
- Data representation: a coordinate system for x
- Feature: a coordinate
- Linearly separable: y_i equals the sign of a linear functional of x_i

Definition: Representation learning

Representation learning

"a set of methods that allows a machine to be fed with raw data and to automatically discover the representations needed for detection or classification" - LeCun et al., 2015





Example for a Poor Representation: Roman Numbers

In particular, poor for the task of addition.
E.g., perform CCCLXIX + DCCCXLV (369 + 845)

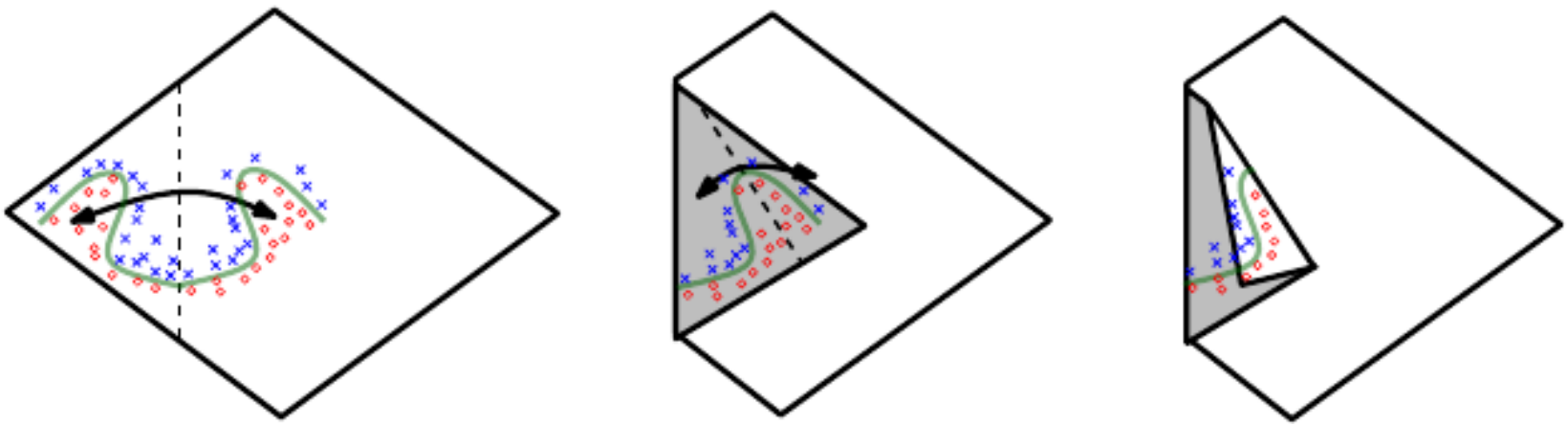
1. Substitute for any subtractives : CCCLXVIIII + DCCCXXXXV
2. Concatenate: CCCLXVIIIIDCCCXXXXV
3. Sort : DCCCCCCLXXXXXVVIIII
4. Combine groups to obtain:
   DCCCCCCLXXXXXXIIII
   DCCCCCCLLXIIII
   DCCCCCCCXIIII
   DDCCXIIII
   MCCXIIII
5. Re-Substitute any subtractives:
   MCCXIV

In contrast, converting to our current number system: $369 + 845 = 1214$.

Definition: Deep learning

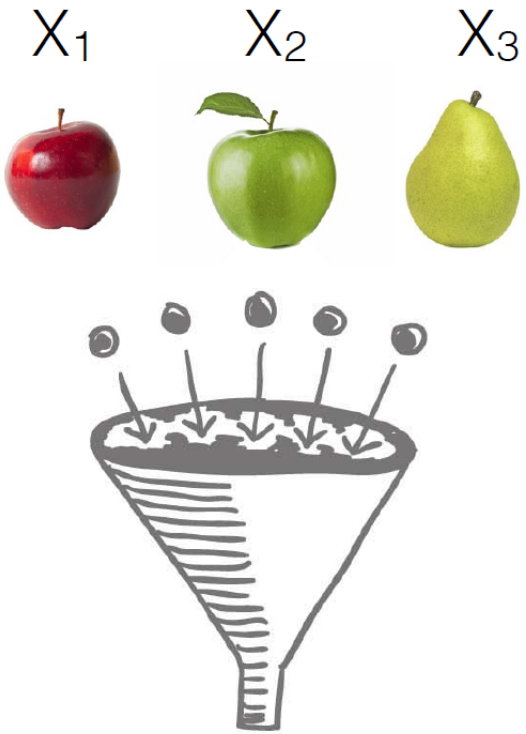
Deep learning

"representation learning methods with multiple levels of representation, obtained by composing simple but nonlinear modules that each transform the representation at one level into a [...] higher, slightly more abstract (one)" - LeCun et al., 2015

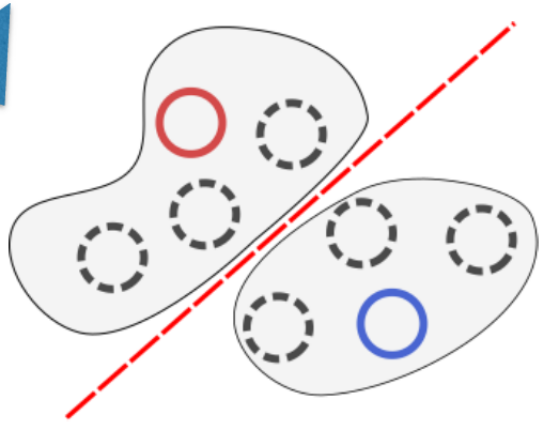


Standard Machine Learning Pipeline

- Standard machine learning algorithms are based on high-level attributes or features of the data
- They require (often substantial) feature engineering, i.e., extraction and selection of features.

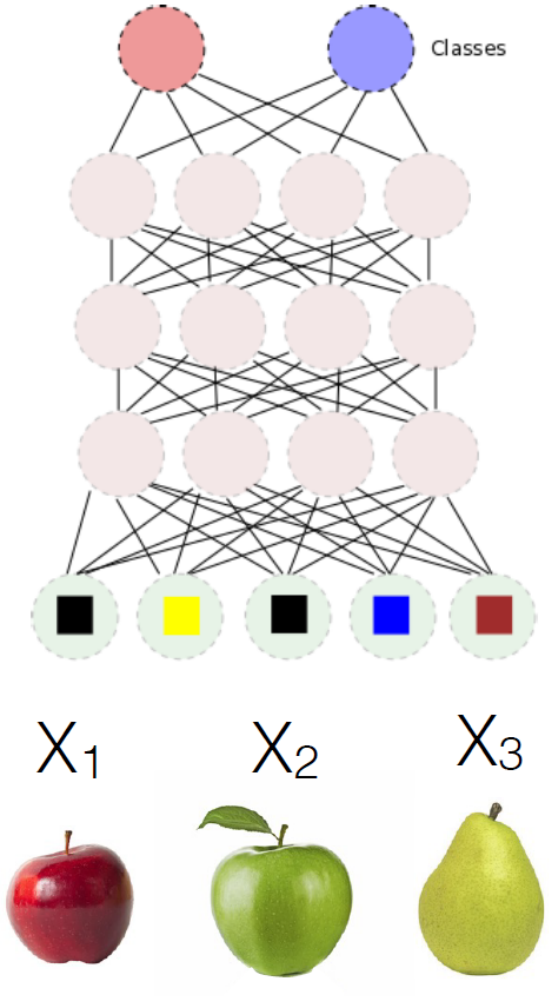


	Merkmale
X_1	rot, 3.5 cm
X_2	grün, 4 cm
X_3	grün, 10 cm

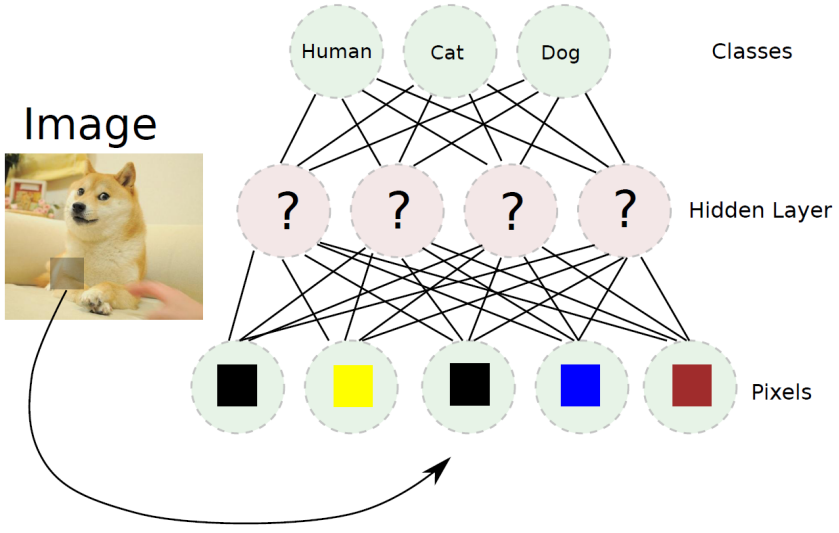


Representation Learning Pipeline

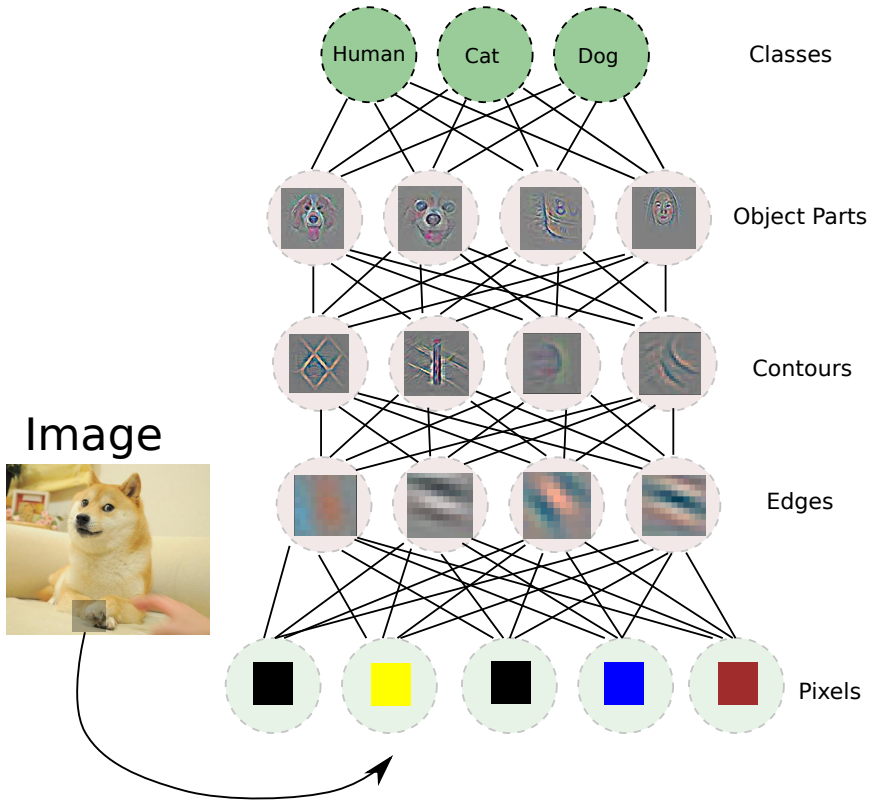
- Jointly learn features and classifier, directly from raw data
- This is also referrred to as end-to-end learning



Shallow vs. Deep Learning



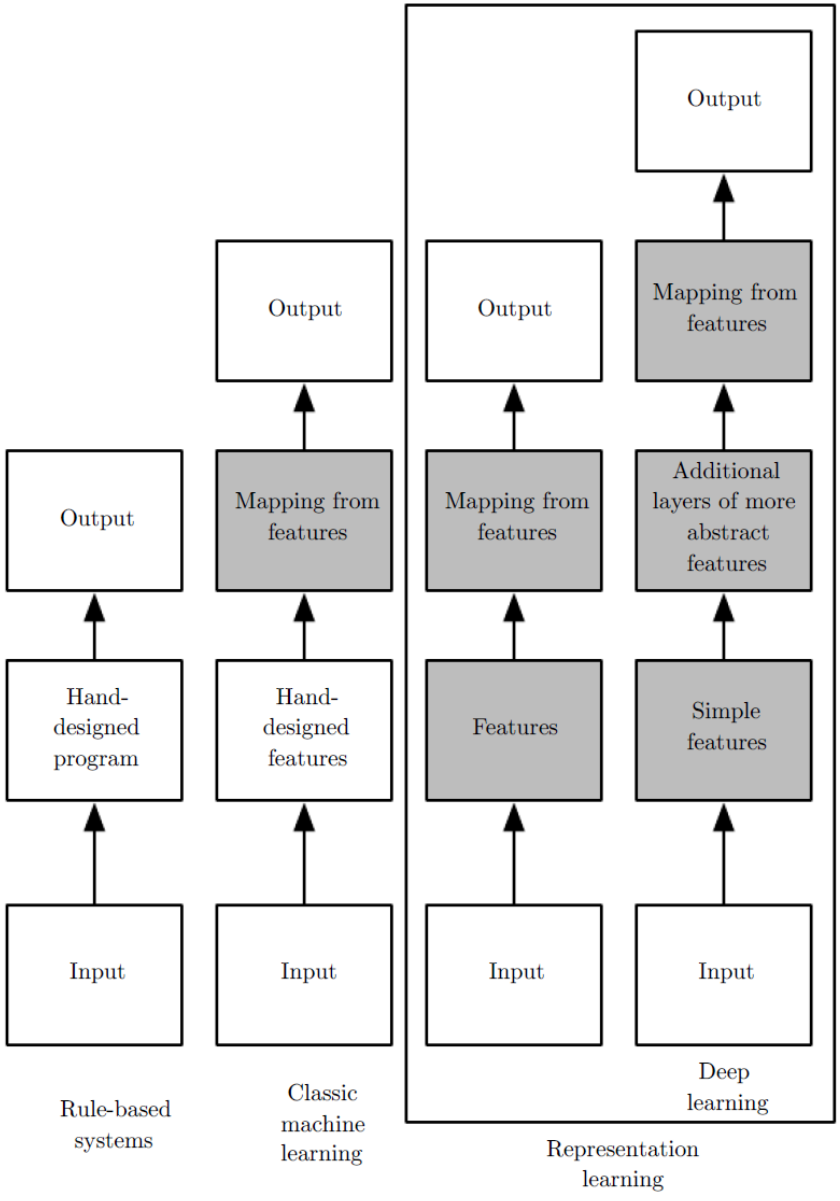
Shallow vs. Deep Learning



[Visualizations of network activations taken from Zeiler [2014]]

- Deep Learning: learning a hierarchy of representations that build on each other, from simple to complex
- Features are learned in an end-to-end fashion, from raw data

Relation to More Traditional Learning Approaches



Questions to Answer for Yourself / Discuss with Friends

- Relation to your interests:
  What would be a good and a bad representation for a problem you find interesting?

- Discussion: Are deep networks always better than shallow ones?

Mathematics of Deep Learning, Summer Term 2020

Week 2, Video 4

Definition of Neural Networks

Philipp Harms Lars Niemann

University of Freiburg

Neural Networks: Definition

Definition

Let $d, L \in \mathbb{N}$. A neural network with input dimension d and L layers is a sequence of matrix-vector tuples

$$\Phi = \big((A_1, b_1), (A_2, b_2), \ldots, (A_L, b_L)\big),$$

where $N_0 := d$, $N_1, \ldots, N_L \in \mathbb{N}$, $A_l \in \mathbb{R}^{N_{l-1} \times N_l}$, and $b_l \in \mathbb{R}^{N_l}$ for $l \in \{1, \ldots, L\}$.

- According to this definition, neural networks are the coefficients of multi-layer perceptrons.
- This distinction is useful but not always made in the literature.

Neural Networks: Definition (cont.)

Definition

The realization of a neural network Φ with activation function $\rho\colon \mathbb{R} \to \mathbb{R}$ is the function

$$\mathrm{R}(\Phi)\colon \mathbb{R}^d \to \mathbb{R}^{N_L}, \quad \mathrm{R}(\Phi)(x) \coloneqq x_L,$$

where the output x_L results from

$$\begin{aligned} x_0 &\coloneqq x, \\ x_l &= \rho(A_l x_{l-1} + b_l) \text{ for } l \in \{1, \dots, L-1\}, \\ x_L &\coloneqq A_L x_{L-1} + b_L. \end{aligned}$$

Here ρ is understood to act component-wise.

- Thus, a multilayer perceptron is the realisation of a neural network.

Neural Networks: Definition (cont.)

Definition

We call $\mathrm{N}(\Phi) := d + \sum_{l=1}^{L} N_l$ the number of neurons, $\mathrm{L}(\Phi) := L$ the number of layers or depth, and

$$\mathrm{M}(\Phi) := \sum_{l=1}^{L} \mathrm{M}_l := \sum_{l=1}^{L} \|A_l\|_0 + \|b_l\|_0$$

the number of weights. Here $\|\cdot\|_0$ denotes the number of non-zero entries of a matrix or vector.

Neural Networks: Definition (cont.)

Definition

Let $L \in \mathbb{N}$. A vector $S = (N_0, \ldots, N_L) \in \mathbb{N}^{L+1}$ is called architecture of a neural network

$$\Phi = ((A_1, b_1), \ldots, (A_L, b_L))$$

if $A_l \in \mathbb{R}^{N_{l-1} \times N_l}$ for $l = 1, \ldots, L$. Given such a vector S, we denote by $\mathcal{NN}(S)$ the set of all neural networks with architecture S.

Note: $\mathcal{NN}(S)$ is a finite-dimensional linear space.

Questions to Answer for Yourself / Discuss with Friends

- Check: Is $\|\cdot\|_0$ a norm?
- Repetition: What are neural networks, and how do they differ from multi-layer perceptrons?
- Discussion: Is the realization map continuous in some sense?

Mathematics of Deep Learning, Summer Term 2020

Week 2, Video 5

Operations on Neural Networks

Philipp Harms Lars Niemann

University of Freiburg

Operations on neural networks

Lemma (Operations)

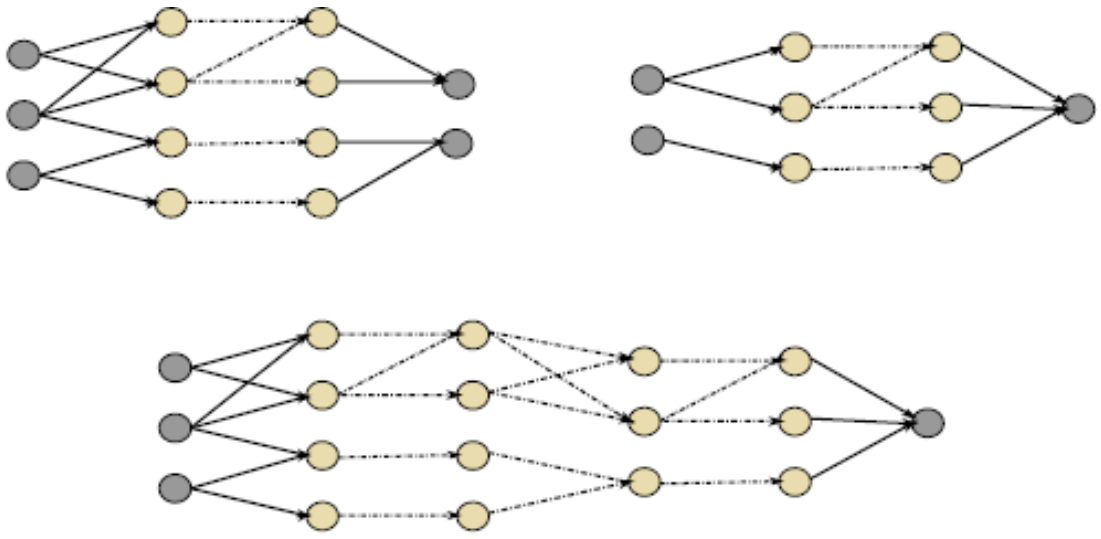
Let Φ^1 and Φ^2 be two neural networks, and let Δ denote the diagonal map $x \mapsto (x, x)$.

- *If the composition $\mathrm{R}(\Phi^1) \circ \mathrm{R}(\Phi^2)$ is well-defined, it can be represented as the realization of a neural network $\Phi^1 \bullet \Phi^2$.*
- *The full parallelization $\big(\mathrm{R}(\Phi^1), \mathrm{R}(\Phi^2)\big)$ can be represented as the realization of a neural network $\mathrm{FP}(\Phi^1, \Phi^2)$.*
- *If the parallelization $\big(\mathrm{R}(\Phi^1), \mathrm{R}(\Phi^2)\big) \circ \Delta$ is well-defined, it can be represented as the realization of a neural network $\mathrm{P}(\Phi^1, \Phi^2)$.*
- *The number of nodes satisfy $\mathrm{M}(\mathrm{P}(\Phi^1, \Phi^2)) = \mathrm{M}(\mathrm{FP}(\Phi^1, \Phi^2)) = \mathrm{M}(\Phi^1) + \mathrm{M}(\Phi^2)$.*

Proof. The networks defined next have the desired properties. □

Concatenation: Intuition

Composition of functions corresponds to concatenation of neural networks:



Concatenation [Figure from Petersen]

Concatenation: Definition

Definition (Concatenation)

Let $L_1, L_2 \in \mathbb{N}$ and let

$$\Phi^1 = \big((A_1^1, b_1^1), \ldots, (A_{L_1}^1, b_{L_1}^1)\big)$$

$$\Phi^2 = \big((A_1^2, b_1^2), \ldots, (A_{L_2}^2, b_{L_2}^2)\big)$$

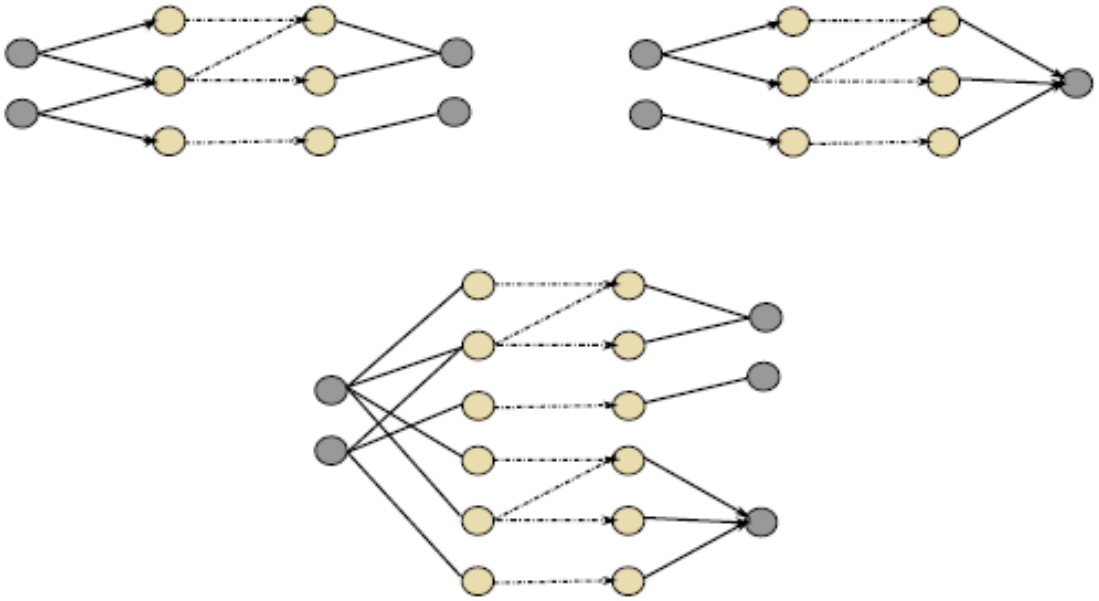
be two neural networks such that the input layer of Φ^1 has the same dimension as the output layer of Φ^2.

Then the concatenation of Φ^1 and Φ^2 is the neural network $\Phi^1 \bullet \Phi^2$ with $L_1 + L_2 - 1$ layers given by

$$\Phi^1 \bullet \Phi^2 := \big((A_1^2, b_1^2), \ldots, (A_{L_2-1}^2, b_{L_2-1}^2), (A_1^1 A_{L_2}^2, A_1^1 b_{L_2}^2 + b_1^1), (A_2^1, b_2^1), \ldots, (A_{L_1}^1, b_{L_1}^1)\big).$$

Parallelisation: Intuition

- The parallelization $\mathrm{P}(\Phi^1, \Phi^2)$ is a neural network with input dimension $d_1 = d_2$, where the inputs are shared.
- The full parallelization $\mathrm{FP}(\Phi^1, \Phi^2)$ is a neural network with input dimension $d_1 + d_2$, where the inputs are not shared.



Parallelisation with shared inputs [Figure from Petersen]

Parallelisation: Definition

Definition

Let Φ^1 and Φ^2 be two neural networks with the same number L of layers and input dimensions d_1 and d_2, respectively:

$$\Phi^1 = \left((A_l^1, b_l^1)\right)_{l\in\{1,\dots,L\}}, \quad \Phi^2 = \left((A_l^2, b_l^2)\right)_{l\in\{1,\dots,L\}}.$$

Then the parallelization and full parallelization of Φ^1 and Φ^2 are the neural networks

$$\mathrm{P}(\Phi^1, \Phi^2) := \Big((\hat{A}_1, \hat{b}_1), (\tilde{A}_2, \tilde{b}_2), \dots, (\tilde{A}_L, \tilde{b}_L)\Big) \quad \text{if } d_1 = d_2,$$

$$\mathrm{FP}(\Phi^1, \Phi^2) := \Big((\tilde{A}_1, \tilde{b}_1), (\tilde{A}_2, \tilde{b}_2), \dots, (\tilde{A}_L, \tilde{b}_L)\Big) \quad \text{for arbitrary } d_1, d_2,$$

where for each $l \in \{1, \dots, L\}$,

$$\hat{A}_l := \begin{pmatrix} A_l^1 \\ A_l^2 \end{pmatrix}, \quad \hat{b}_l := \begin{pmatrix} b_l^1 \\ b_l^2 \end{pmatrix}, \quad \tilde{A}_l := \begin{pmatrix} A_l^1 & 0 \\ 0 & A_l^2 \end{pmatrix}, \quad \tilde{b}_l := \begin{pmatrix} b_1^1 \\ b_1^2 \end{pmatrix}.$$

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Take a pen and paper and verify that the network concatenations and parallelizations satisfy the properties claimed in the lemma.
- Check: Can multiplication of functions be represented as an operation on neural networks?
- Discussion: Can you think of any further operations on neural networks?

Mathematics of Deep Learning, Summer Term 2020

Week 2, Video 6

Universality of Neural Networks

Philipp Harms Lars Niemann

University of Freiburg

Universality

Definition

Let $d, L \in \mathbb{N}$, and let $\rho\colon \mathbb{R} \to \mathbb{R}$ be a continuous activation function. For $K \subseteq \mathbb{R}^d$ compact, denote by $\mathrm{MLP}(\rho, d, L; K)$ the set of multilayer perceptrons with input dimension d, L layers and output dimension 1, restricted to K.

We say that $\mathrm{MLP}(\rho, d, L; K)$ is universal if it is dense in $C(K)$, the space of real-valued continuous functions on K with the supremum norm.

Universal approximation theorem

Definition (Discriminatory activation functions)

Let $d \in \mathbb{N}, K \subseteq \mathbb{R}^d$ compact. A continuous function $\rho : \mathbb{R} \to \mathbb{R}$ is called discriminatory (on K) if the only signed Radon measure μ on K with

$$\int_K \rho(ax - b) d\mu(x) = 0 \quad (a \in \mathbb{R}^d, b \in \mathbb{R})$$

is the zero measure $\mu = 0$.

Theorem (Universal approximation theorem of Cybenko)

Let $d \in \mathbb{N}, K \subseteq \mathbb{R}^d$ compact, and $\rho : \mathbb{R} \to \mathbb{R}$ discriminatory. Then $\mathrm{MLP}(\rho, d, 2; K)$ is universal.

Tool: Riesz–Markov–Kakutani representation

Notation

- Let K be a compact Hausdorff topological space.
- Denote by $C(K)$ the Banach space of real-valued continuous functions on K with the supremum norm.
- Denote by $\mathcal{M}(K)$ the Banach space of finite signed Radon measures on K with the total variation norm.
- Recall that a Borel measure is called Radon if it is regular and locally finite.

Theorem (Riesz–Markov–Kakutani representation)

On any compact Hausdorff topological space K, the topological dual of $C(K)$ is $\mathcal{M}(K)$.

Tool: Hahn–Banach extension

Theorem (Hahn–Banach extension)

If $\mathcal{X}$ is a normed space, M a linear subspace, and λ a continuous linear functional on M, then λ can be extended to a functional $\Lambda\colon \mathcal{X} \to \mathbb{R}$ such that $\|\lambda\| = \|\Lambda\|$.

Consequently, M is dense if and only if every continuous linear functional on $\mathcal{X}$ that vanishes on M is trivial.

Proof of the universal approximation theorem

- Note that $\mathrm{MLP}(\rho, d, 2; K) \subseteq C(K)$ is a linear subspace
- Assume for contradiction that $\mathrm{MLP}(\rho, d, 2; K)$ is not dense
- By Hahn-Banach, there is a non-zero measure μ with

$$\int_K f d\mu = 0 \quad (f \in \mathrm{MLP}(\rho, d, 2; K))$$

- However, the functions $f_{a,b}(x) := \rho(ax - b)$ belong to $\mathrm{MLP}(\rho, d, 2; K)$ for all $a \in \mathbb{R}^d$ and $b \in \mathbb{R}$.
- As ρ is discriminatory, this gives the desired contradiction □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Recount the universal approximation theorem and its proof.
- Check: Verify that one has indeed $K \ni x \mapsto \rho(ax - b) \in \mathrm{MLP}(\rho, d, 2; K)$ for $a \in \mathbb{R}^d, b \in \mathbb{R}$
- Transfer: How does Cybenko's universality theorem differ from the Stone–Weierstrass approximation theorem?

Mathematics of Deep Learning, Summer Term 2020

Week 2, Video 7

Discriminatory Activation Functions

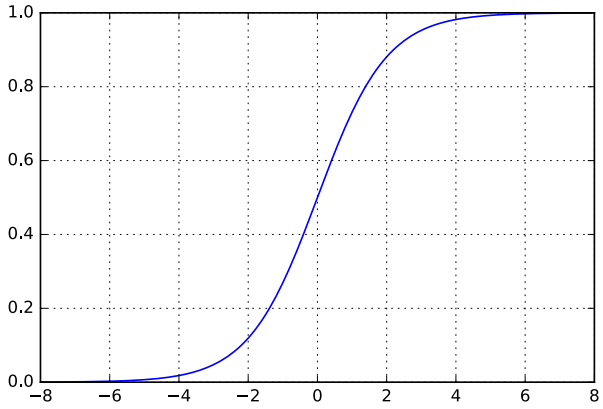
Philipp Harms Lars Niemann

University of Freiburg

Sigmoidal functions

Definition

A continuous function $\rho : \mathbb{R} \to \mathbb{R}$ is called sigmoidal, if $\rho(x) \to 1$ for $x \to \infty$ and $\rho(x) \to 0$ for $x \to -\infty$.



Example: The logistic (aka. sigmoidal) function $x \mapsto (1 + e^{-x})^{-1}$ is sigmoidal

Theorem (Cybenko)

Let $d \in \mathbb{N}, K \subseteq \mathbb{R}^d$ compact. Then every sigmoidal function $\rho\colon \mathbb{R} \to \mathbb{R}$ is discriminatory on K.

Proof that sigmoidal functions are discriminatory

- Let $\mu \in \mathcal{M}(K)$ such that $\int_K \rho(ax-b)d\mu(x) = 0$ for $a \in \mathbb{R}^d, b \in \mathbb{R}$
- For any $\theta \in \mathbb{R}$,

$$\lim_{\lambda \to \infty} \rho(\lambda(ax-b)+\theta) = \begin{cases} 1 & ax-b > 0 \\ \rho(\theta) & ax-b = 0 \\ 0 & ax-b < 0 \end{cases}$$

- Thus, by dominated convergence,

$$\mu(\{ax > b\}) + \rho(\theta)\mu(\{ax = b\}) = \lim_{\lambda \to \infty} \int_K \rho(\lambda(ax-b)+\theta)d\mu(x) = 0$$

- Taking the limit $\theta \to -\infty$, we conclude that

$$\mu(\{ax > b\}) = 0 \quad (a \in \mathbb{R}^d, b \in \mathbb{R})$$

Proof that sigmoidal functions are discriminatory (cont.)

- In particular, for any $b_1 < b_2$,

$$\mu(\{ax > b_1\}) - \mu(\{ax > b_2\}) = \int_K \mathbb{1}_{(b_1,b_2]}(ax)d\mu(x) = 0$$

- This extends first by linearity to step functions and then by density to continuous bounded functions:

$$\int_K g(ax)d\mu(x) = 0 \quad (g \in C_b(\mathbb{R}))$$

- By choosing $g = \sin$ and $g = \cos$, we arrive at

$$0 = \int_K \exp(iax)d\mu(x) \quad (a \in \mathbb{R}^d)$$

- This means the Fourier transform of μ vanishes; whence $\mu = 0$. □

Extensions and variations

- The above proof also works for other dual pairings such as e.g. $L^1(\mathbb{R}^d)$ and $L^\infty(\mathbb{R}^d)$.
- Alternatively, for activation functions $\rho \in \{\sin, \cos, \exp\}$, density of $\{\rho(a \cdot + b); a \in \mathbb{R}^d, b \in \mathbb{R}\}$ in $C(K)$ follows directly from Stone–Weierstrass.
- Alternatively, for activation functions ρ with $\int \rho(x)dx \neq 0$, density in $L^1(K)$ can be shown using the Tauberian theorem of Wiener: any translation-invariant subspace of $L^1(\mathbb{R})$, which contains for any $\xi \in \mathbb{R}$ a function f with $\hat{f}(\xi) \neq 0$, is dense. [Cybenko]

Questions to Answer for Yourself / Discuss with Friends

- Check: Are sigmoidal functions bounded?
- Background: Do you recall the proof of the injectivity of the Fourier transform on measures? (Hint: Stone–Weierstrass for trigonometric polynomials.)

Mathematics of Deep Learning, Summer Term 2020

Week 2, Video 8

Wrapup

Philipp Harms Lars Niemann

University of Freiburg

Outlook on this week's discussion and reading session

- Reading:
  - Hornik (1989): Multilayer Feedforward Networks are Universal Approximators
  - Cybenko (1989): Approximation by superpositions of a sigmoidal function
  - Hornik (1991): Approximation capabilities of multilayer feedforward networks

Summary by learning goals

Having heard this lecture, you can now . . .

- Describe the structure of multi-layer perceptrons and neural networks
- Sketch a brief history of deep learning and put it into the perspective of representation learning.
- State the universal approximation theorem and understand its elegant proof

Mathematics of Deep Learning, Summer Term 2020

Week 3

Dictionary Learning

Philipp Harms Lars Niemann

University of Freiburg

Overview of Week 3

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

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

- 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 $\square$

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

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

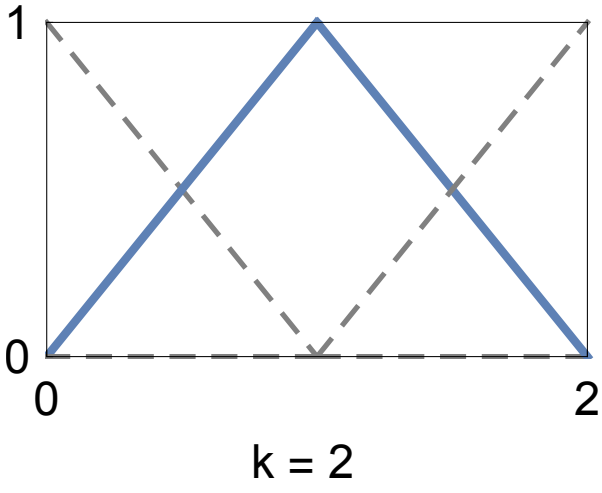
  where $(\cdot)_+ := \max\{0, \cdot\}$.

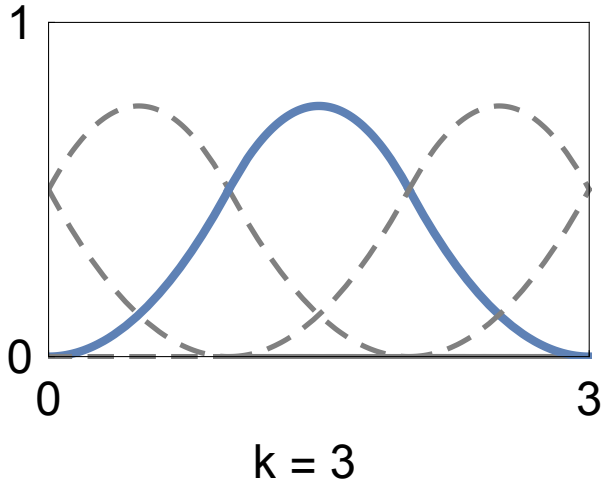
- For $t \in \mathbb{R}$ and $l \in \mathbb{N}$ we define the univariate basis splines by rescalings and translations:

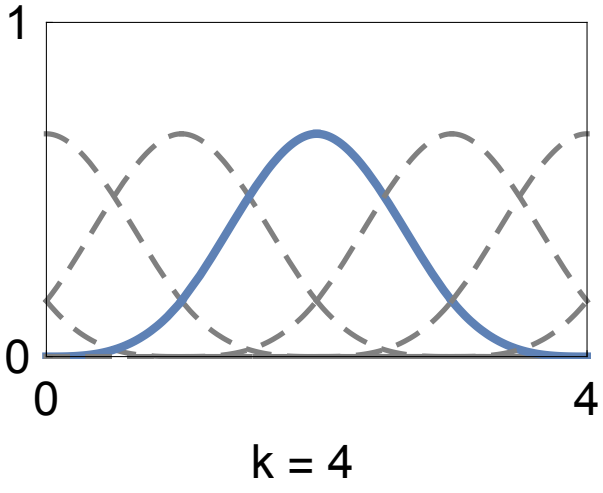
$$\mathcal{N}_{l,t,k}(x) := \mathcal{N}_k(2^l(x-t)) \quad \text{for } x \in \mathbb{R}\,.$$

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 $\mathcal{N}_{l,t,k}^{d}$?
- 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:

- $\frac{\rho(x)}{x^q} \to 0 \quad$ for $x \to -\infty$.
- $\frac{\rho(x)}{x^q} \to 1 \quad$ for $x \to \infty$.
- $|\rho(x)| \lesssim (1 + |x|)^q \quad$ 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 $\left\lceil \max\{\log_q(k), 0\} \right\rceil + 1$ 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

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

Remark:

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

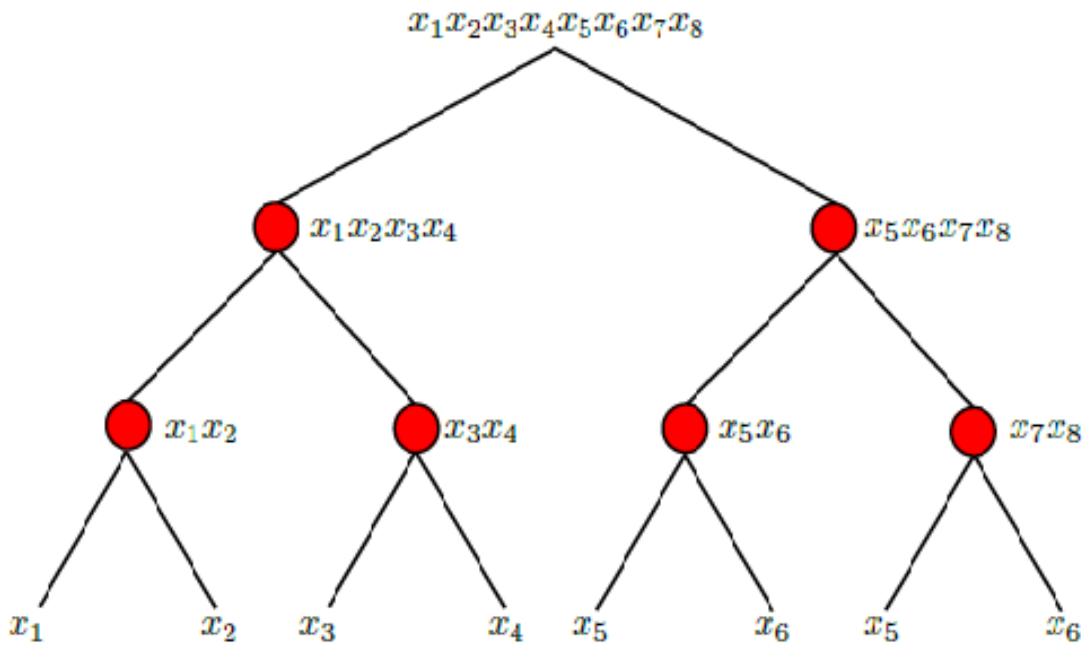
Proof: Representing Products by Square Units

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 $\lceil \log_2(d) \rceil + \lceil \max\{\log_q(k-1), 0\} \rceil + 1$ 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.

Mathematics of Deep Learning, Summer Term 2020

Week 4

Kolmogorov–Arnold Representation

Philipp Harms Lars Niemann

University of Freiburg

Overview of Week 4

1. Hilbert's 13th Problem
2. Kolmogorov–Arnold Representation
3. Approximate Hashing for Specific Functions
4. Approximate Hashing for Generic Functions
5. Proof of the Kolmogorov–Arnold Theorem
6. Approximation by Networks of Bounded Size
7. Wrapup

Acknowledgement of Sources

Sources for this lecture:

- Arnold (1958): On the representation of functions of several variables
- Torbjörn Hedberg: The Kolmogorov Superposition Theorem. In Shapiro (1971): Topics in Approximation Theory
- Philipp Christian Petersen (Faculty of Mathematics, University of Vienna): Course on Neural Network Theory.

Mathematics of Deep Learning, Summer Term 2020

Week 4, Video 1

Hilbert's 13th Problem

Philipp Harms Lars Niemann

University of Freiburg

Hilbert's 13th Problem

Hilbert's 13th problem

Can the roots of the equation

$$x^7 + ax^3 + bx^2 + cx + 1 = 0$$

be represented as superpositions of continuous functions of two variables?

Remark:

- This is the general form of a septic equation after some algebraic transformations. The roots are functions of (a, b, c).
- A single superposition is $w\big(u(a,b), v(b,c)\big)$, and a double superposition is $w\Big(u\big(p(a,b), q(b,c)\big), v\big(r(b,c), s(c,a)\big)\Big)$.
- More generally, the question becomes: Do functions of three variables exist at all, or can they be represented as superpositions of functions of less than three variables?

Hilbert's Conjecture

Conjecture: Hilbert conjectured that such reductions to smaller numbers of variables are impossible. The strongest supporting evidence is:

Theorem (Vitushkin 1955)

There is a polynomial such that neither the polynomial itself nor any function sufficiently close to it (in the sense of uniform convergence) can be decomposed into a simple superposition of continuous functions of two variables in any region or in any system of coordinates.

Dimension theory

Remark: Kolmogorov interpreted Hilbert's problem using dimension theory:

- Let $N(\epsilon)$ be the smallest number of ϵ-balls needed to cover a metric space X.
- On $X = [0,1]^n$ one has $\dim(X) := \liminf_{\epsilon \to 0} \frac{-\log N(\epsilon)}{\log \epsilon} = n$.
- On $X = C^s([0,1]^n)$ one has $\dim(X) := \liminf_{\epsilon \to 0} \frac{-\log \log N(\epsilon)}{\log \epsilon} = n/s$.
- In this sense, Hölder functions of 3 variables are strictly richer than Hölder functions of 2 variables.
- However, as we will see, this argument does not generalize to continuous functions.

Reduction to three variables

Theorem (Kolmogorov 1956)

Any continuous function f of $n \in \mathbb{N}$ variables can be represented as a finite number of superpositions of functions of 3 variables. For instance, for $n = 4$ one has

$$f(x_1, x_2, x_3, x_4) = \sum_{i=1}^{4} g^i\big(u(x_1, x_2, x_3), v(x_1, x_2, x_3), x_4\big)$$

for some continuous functions $g^i, u, v \colon \mathbb{R}^3 \to \mathbb{R}$.

Sketch of Proof: Reduction to three variables

Sketch of Proof:

- The level sets (aka. contour lines) of a continuous function form a tree (Kronrod, Menger):

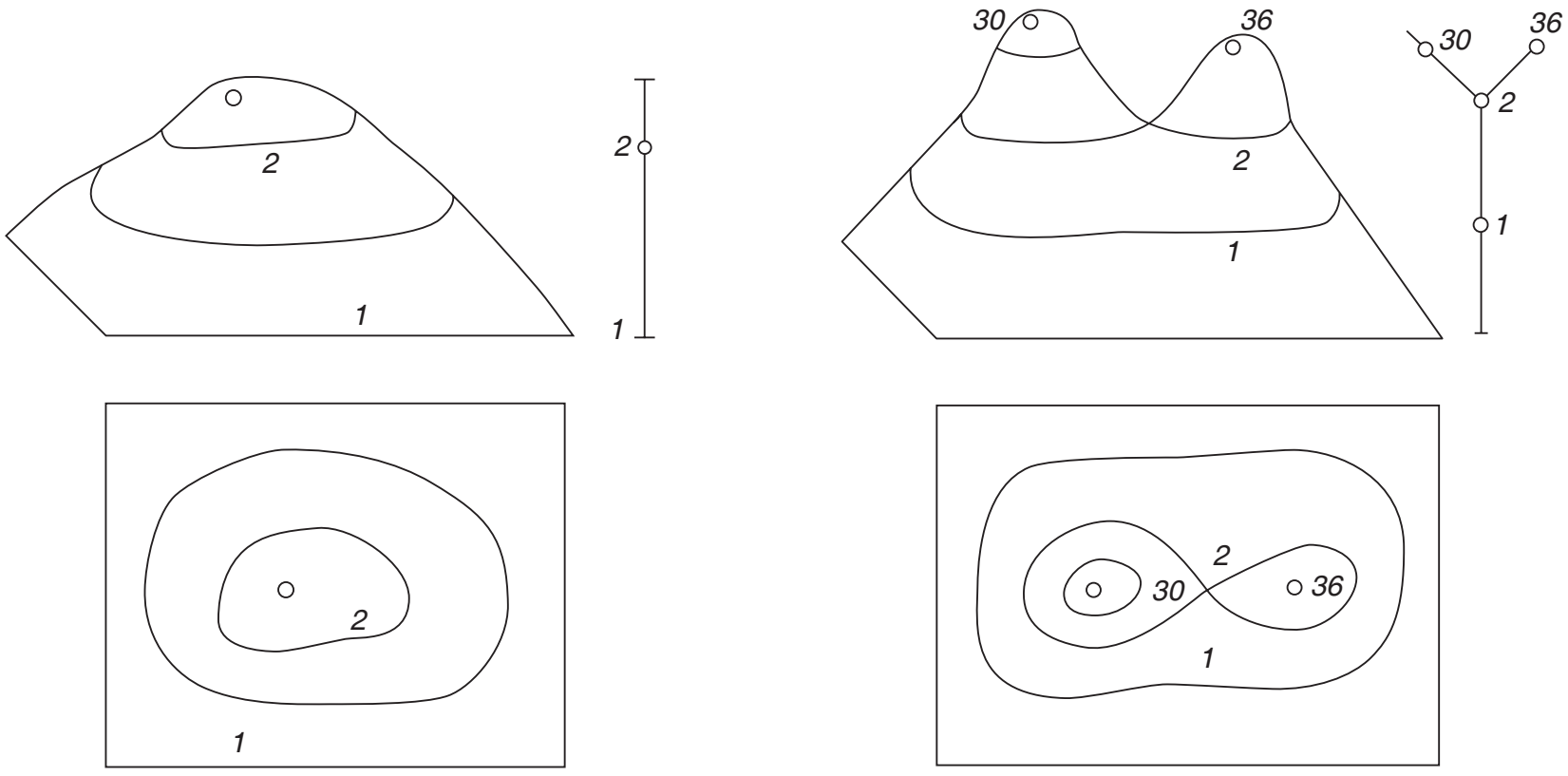


Figure: Figure from Arnold (1956)

Sketch of Proof: Reduction to three variables (cont.)

- Any continuous function of n variables can be written as a sum of $n+1$ continuous functions with standard trees, i.e., trees which do not depend on the given function (Kolmogorov):

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n+1} f^i(x_1, \ldots, x_n).$$

- Each of function f_i can be written as a one-parameter family of functions of $n-1$ variables:

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n+1} f^i_{x_n}(x_1, \ldots, x_{n-1})$$

Sketch of Proof: Reduction to three variables (cont.)

- Each of the functions $f^i_{x_n}$ factors through a function on the corresponding standard tree:

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n+1} g^i_{x_n}(\ell^i(x_1, \ldots, x_{n-1})).$$

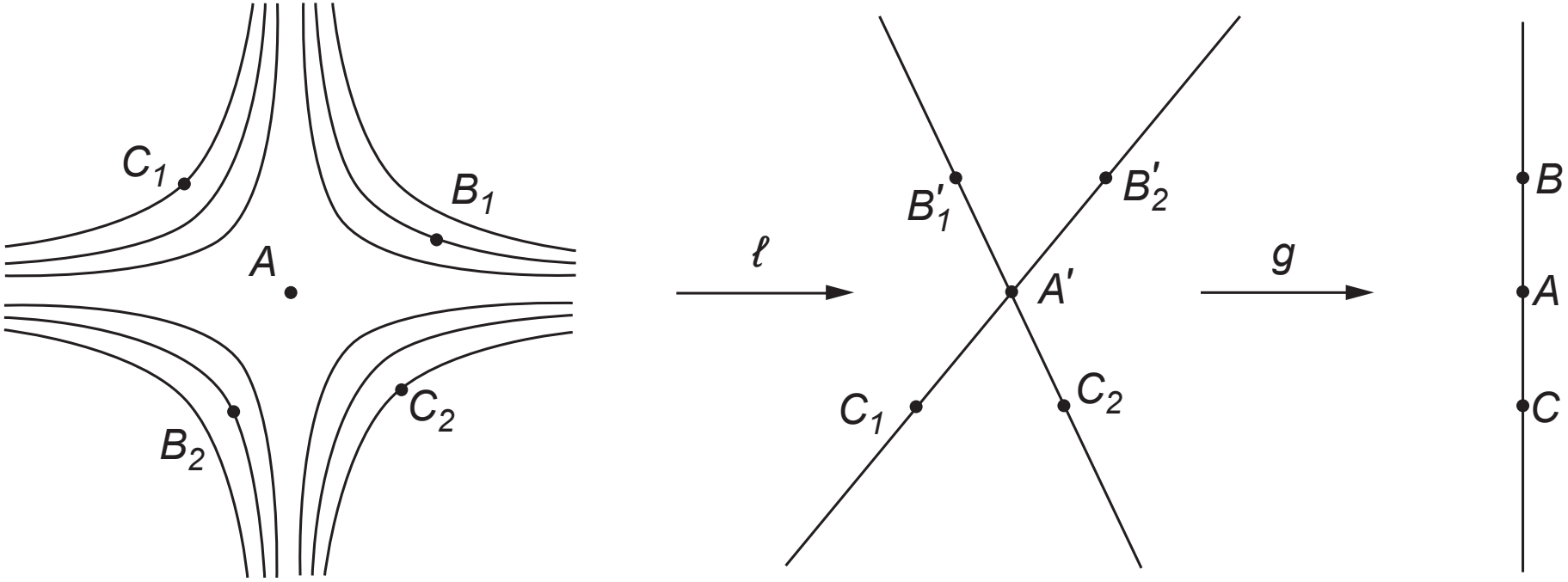


Figure: Figure from Arnold (1956)

Sketch of Proof: Reduction to three variables (cont.)

- Embedding the trees in a plane with a two-dimensional coordinate system (u, v) transforms this into:

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n+1} g^i_{x_n}\big(u^i(x_1, \ldots, x_{n-1}), v^i(x_1, \ldots, x_{n-1})\big).$$

- This yields 3-variate functions g_i and $(n-1)$-variate functions u^i, v^i:

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n+1} g^i\big(u^i(x_1, \ldots, x_{n-1}), v^i(x_1, \ldots, x_{n-1}), x_n\big).$$

- Applying this construction iteratively to u^i and v^i yields the reduction to superpositions of functions of 3 variables. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: State Hilbert's 13th problem and describe how Kolmogorov cast it in the frameworks of dimension and graph theory.
- Check: What happens to Hilbert's problem when continuous functions are replaced by measurable or arbitrary functions?
- Background: Find out about generalizations, limitations, and open problems related to Hilbert's thirteenth problem.

Mathematics of Deep Learning, Summer Term 2020

Week 4, Video 2

Kolmogorov–Arnold Representation

Philipp Harms Lars Niemann

University of Freiburg

Kolmogorov–Arnold Representation

Theorem (Kolmogorov–Arnold 1956–1957)

For every $n \in \mathbb{N}_{\geq 2}$, there exist $\varphi_{i,j} \in C([0,1])$ such that any $f \in C([0,1]^n)$ can be represented as

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{2n+1} g_i \left(\sum_{j=1}^{n} \varphi_{i,j}(x_j) \right),$$

for some $g_i \in C(\mathbb{R})$.

Remark:

- This disproves Hilbert's conjecture and shows that "the only" multivariate function is a sum.
- The inner functions $\varphi_{i,j}$ are universal, i.e., they do not depend on f.
- The outer functions g_i can be learned by linear regression.

Sprecher's Refinement: Universal Inner Function

Theorem (Sprecher 1965, Köppen 2002)

For every $n \in \mathbb{N}_{\geq 2}$, there exists a continuous function $\varphi : \mathbb{R} \to \mathbb{R}$ and constants $a, \lambda_j \in \mathbb{R}$ such that any $f \in C([0,1]^n)$ can be represented as

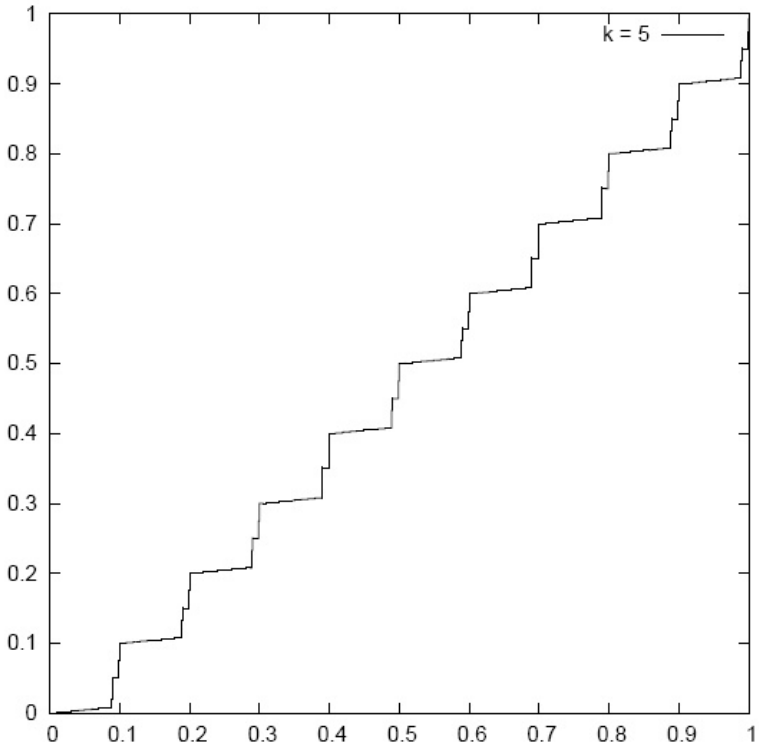
$$f(x_1, \ldots, x_n) = \sum_{i=1}^{2n+1} g_i \left(\sum_{j=1}^{n} \lambda_j \varphi(x_j + ia) \right),$$

for some $g_i \in C(\mathbb{R})$.

Remark:

- The function φ and the constants λ_j and a can be constructed explicitly and are universal, i.e., independent of f.
- Sprecher's representation can be interpreted as a neural network.
- There are many further versions of the Kolmogorov–Arnold theorem with varying regularity and structural assumptions.

Sprecher's Refinement: Universal Inner Function



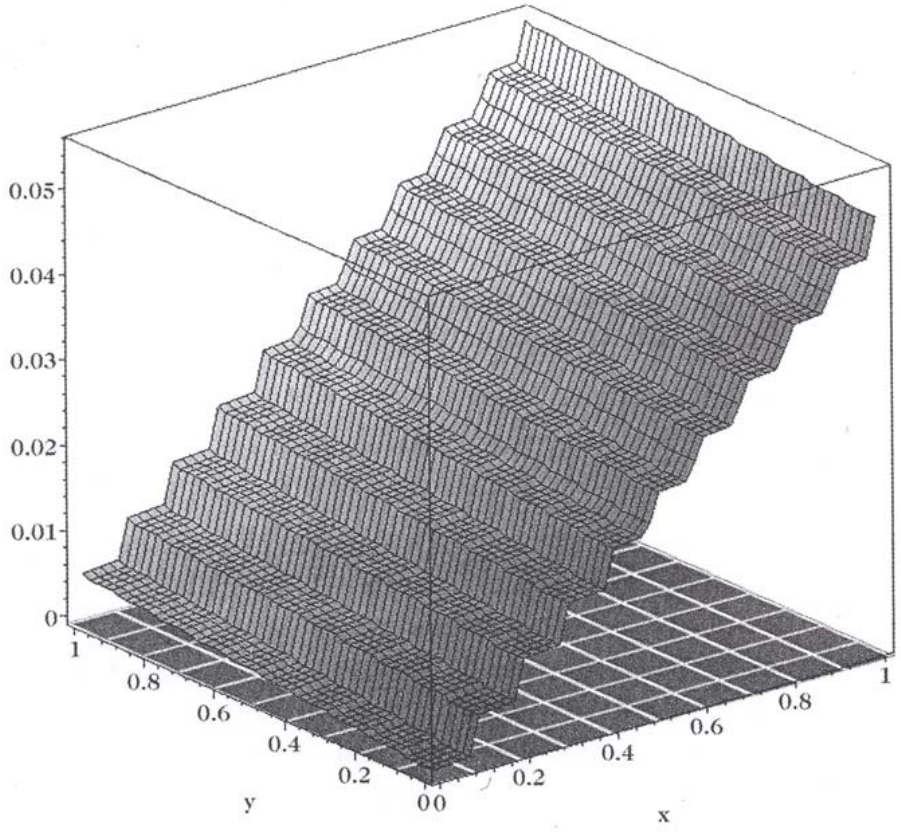


Figure: Sprecher's universal inner functions φ (left) and ψ_1 (right), where $\psi_i(x_1, x_2) := \lambda_1\varphi(x_1 + ia) + \lambda_2\varphi(x_2 + ia)$ for some constants λ_1, λ_2, a. [Leni Fougerolle Truchetet 2008]

Hashing

Remark:

- The inner functions in the Kolmogorov–Arnold representation theorem can be interpreted as hash functions.

Background:

- Hash functions are widely used in computer science for array indexing operations.
- They map high-dimensional/unstructured/variable-length data to scalar hash values.
- Hash functions should be fast to compute and should be "nearly" injective, i.e., minimize duplication of output values.

Hashing and Kolmogorov–Arnold Representation

Lemma

For each $i \in \{1, \ldots, 2n+1\}$, Sprecher's inner function

$$\psi_i \colon [0,1]^n \ni (x_1, \ldots, x_n) \mapsto \sum_{j=1}^{n} \lambda_j \varphi(x_j + ia) \in \mathbb{R}$$

is injective on a countable dense subset $D \subseteq [0,1]^n$.

Remark:

- It is sufficient to establish injectivity of $\psi(x) := \sum_j \lambda_j \varphi(x_j)$ on D.
- This follows from the following two facts: ϕ takes rational values on D, and the coefficients λ_j are independent over the rational numbers.
- Of course, ψ is not injective everywhere; otherwise the Kolmogorov–Arnold theorem would be trivial.

Space-filling curves

- Intuitively, the inverse of a hash function $[0,1]^n \to [0,1]$ is a space-filling curve, i.e., a surjective continuous map $[0,1] \to [0,1]^n$.
- For Sprecher's hash function, this is made precise as follows: By carefully examining the properties of ψ, one may construct an "inverse" map $\lambda : [0,1] \to [0,1]^n$ with the following properties:

Lemma

1. *The map $\lambda : [0,1] \to [0,1]^n$ is a space-filling curve.*
2. *Its image may be approximated by discrete curves Λ_k as $k \to \infty$.*

Remark:

- By the Hahn–Mazurkiewicz theorem, a non-empty Hausdorff topological space is a continuous image of the unit interval if and only if it is compact, connected, locally connected, and second-countable.

Space-filling curves

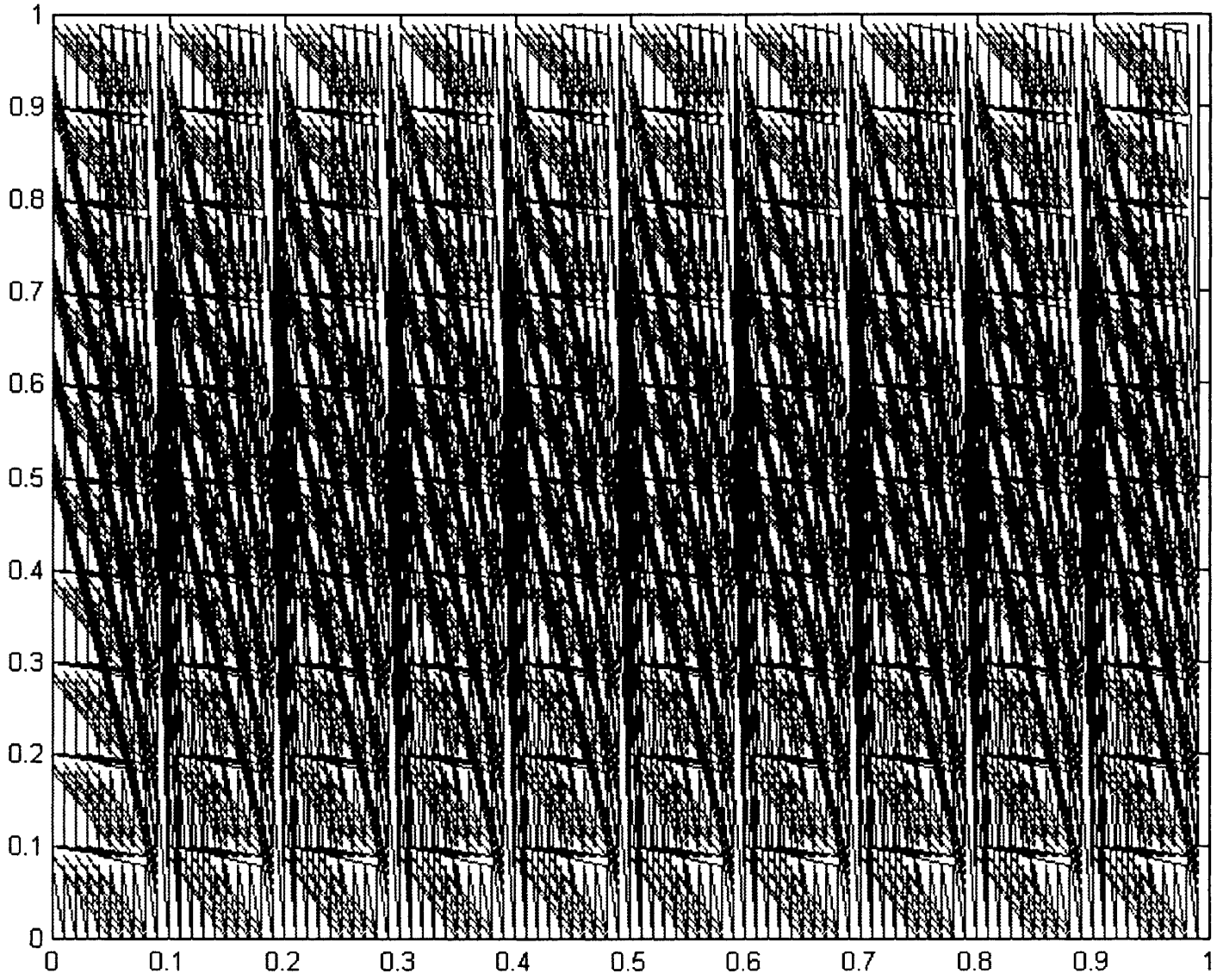


Figure: An approximation Λ_k of the space-filling curve λ. [Sprecher Draghici 2002]

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Recall and compare the presented versions of the Kolmogorov–Arnold Theorem.
- Check: Why exactly does the Kolmogorov–Arnold representation theorem disprove Hilbert's conjecture?
- Check: Show that there is no continuous bijection $[0,1]^n \to [0,1]$ for any $n \geq 2$.
- Discussion: How would you implement Sprecher's theorem using neural networks? Do you think this could work well for supervised learning?

Mathematics of Deep Learning, Summer Term 2020

Week 4, Video 3

Approximate Hashing for Specific Functions

Philipp Harms Lars Niemann

University of Freiburg

Hashing rational numbers

Lemma

There exists a linear map $\ell\colon \mathbb{R}^n \to \mathbb{R}$ whose restriction to rational numbers is injective.

Proof:

- $n = 2$: Set $\ell(x, y) = x + \lambda y$ for any irrational number λ.
- $n \geq 2$: Set $\ell(x_1, \ldots, x_n) := \lambda_1 x_1 + \cdots + \lambda_n x_n$, where λ_i are independent over $\mathbb{Q}$, e.g. $\lambda_i = \pi^{i-1}$ or some other powers of any transcendental number. □

Remark:

- Thus, any $f\colon \mathbb{Q}^n \to \mathbb{R}$ can be written as $f = g \circ \ell$, where ℓ is the above linear hashing function. However, g cannot be chosen continuously, and the approximation error cannot be controlled on non-rational numbers—a more elaborate construction is needed.
- We fix an irrational number $\lambda \in \mathbb{R} \setminus \mathbb{Q}$ throughout this section.

Approximate Hashing for a Specific Function

Remark:

- The key step in the proof of the Kolmogorov–Arnold theorem is the construction of approximate hashing functions.
- This is done here for a given specific function and in the next section for generic functions.
- We restrict ourselves to bivariate functions.

Definition (Approximate hashing functions, specific f)

A function $\varphi \in C([0,1], \mathbb{R}^5)$ is called approximate hashing function for $f \in C([0,1]^2)$ if there exists $g \in C(\mathbb{R})$ such that

$$\sup_{t \in \mathbb{R}} |g(t)| \leq 1/7, \qquad \sup_{x,y \in [0,1]} \left| f(x,y) - \sum_{i=1}^{5} g\big(\varphi_i(x) + \lambda \varphi_i(y)\big) \right| < 7/8.$$

Approximate Hashing for a Specific Function

Lemma

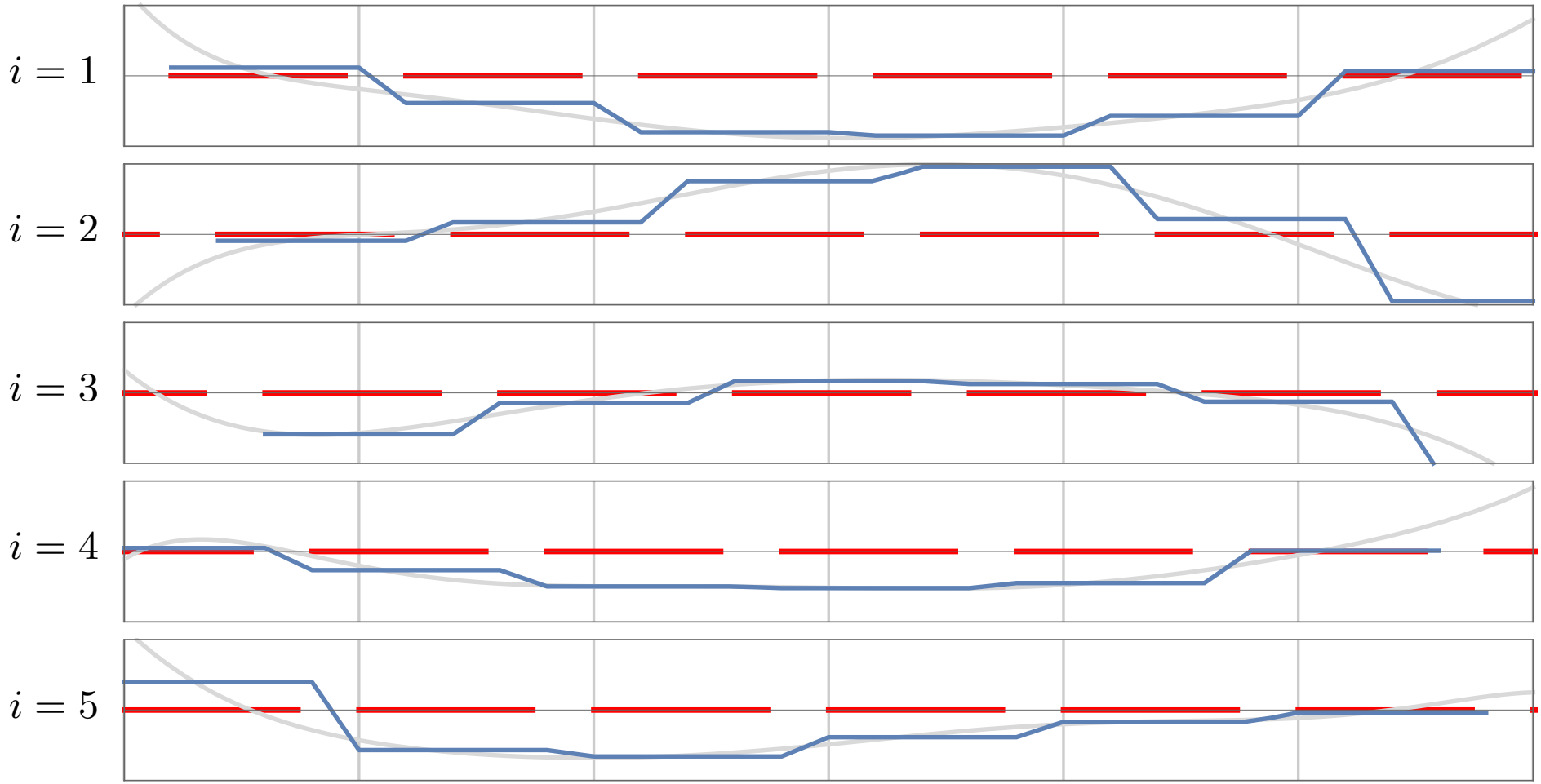
For any $f \in C^2([0,1]^2)$ with $\|f\|_\infty \leq 1$, the set of approximate hashing functions for f is open and dense in $C([0,1], \mathbb{R}^5)$.

Proof:

- The set is open, since if g works for a particular φ, it does so for every nearby φ.
- It remains to show that the set is dense in $C([0,1], \mathbb{R}^5)$.
- Thus, given $\epsilon > 0$ and $\chi \in C([0,1], \mathbb{R}^5)$, we have to find an approximate hashing function φ for f such that $\|\varphi - \chi\| \leq \epsilon$.

Proof: Approximate Hashing for a Specific Function

- Divide $[0,1]$ into $N \in \mathbb{N}$ intervals, cut out the i-th fifth of each interval, and color all remaining intervals red.
- Approximate χ_i (gray) by functions φ_i (blue), which are constant on red intervals of type i.



Proof: Approximate Hashing for a Specific Function

- It can be arranged that each function φ_i assumes distinct rational numbers on each of the red intervals, and that these numbers are distinct for different i.
- Moreover, for sufficiently large N, $\|\varphi - \chi\| \leq \epsilon$, as desired.
- Furthermore, by the uniform continuity of f on $[0,1]^2$, we can make N even larger to get

$$\left|f(x,y) - f(x',y')\right| \leq 1/7 \text{ whenever } \max\{\left|x - x'\right|, \left|y - y'\right|\} \leq 4/N.$$

Proof: Approximate Hashing for a Specific Function

- The function $\psi_i(x, y) := \varphi_i(x) + \lambda\varphi_i(y)$ is constant on red rectangles of type i, which are defined as products of red intervals of type i.
- The irrational numbers, which the functions ψ_i assume on rectangles of type i, are all distinct for different rectangles and/or different i.
- Thus, there is $g \in C(\mathbb{R})$ such that $g(\psi_i(x, y)) = \pm 1/7$ if (x, y) belongs to a red rectangle of type i where $f \gtrless 0$.
- Without loss of generality, $\|g\| \leq 1/7$.
- Intuitively, g tracks the sign of f on each rectangle.

Proof: Approximate Hashing for a Specific Function

- For any point (x, y), consider the approximation error

$$\left| f(x,y) - \sum_{i=1}^{5} g\big(\psi_i(x,y)\big) \right|. \qquad (*)$$

- If $f(x,y) \gtrless 1/7$, then $f \gtrless 0$ on each red rectangle containing (x,y).
- There are at least 3 such rectangles because out of 5 types, one may fail on the x-axis and another one on the y-axis.
- Thus, the majority of the summands in $(*)$ tracks the sign of f correctly, and the approximation error is bounded by $6/7$.
- If $|f(x,y)| \leq 1/7$, the approximation error is again bounded by $6/7$, regardless of correct or incorrect tracking.
- As $6/7 < 7/8$, we have shown that φ is an approximate hashing function, which is ϵ-close to χ. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Recall the definition of and main result on approximate hashing.
- Background: Refresh your memory of algebraic closures and the definition of algebraic and transcendental numbers, if necessary.
- Check: Draw the red rectangles of types 1 to 5 and verify that each point is contained in at least three rectangles.
- Check: What is the role of the numbers 5 and $1/7$ in the lemma? Can they be altered?

Mathematics of Deep Learning, Summer Term 2020

Week 4, Video 4

Approximate Hashing for Generic Functions

Philipp Harms Lars Niemann

University of Freiburg

Approximate Hashing for Generic Functions

Remark:

- As before, we fix an irrational number $\lambda \in \mathbb{R} \setminus \mathbb{Q}$.

Definition (Approximate hashing functions)

A function $\varphi \in C([0,1], \mathbb{R}^5)$ is called approximate hashing function if for any $f \in C([0,1]^2)$, there exists $g \in C(\mathbb{R})$ such that

$$\|g\|_\infty \leq \frac{1}{7} \|f\|_\infty , \qquad \left\| f - \sum_{i=1}^{5} g \circ \psi_i \right\|_\infty \leq \frac{8}{9} \|f\|_\infty ,$$

where $\psi_i(x,y) = \varphi_i(x) + \lambda \varphi_i(y)$.

Remark:

- Compared to hashing for specific functions f, this definition imposes the hashing property simultaneously for all f and with a slightly worse error bound.

Approximate Hashing for Generic Functions

Lemma

The set of approximate hashing functions is dense in $C([0,1],\mathbb{R}^5)$.

Proof:

- Let U_k be the sets of approximate hashing functions of f_k, for some dense sequence $(f_k)_{k\in\mathbb{N}}$ in the unit sphere of $C([0,1]^2)$.
- The sets U_k are open and dense. By Baire's category theorem, its intersection U is dense.
- Any function $\varphi \in U$ is an approximate hashing function: for any f with $\|f\|_\infty \leq 1$, there exists f_k and g such that

$$\begin{aligned}\left\|f - \sum_i g\circ\psi_i\right\|_\infty &\leq \|f - f_k\|_\infty + \left\|f_k - \sum_i g\circ\psi_i\right\|_\infty \\ &\leq \left(\tfrac{8}{9} - \tfrac{7}{8}\right) + \tfrac{7}{8} = \tfrac{8}{9}.\end{aligned}$$

- Extend to general f by scaling. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What is the difference between hashing for specific versus generic functions, and how does the former imply the latter?
- Background: Refresh your memory of the Baire category theorem if necessary.
- Discussion: Can you strengthen the proof to get monotonically increasing approximate hashing functions?

Mathematics of Deep Learning, Summer Term 2020

Week 4, Video 5

Proof of the Kolmogorov–Arnold Theorem

Philipp Harms Lars Niemann

University of Freiburg

Kolmogorov–Arnold Representation, Refined Version

Remark: The approximate hashing results imply the following refined version of the Kolmogorov–Arnold representation theorem:

Theorem (Kolmogorov–Arnold representation, refined version)

For any $n \in \mathbb{N}_{\geq 2}$, there exist $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$ and $\varphi_1, \ldots, \varphi_{2n+1} \in C([0,1])$ such that any $f \in C([0,1]^n)$ admits a representation

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{2n+1} g\big(\lambda_1 \varphi_i(x_1) + \cdots + \lambda_n \varphi_i(x_n)\big)$$

for some continuous function g.

Remark: The difference to Kolmogorov's original result is that g does not depend on i.

Proof: Kolmogorov–Arnold Representation for $n = 2$

Proof: Iterative improvement of the approximate hashing representation

- Let $\varphi \in C([0,1], \mathbb{R}^5)$ be an approximate hashing function, define $\psi_i(x,y) = \lambda_1 \varphi_i(x) + \lambda_2 \varphi_i(y)$ for $\lambda_1 := 1$ and λ_2 irrational, and define $Tg := \sum_{i=1}^{5} g \circ \psi_i$.
- Set $f_1 := f$ and find g_1 with $\|g_1\|_\infty \leq \frac{1}{7}\|f_1\|_\infty$ and $\|f_1 - Tg_1\|_\infty \leq \frac{7}{8}\|f_1\|_\infty$.
- Set $f_2 := f_1 - Tg_1$ and find g_2 with $\|g_2\|_\infty \leq \frac{1}{7}\|f_2\|_\infty$ and $\|f_2 - Tg_2\|_\infty \leq \frac{7}{8}\|f_2\|_\infty$.
- Continue to eternity. When done, set $g = \sum_k g_k$ and note that $f = Tg$ as required. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Recall the proof of the Kolmogorov–Arnold theorem via the construction of approximate hashing functions.
- Discussion: How does the proof work in higher dimensions?

Mathematics of Deep Learning, Summer Term 2020

Week 4, Video 6

Approximation by Networks of Bounded Size

Philipp Harms Lars Niemann

University of Freiburg

Approximation by Networks of Bounded Size

Theorem

There exists a continuous, piece-wise polynomial activation function $\rho\colon \mathbb{R} \to \mathbb{R}$ *which allows one to approximate continuous multivariate functions by realizations of neural networks with bounded size, that is, for all* $n \in \mathbb{N}$ *there exists a constant* $C = C(n)$ *such that*

$$\forall \epsilon > 0 \; \forall f \in C([0,1]^n) \; \exists \Phi : \quad \mathrm{L}(\Phi) = 3, \, \mathrm{M}(\Phi) \leq C(n), \, \|f - \mathrm{R}(\Phi)\|_\infty \leq \epsilon .$$

Remark:

- This theorem is in a sense "too good" because it provides an approximate representation of continuous functions by finitely many real numbers.
- It highlights the influence of the choice of activation function on the resulting approximation theory.
- It also points to the importance of asking for bounded weights.

Approximation by Networks of Bounded Size

Lemma (Univariate case)

The theorem holds in the univariate case $n = 1$: there exists a continuous, piecewise polynomial activation function $\rho\colon \mathbb{R} \to \mathbb{R}$ such that

$$\forall \epsilon > 0 \; \forall f \in C([0,1]) \; \exists \Phi : \quad \mathrm{L}(\Phi) = 2, \quad \mathrm{M}(\Phi) \leq 3, \quad \|f - \mathrm{R}(\Phi)\|_\infty \leq \epsilon \,.$$

Remark: By translation and scaling, this extends to continuous functions f on every closed interval $[a, b] \subseteq \mathbb{R}$.

Proof: Approximation by Networks of Bounded Size

Proof of the lemma:

- Recall that the set Π of polynomials with rational coefficients is dense in the Polish space $C([0,1])$, and let $(\pi_i)_{i\in\mathbb{Z}}$ be an enumeration of Π.
- Define the activation function ρ by

$$\rho(x) := \begin{cases} \pi_i(x-2i), & x \in [2i, 2i+1] \\ \pi_i(1)(2i+2-x) + \pi_{i+1}(0)(x-2i-1), & x \in (2i+1, 2i+2) \end{cases}$$

- Note that, by the very definition of ρ, one has $\rho(x+2i) = \pi_i(x)$ for $x \in [0,1]$.
- Hence, the neural network $\Phi := ((1, 2i), (1, 0))$ has the desired properties. □

Proof: Approximation by Networks of Bounded Size

Proof of the theorem:

- By the Kolmogorov–Arnold theorem (refined version),

$$f = \sum_{i=1}^{2n+1} g \circ \psi_i, \quad \psi_i(x_1, \ldots, x_n) = \lambda_1 \varphi_i(x_1) + \cdots + \lambda_n \varphi_i(x_n).$$

  for some $g \in C(\mathbb{R})$, $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$ and $\varphi_1, \ldots, \varphi_{2n+1} \in C([0,1])$.
- By the previous lemma, $\varphi_i \approx \mathrm{R}(\Phi_i) \in C([0,1])$ for some networks Φ_i and a piece-wise polynomial activation function ρ, where $\approx$ denotes approximation up to arbitrary accuracy.
- Then $\psi_i \approx \mathrm{R}(\Psi_i) \in C([0,1]^n)$ for each $i \in \{1, \ldots, 2n+1\}$, where

$$\Psi_i = (((\lambda_1, \ldots, \lambda_n), 0)) \bullet \mathrm{FP}(\Phi_i, \ldots, \Phi_i).$$

Proof: Approximation of Multivariate Functions (cont.)

- By the previous lemma, $g \approx \mathrm{R}(\Xi) \in C([-K, K])$, where K is sufficiently large such that $\psi_i([0,1]^n) \subseteq [-K, K]$.
- Then the network

$$\Phi := (((1, \ldots, 1), 0)) \bullet \mathrm{FP}(\Xi, \ldots, \Xi) \bullet \mathrm{P}(\Psi_1, \ldots, \Psi_{2n+1}).$$

  has the desired number of layers and weights.
- Moreover, $f \approx \mathrm{R}(\Phi)$ thanks to the estimate

$$\begin{aligned} \|f - \mathrm{R}(\Phi)\| &\leq \sum_i \|\mathrm{R}(\Xi) \circ \mathrm{R}(\Psi_i) - g \circ \psi_i\| \\ &\leq \sum_i \|\mathrm{R}(\Xi) \circ \mathrm{R}(\Psi_i) - \mathrm{R}(\Xi) \circ \psi_i\| + \|\mathrm{R}(\Xi) \circ \psi_i - g \circ \psi_i\|, \end{aligned}$$

  and thanks to the uniform continuity of $\mathrm{R}(\Xi)$ on $[-K, K]$. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Recall the approximation of univariate and multivariate functions by networks of bounded size.
- Check: Verify that the activation function ρ constructed in the univariate case is continuous.
- Discussion: What are theoretical implications to approximation theory and practical implications to supervised learning?

Mathematics of Deep Learning, Summer Term 2020

Week 4, Video 7

Wrapup

Philipp Harms Lars Niemann

University of Freiburg

Outlook on this week's discussion and reading session

- Reading:
  - Arnold (1958): On the representation of functions of several variables
  - Bar-Natan (2009): Hilberts 13th problem, in full color
  - Hecht-Nielsen (1987): Kolmogorov's mapping neural network existence theorem

Summary by learning goals

Having heard this lecture, . . .

- You can describe the Kolmogorov–Arnold representation theorem and its proof.
- You can appreciate the fundamental distinction between inner and outer network layers.
- You are aware that different choices of activation functions may lead to very different approximation theories.

Mathematics of Deep Learning, Summer Term 2020

Week 5

Harmonic Analysis

Philipp Harms Lars Niemann

University of Freiburg

Overview of Week 5

1. Banach frames
2. Group representations
3. Signal representations
4. Regular Coorbit Spaces
5. Duals of Coorbit Spaces
6. General Coorbit Spaces
7. Discretization
8. Wrapup

Acknowledgement of Sources

Sources for this lecture:

- Christensen (2016): An introduction to frames and Riesz bases
- Dahlke, De Mari, Grohs, Labatte (2015): Harmonic and Applied Analysis

Mathematics of Deep Learning, Summer Term 2020

Week 5, Video 1

Banach frames

Philipp Harms Lars Niemann

University of Freiburg

Bases in Banach spaces

Definition (Schauder 1927)

Let X be a Banach space. A Schauder basis is a sequence $(e_k)_{k\in\mathbb{N}}$ in X with the following property: for every $f \in X$ there exists a unique scalar sequence $(c_k(f))_{k\in\mathbb{N}}$ such that

$$f = \sum_{k=1}^{\infty} c_k(f) e_k.$$

The Schauder basis is called unconditional if this sum converges unconditionally.

Remark:

- Any Banach space with a Schauder basis is necessarily separable.
- Not all separable Banach spaces have a Schauder basis (Enflo 1972).
- The coefficient functionals c_k are continuous, i.e., belong to X^*.

Translations, Modulations, and Scalings

Remark: Many useful bases are constructed by translations, modulations, and scalings of a given "mother wavelet."

Lemma

The following are unitary operators on $L^2(\mathbb{R})$, which depend strongly continuously on their parameters $a, b \in \mathbb{R}$ and $c \in \mathbb{R} \setminus \{0\}$:

- *Translation:* $T_a f(x) := f(x - a)$.
- *Modulation:* $E_b f(x) := e^{2\pi i b x} f(x)$.
- *Scaling (aka. dilation):* $D_c f(x) := c^{-1/2} f(xc^{-1})$.

Remark:

- These are actually group representations; more on this later.

Examples of Bases

Example: Fourier series

- The functions $(E_k 1)_{k\in\mathbb{Z}}$ are an orthonormal basis in $L^2([0,1])$.

Example: Gabor bases

- The functions $(E_k T_n \mathbb{1}_{[0,1]})_{k,n\in\mathbb{Z}}$ are an orthonormal basis in $L^2(\mathbb{R})$.

Example: Haar bases

- The functions $(D_{2^j} T_k \psi)_{j,k\in\mathbb{Z}}$ are an orthonormal basis of $L^2(\mathbb{R})$.
- Here ψ is the Haar wavelet

$$\psi(x) = \begin{cases} 1, & 0 \le x < \frac{1}{2}, \\ -1, & \frac{1}{2} \le x < 1, \\ 0, & \text{otherwise.} \end{cases}$$

Example: Wavelet bases

- Replace ψ by functions with better smoothness or support properties

Limitations of Bases

Requirements: continuous operations for

- Analysis: encoding f into basis coefficients (c_k)
- Synthesis: decoding f from basis coefficients (c_k)
- Reconstruction: writing $f = \sum_k c_k e_k$.

Limitations:

- It is often impossible to construct bases with special properties
- Even a slight modification of a Schauder basis might destroy the basis property

Idea: use "over-complete" bases, aka. frames

- Drop linear independence of (e_k) and uniqueness of (c_k)
- Require continuity of the analysis and synthesis operators
- Get additional benefits such as noise suppression and localization in time and frequency

Banach Frames

Definition (Gröchenig 1991)

Let X be a Banach space, and let Y be a Banach space of sequences indexed by $\mathbb{N}$. A Banach frame for X with respect to Y is given by

- Analysis: A bounded linear operator $A\colon X \to Y$, and
- Synthesis: A bounded linear operator $S\colon Y \to X$, such that
- Reconstruction: $S \circ A = \mathrm{Id}_X$.

Remark:

- The k-th frame coefficient is $c_k := \mathrm{ev}_k \circ A \in X^*$.
- If the unit vectors $(\delta_k)_{k\in\mathbb{N}}$ are a Schauder basis in Y, one obtains an atomic decomposition into frames $e_k := S\delta_k \in X$ as follows:

$$\forall f \in X: \qquad f = \sum_{k\in\mathbb{N}} c_k(f) e_k.$$

- Every separable Banach space has a Banach frame.

Examples of Banach frames

Example: Hilbert frames

- A Banach frame on a Hilbert space H with respect to ℓ^2 is a sequence $(e_k)_{k\in\mathbb{N}}$ s.t. for all $f \in H$,

$$\|f\|_H^2 \lesssim \sum_{k\in\mathbb{N}} |\langle f, e_k\rangle_H|^2 \lesssim \|f\|_H^2.$$

Example: Projections

- The projection of a Schauder basis to a subspace is a Banach frame.
- E.g., the functions $(E_k 1)_{k\in\mathbb{Z}}$ are a frame but not a basis in $L^2(I)$ for any $I \subsetneq [0,1]$.

Example: Wavelet frames

- If $\psi \in L^2(\mathbb{R}) \cap C^\infty(\mathbb{R})$ is required to have exponential decay and bounded derivatives, then $(D_{2^j} T_k \psi)_{j,k\in\mathbb{Z}}$ cannot be a basis but can be a frame.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What are Schauder bases versus frames?
- Repetition: Give some examples of frames constructed via translations, scalings, and modulations.
- Check: Is a Schauder basis a basis?
- Check: Verify the strong continuity of the translation, scaling, and modulation group actions.

Mathematics of Deep Learning, Summer Term 2020

Week 5, Video 2

Group representations

Philipp Harms Lars Niemann

University of Freiburg

Locally compact groups

Definition (Locally compact group)

A locally compact group is a group endowed with a Hausdorff topology such that the group operations are continuous and every point has a compact neighborhood.

Theorem (Haar 1933)

Every locally compact group has a left Haar measure, i.e., a non-zero Radon measure which is invariant under left-multiplication. This measure is unique up to a constant. Similarly for right Haar measures.

Definition (Unimodular groups)

A group is unimodular if its left Haar measure is right-invariant.

Convolutions

Lemma (Young inequality)

For any $p \in [1, \infty]$, $f \in L^1(G)$, and $g \in L^p(G)$, the convolution

$$f * g(x) := \int_G f(y) g(y^{-1}x) dy = \int_G f(xy) g(y^{-1}) dy$$

is well-defined, belongs to L^p, and $\|f * g\|_{L^p(G)} \leq \|f\|_{L^1(G)} \|g\|_{L^p(G)}$.

Proof: This follows from Minkowski's integral inequality,

$$\left\| \int_G f(y) g(y^{-1} \cdot) dy \right\|_{L^p(G)} \leq \int_G |f(y)| \ \|g(y^{-1} \cdot)\|_{L^p(G)} dy,$$

and from the invariance of the L^p norm. □

Remark: The same conclusion holds for $g * f$ if G is unimodular or f has compact support.

Group Representations

Definition (Representation)

Let G be a locally compact group, and let H be a Hilbert space.

- A representation of G on H is a strongly continuous group homomorphism $\pi\colon G \to L(H)$.
- π is unitary if it takes values in $U(H)$.
- π is irreducible if $\{0\}$ and H are the only invariant closed subspaces of H, where invariance of $V \subseteq H$ means $\pi_g(V) \subseteq V$ for all $g \in G$.
- π is integrable if it is unitary, irreducible, and $\int_G |\langle \pi_g f, f\rangle_H| dg < \infty$ for some $f \in H$. Similarly for square integrability.

Remark: Unless stated otherwise, all integrals over G are with respect to the left Haar measure.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What is a square integrable representation of a locally compact group?
- Check: What condition is more stringent, integrability or square integrability? Hint: $g \mapsto \langle \pi_g f, f \rangle_H$ is continuous and bounded.
- Check: Suppose that π is reducible, can you extract a subrepresentation? Can you reduce it further down to an irreducible subrepresentation?
- Background: How are group representations related to group actions?
- Background: Look up the proof of Young's and Minkowski's inequalities!

Mathematics of Deep Learning, Summer Term 2020

Week 5, Video 3

Signal representations

Philipp Harms Lars Niemann

University of Freiburg

Voice transform

Setting: Throughout, we fix a square-integrable representation $\pi\colon G \to U(H)$ of a locally compact group G on a Hilbert space H.

Definition (Voice transform)

For any $\psi \in H$, the voice transform (aka. representation coefficient) is the linear map

$$V_\psi\colon H \to C(G), \qquad V_\psi f(g) = \langle f, \pi_g \psi \rangle_H.$$

Remark:

- The voice transform represents signals in H as coefficients in $C(G)$.
- For any $\psi \neq 0$, injectivity of V_ψ is equivalent to irreducibility of π.

Orthogonality Relations

Theorem (Duflo–Moore 1976)

There exists a unique densely defined positive self-adjoint operator $A\colon D(A) \subseteq H \to H$ *such that*

- $V_\psi(\psi) \in L^2(G)$ *if and only if* $\psi \in D(A)$, *and*
- *For all* $f_1, f_2 \in H$ *and* $\psi_1, \psi_2 \in D(A)$,

$$\langle V_{\psi_1} f_1, V_{\psi_2} f_2 \rangle_{L^2(G)} = \langle f_1, f_2 \rangle_H \langle A\psi_2, A\psi_1 \rangle_H .$$

G is unimodular if and only if A is bounded, and in this case A is a multiple of the identity.

Remark:

- This is wrong without the square-integrability assumption on π.
- This is difficult to show in general but easy in many specific cases.
- An immediate consequence is the existence (even density) of such ψ.
- $V_\psi\colon H \to L^2(G)$ is isometric for any $\psi \in D(A)$ with $\|A\psi\| = 1$.

Equivalence to the regular representation

Definition (Regular representation)

The left-regular representation of G is the map

$$L\colon G \to U(L^2(G)), \qquad L_g F = F(g^{-1}\cdot).$$

Lemma

π is unitarily equivalent to a sub-representation of the left-regular representation, i.e., there exists an isometry $V : H \to L^2(G)$ such that $V \circ \pi_g = L_g \circ V$ holds for all $g \in G$.

Proof: Set $V = V_\psi$ for some $\psi \in D(A)$ with $\|A\psi\| = 1$ and use that

$$V \circ \pi_{g_1}(f)(g_2) = \langle \pi_{g_1} f, \pi_{g_2}\psi\rangle_H = \langle f, \pi_{g_1^{-1}g_2}\psi\rangle_H = L_{g_1} \circ V(f)(g_2). \qquad \square$$

Analysis, Synthesis, and Reconstruction

Lemma

Let $\psi \in D(A)$ with $\|A\psi\| = 1$.

- *Analysis:* $V_\psi \colon H \to L^2(G)$ *is an isometry onto its range,*

$$V_\psi(H) = \{F \in L^2(G) : F = F * V_\psi\psi\}.$$

- *Synthesis: The adjoint of V_ψ is given by the weak integral*

$$V_\psi^* \colon L^2(G) \to H, \qquad V_\psi^*(F) = \int_G F(g)\pi_g\psi \; dg.$$

- *Reconstruction: Every $f \in H$ satisfies $f = V_\psi^* V_\psi f$.*

Remark:

- This can be seen as a continuous Banach frame.
- The coefficient space is the reproducing kernel Hilbert space $V_\psi(H)$.

Proof: Analysis, Synthesis, and Reconstruction

Proof:

- V_ψ is isometric thanks to the orthogonality relation and $\|A\psi\|_H = 1$.
- V_ψ^* is given by the above weak integral because

$$\langle F, V_\psi f\rangle_{L^2(G)} = \int_G F(g)\langle \pi_g\psi, f\rangle_H dg = \left\langle \int_G F(g)\pi_g\psi \; dg, f \right\rangle_H .$$

- $V_\psi V_\psi^* F = F * V_\psi\psi$ because

$$\begin{aligned} V_\psi V_\psi^* F(g) &= \langle V_\psi^* F, \pi_g\psi\rangle_H = \langle F, V_\psi(\pi_g\psi)\rangle_{L^2(G)} \\ &= \langle F, L_g V_\psi\psi\rangle_{L^2(G)} = (F * V_\psi\psi)(g). \end{aligned}$$

- As V_ψ is isometric, $V_\psi^* V_\psi = \mathrm{Id}_H$ and $V_\psi V_\psi^*$ is the orthogonal projection onto the range of V_ψ, which equals the range of $V_\psi V_\psi^*$. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What is the voice transform, and how does it lead to signal representations?
- Check: Where is square integrability of the representation used?
- Background: There is a definition of continuous frames—can you guess what it is and/or find it in the literature?
- Transfer: What is a reproducing kernel Hilbert space, and what is the relation to the condition $F * V_\psi \psi = F$?

Mathematics of Deep Learning, Summer Term 2020

Week 5, Video 4

Regular Coorbit Spaces

Philipp Harms Lars Niemann

University of Freiburg

Orbits and Coorbits

Setting: $\pi\colon G \to U(H)$ is a square integrable representation of a locally compact group G on a Hilbert space H, and A is the Duflo–Moore operator of π.

Remark:

- The orbit of π through $\psi \in H$ is $\{\pi_g\psi : g \in G\}$.
- V^* extends the action $\pi\colon G \times H \to H$ to

$$V^*\colon L^2(G) \times D(A) \to H, \qquad V_\psi^* F = \int_G F(g)\pi_g\psi \, dg.$$

Definition

Let X be a Banach subspace of $L^2(G)$, and let $\psi \in D(A)$.

- The orbit space associated to X and ψ is the subset $\{V_\psi^* F : F \in X\}$ of H with norm $\|f\| := \inf\{\|F\| : F \in X, V_\psi^* F = f\}$.
- The coorbit space associated to X and ψ is the set of all $f \in H$ such that $V_\psi f \in X$ with norm $\|f\| := \|V_\psi f\|_X$.

Weighted Spaces

Remark:

- The definitions of orbit and coorbit spaces work best when further structure is imposed on X.
- The main examples for X are weighted L^p spaces.

Definition

- A weight function is a continuous function $w\colon G \to \mathbb{R}_+$ which is submultiplicative and symmetric, i.e.,

$$w(gh) \leq w(g)w(h), \qquad w(g) = w(g^{-1}).$$

- The weighted space $L^p_w(G)$, $p \in [1, \infty]$, is defined as

$$L^p_w(G) := \{F : Fw \in L^p(G)\}, \qquad \|F\|_{L^p_w(G)} := \|Fw\|_{L^p(G)}.$$

Remark: $L^p_w(G)$ makes sense for arbitrary measurable functions w.

Properties of Weighted Spaces

Lemma

Let w be a weight function and $p \in [1, \infty]$.

1. *$L^p_w(G)$ is continuously included in $L^p(G)$.*
2. *The space $L^p_w(G)$ is L-invariant.*
3. *L acts strongly continuously on $L^p_w(G)$.*

Proof:

1. The symmetry of w implies $w(g)^2 = w(g)w(g^{-1}) \geq w(e) \geq 1$.
2. The submodularity of w implies that

$$\begin{aligned} \|L_g F\|_{L^p_w(G)} &= \|(L_g F)w\|_{L^p(G)} = \|F(L_{g^{-1}} w)\|_{L^p(G)} \\ &\leq w(g)\|Fw\|_{L^p(G)} = w(g)\|F\|_{L^p_w(G)}. \end{aligned}$$

3. It suffices to verify $\lim_{g \to e} \|L_g F - F\|_{L^2(G)} = 0$ for $F \in C_c(G)$. □

Regular Coorbit Spaces

Remark:

- The following coorbit space $H_{1,w}$ plays the role of test functions in the theory of distributions.
- More general coorbit spaces, which are not subspaces of H, are defined later on.

Definition

Let w be a weight function.

- An analyzing vector is a function $\psi \in D(A)$ with $\|A\psi\|_H = 1$ such that $V_\psi \psi \in L^1_w(G)$.
- $H_{1,w}$ is defined as the coorbit space associated to $L^1_w(G)$ and an analyzing vector ψ, i.e.,

$$H_{1,w} := \{f \in H : V_\psi f \in L^1_w(G)\}, \qquad \|f\|_{H_{1,w}} := \|V_\psi f\|_{L^1_w(G)}.$$

Correspondence Principle

Setting: We fix a weight function w and an analyzing vector ψ.

Theorem

The voice transform is an isometric isomorphism

$$V_\psi \colon H_{1,w} \to \{F \in L^1_w(G) : F = F * V_\psi\psi\}.$$

Proof:

- $X := \{F \in L^1_w(G) : F = F * V_\psi\psi\}$ is well-defined and a Banach subspace of $L^2(G)$ thanks to Young's inequality and $w \geq 1$:

$$\|F * V_\psi\psi\|_{L^2(G)} \leq \|F\|_{L^1(G)}\|V_\psi\psi\|_{L^2(G)} \leq \|F\|_{L^1_w(G)}\|V_\psi\psi\|_{L^2(G)}.$$

- The definition of the orbit and coorbit spaces is unaffected when $L^1_w(G)$ is replaced by X.

□

Independence of the Analyzing Vector

Lemma

$H_{1,w}$ does not depend on the choice of analyzing vector ψ.

Proof:

- Let ψ_1, ψ_2, ψ_3 be analyzing vectors. We will show that $V_{\psi_1} f \in L^1_w(G)$ implies $V_{\psi_3} f \in L^1_w(G)$.
- By the orthogonality relations, one has for any $g \in G$ that

$$\begin{aligned} V_{\psi_1} f * V_{\psi_2}\psi_2(g) &= \langle V_{\psi_1} f, L_g V_{\psi_2}\psi_2\rangle_{L^2(G)} = \langle V_{\psi_1} f, V_{\psi_2}(\pi_g \psi_2)\rangle_{L^2(G)} \\ &= \langle A\psi_2, A\psi_1\rangle_H \langle f, \pi_g\psi_2\rangle_H = \langle A\psi_2, A\psi_1\rangle_H V_{\psi_2} f(g), \end{aligned}$$

$$\begin{aligned} V_{\psi_1} f * V_{\psi_2}\psi_2 * V_{\psi_3}\psi_3 &= \langle A\psi_2, A\psi_1\rangle_H V_{\psi_2} f * V_{\psi_3}\psi_3 \\ &= \langle A\psi_2, A\psi_1\rangle_H \langle A\psi_3, A\psi_2\rangle_H V_{\psi_3} f. \end{aligned}$$

- The left-hand side belongs to $L^1_w(G)$ by Young's inequality. Assuming wlog. that ψ_2 satisfies $\langle A\psi_1, A\psi_2\rangle_H \neq 0 \neq \langle A\psi_2, A\psi_3\rangle_H$, one deduces that $V_{\psi_3} f$ on the right-hand side belongs to $L^1_w(G)$. □

Further Properties

Lemma

$H_{1,w}$ is π-invariant, and π acts strongly continuously on it.

Proof: Correspondence $H_{1,w} \cong X := \{F \in L^1_w(G) : F = F * V_\psi \psi\}$

- $H_{1,w}$ is π-invariant because X is L-invariant.
- π acts strongly continuously on $H_{1,w}$ because L acts strongly continuously on X. □

Lemma

$H_{1,w}$ coincides with the orbit space associated to $L^1_w(G)$ and ψ.

Proof:

- $H_{1,w}$ is an orbit space because $H_{1,w} = V^*_\psi V_\psi H_{1,w} = V^*_\psi L^1_w(G)$. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What is a (regular) coorbit space?
- Check: Are weighted L^p spaces Banach? Do they increase or decrease in p?
- Check: If $\lim_{g \to e} \|L_g F - F\|_{L^2(G)} = 0$ holds for all F in a dense subset of $L^2(G)$, why does it then hold for all F?

Mathematics of Deep Learning, Summer Term 2020

Week 5, Video 5

Duals of Coorbit Spaces

Philipp Harms Lars Niemann

University of Freiburg

Gelfand triples

Definition

A Gelfand triple is a triple (K, H, K^*), where K is a topological vector space, which is densely and continuously included in a Hilbert space H.

Lemma

Let (K, H, K^*) be a Gelfand triple. Then the inner product $\langle \cdot, \cdot \rangle_H$ extends to a sesquilinear form on $K^* \times K$.

Proof: Let $i \colon K \to H$ be the inclusion, and let $j = \langle \cdot, \cdot \rangle_H \colon H \to H^*$. Then $i^* \colon H^* \to K^*$ is injective because i has dense range, $i^* \circ j$ includes H into K^*, and the desired extension is just the duality $K^* \times K \to \mathbb{R}$. $\square$

Gelfand Triples of Coorbit Spaces

Setting: $\pi : G \to U(H)$ is a square-integrable representation with Duflo–Moore operator A, w is a weight function, and ψ is an analyzing vector.

Lemma

The spaces $(H_{1,w}, H, H_{1,w}^*)$ form a Gelfand triple.

Proof:

- $H_{1,w}$ is isomorphic via the voice transform to the space $\{F \in L^1_w(G) : F = F * V_\psi\psi\}$, which is continuously included in the space $\{F \in L^2(G) : F = F * V_\psi\psi\}$, which is isomorphic via the inverse voice transform to H.
- $H_{1,w}$ contains the orbit $\{\pi_g\psi : g \in G\}$ because

$$\|\pi_g\psi\|_{H_{1,w}} = \|V_\psi(\pi_g\psi\|_{L^1_w(G)} = \|L_g V_\psi\psi\|_{L^1_w(G)} \lesssim \|V_\psi\psi\|_{L^1_w(G)} < \infty.$$

The orbit is dense in H because π is irreducible. □

Duals of Coorbit Spaces

Remark: As $H_{1,w}$ plays the role of test functions, $H_{1,w}^*$ plays the role of distributions.

Definition

The extended voice transform is defined for any $f \in H_{1,w}^*$ and $g \in G$ as

$$V_\psi(f)(g) := \langle f, \pi_g \psi \rangle_{H_{1,w}^* \times H_{1,w}} .$$

Remark: This extends the voice transform on H because the dual pairing between $H_{1,w}^*$ and $H_{1,w}$ extends the inner product on H.

Correspondence Principle

Remark: $L^1_w(G)^* = L^\infty_{1/w}(G)$.

Theorem (Correspondence principle)

$V_\psi : H^*_{1,w} \to \{F \in L^\infty_{1/w} : F = F * V_\psi\psi\}$ is an isometric isomorphism.

Proof: In the proof of the correspondence principle for the regular voice transform, replace the Hilbert inner product on H by the dual pairing between $H^*_{1,w}$ and $H_{1,w}$. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How does the voice transform extend to duals of coorbit spaces?
- Check: If (K, H, K^*) is a Gelfand triple, and H is seen as a subspace of K^*, how are elements of H applied to elements of K?
- Check: Prove that the topological dual of $L^1_w(G)$ is $L^\infty_{1/w}(G)$.
- Transfer: What Gelfand triples are used to define distributions and tempered distributions?

Mathematics of Deep Learning, Summer Term 2020

Week 5, Video 6

General Coorbit Spaces

Philipp Harms Lars Niemann

University of Freiburg

Weighted Spaces

Setting: $\pi\colon G \to U(H)$ is a square-integrable representation with Duflo–Moore operator A, w is a weight function, and ψ is an analyzing vector subject to some further conditions.[1]

Definition

- A w-moderate weight is a continuous function $m\colon G \to \mathbb{R}_+$ satisfying

$$m(ghk) \leq w(g)m(h)w(k), \qquad g, h, k \in G.$$

- The weighted space $L^p_m(G)$ is defined for any $p \in [1, \infty]$ as

$$L^p_m(G) := \{F : Fm \in L^p(G)\}, \qquad \|F\|_{L^p_m(G)} := \|Fm\|_{L^p(G)}.$$

Remark:

- This extends the def. of $L^p_w(G)$ since w is a w-moderate weight.
- $\|\cdot\|_{L^p_w(G)}$ is a norm, but $\|\cdot\|_{L^p_m(G)}$ may be only a seminorm.

[1] See Theorem 3.12 in Dahlke, De Mari, Grohs, Labatte (2015).

Coorbit Spaces

Setting: We fix a w-moderate weight m.

Definition

The coorbit space $H_{p,m}$ is defined as

$$H_{p,m} := \{F \in H^*_{1,w} : V_\psi(F) \in L^p_m(G)\}\,.$$

Remark:

- This extends the definition of $H_{1,w}$, and $H = H_{2,1}$.
- $H_{p,m}$ is independent of the choice of analyzing vector ψ.
- $H_{p,m}$ coincides as a set with an orbit space.

Theorem (Correspondence principle)

Under an additional condition on ψ, the voice transform $V_\psi \colon H_{p,m} \to \{F \in L^p_m(G) : F = F * V_\psi\psi\}$ is an isometric isomorphism.

Structure of Coorbit Spaces

Uniqueness: $H_{p_1,m_1} = H_{p_2,m_2}$ if and only if $p_1 = p_2$ and $m_1 \lesssim m_2 \lesssim m_1$.

Duality: $H^*_{p,m} = H_{q,1/m}$ for any $p \in [1,\infty)$ and $\frac{1}{p} + \frac{1}{q} = 1$.

Embeddings: $H_{p,m}$ is increasing in p and decreasing in m.

Compact Embeddings: H_{p_1,m_1} embeds compactly in H_{p_2,m_2} if $m_1/m_2 \in L^r(G)$ for some $r \leq \frac{1}{p_2} - \frac{1}{p_1} > 0$.

Complex Interpolation: For any $\theta \in [0,1]$ and $p_1 < \infty$, $[H_{p_1,m_1}, H_{p_2,m_2}]_\theta = H_{p,m}$ with $\frac{1}{p} = \frac{1-\theta}{p_1} + \frac{\theta}{p_2}$ and $m = m_1^{1-\theta} m_2^\theta$.

Generalizations: $L^p_m(G)$ is a left- and right-invariant solid Banach function space on G, and coorbit spaces can be defined for such spaces.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How are (general) coorbit spaces $H_{p,m}$ defined?
- Check: $H_{p,m} \subseteq H^*_{1,w}$ implies $L^p_m(G) \subseteq L^1_w(G)^*$—how can this be seen directly? Hint: show that $m(e) = m(gg^{-1}) \lesssim m(g)w(g^{-1})$.
- Background: Read up on duality, embedding, and interpolation properties of L^p spaces.

Mathematics of Deep Learning, Summer Term 2020

Week 5, Video 7

Discretization

Philipp Harms Lars Niemann

University of Freiburg

Towards Banach Frames on Coorbit Spaces

Setting: $\pi : G \to U(H)$ is a square-integrable representation with Duflo–Moore operator A, w is a weight function, m is a w-moderate weight, $p \in [1, \infty]$, and ψ is an analyzing vector subject to some further conditions.[2]

Strategy:

- Define a Banach frame for $\{F \in L^p_m(G) : F = F * V_\psi \psi\}$ via left-translations of the kernel $V_\psi \psi$, i.e., by writing

$$F = \sum_k c_k(F) L_{g_k} V_\psi \psi$$

  for a well-chosen sequence of $g_k \in G$.
- Get a Banach frame for $H_{p,m}$ via the correspondence principle.

[2] See Theorem 3.19 in Dahlke, De Mari, Grohs, Labatte (2015).

Density and Separation

Remark: Intuitively, translations of a kernel by (g_k) are a frame if (g_k) spreads out over all of G and does not accumulate anywhere.

Definition

A sequence $(g_k)_{k\in\mathbb{N}}$ in G is called

- U-dense if U is a compact neighborhood of $e \in G$ and $\bigcup_k L_{g_k} U = G$.
- separated if there exists a compact neighborhood U of $e \in G$ such that $L_{g_k} U \cap L_{g_l} U = \emptyset$ for $k \neq l$.
- relatively separated if it is a finite union of separated sequences.

Banach Frames on Weighted Spaces

Definition

The weighted sequence space ℓ^p_m is defined as

$$\ell^p_m := \{\lambda \colon \lambda m \in \ell^p\}, \qquad \|\lambda\|_{\ell^p_m} := \|\lambda m\|_{\ell^p}.$$

Theorem

If U is a sufficiently small neighborhood of $e \in G$ and (g_k) is a U-dense and relatively separated sequence in G, then $(L_{g_k} V_\psi \psi)_{k\in\mathbb{N}}$ is a Banach frame for $X := \{F \in L^p_m(G) : F = F * V_\psi \psi\}$ with respect to ℓ^p_m.

Remark: the frame coefficients are specified in the proof.

Proof: Banach Frames on Weighted Spaces

Proof for $p = 1$ and $m = w$:

- Let (Ψ_k) be a partition of unity subordinated to $(L_{g_k} U)$.
- We define some preliminary analysis and synthesis operators:

$$X \ni F \mapsto (\langle \Psi_k, F \rangle_{L^2(G)})_{k \in \mathbb{N}} \in \ell^1_w, \qquad \ell^1_w \ni \lambda \mapsto \sum_k \lambda_k L_{g_k} V_\psi \psi \in X.$$

- These operators are well-defined and continuous: letting $C := \sup_{g \in U} w(z)$, one has

$$\begin{aligned}
\|(\langle \Psi_k, F \rangle_{L^2(G)})_{k \in \mathbb{N}}\|_{\ell^1_w} &= \sum_k |\langle \Psi_k, F \rangle_{L^2(G)}| w(g_k) \\
&\leq C \sum_k \langle \Psi_k, |F| w \rangle_{L^2(G)} = C \|F\|_{L^1_w(G)}, \\
\Big\| \sum_k \lambda_k L_{g_k} V_\psi \psi \Big\|_{L^1_w(G)} &\leq \sum_k |\lambda_k| \|L_{g_k} V_\psi \psi\|_{L^1_w(G)} \\
&\leq \sum_k |\lambda_k| w(g_k) \|V_\psi \psi\|_{L^1_w(G)} = \|\lambda\|_{\ell^1_w} \|V_\psi \psi\|_{L^1_w(G)}.
\end{aligned}$$

Proof: Banach Frames on Weighted Spaces (cont.)

- The reconstruction operator (i.e., analysis followed by synthesis),

$$R\colon X \to X, \qquad RF := \sum_{k\in\mathbb{N}} \langle F, \Psi_k\rangle L_{g_k} V_\psi \psi,$$

tends to Id_X as U tends to $\{e\}$ because for any $F \in X$,

$$\begin{aligned}
&\left\| F * V_\psi\psi - \sum_k \langle \Psi_k, F\rangle_{L^2(G)} L_{g_k} V_\psi\psi \right\|_{L^1_w(G)} \\
&\quad = \left\| \sum_k \int_G F(g)\Psi_k(g)(L_g - L_{g_k})V_\psi\psi dg \right\|_{L^1_w(G)} \\
&\quad \le \sum_k \langle \Psi_k, |F|\rangle_{L^2(G)} \sup_{g\in L_{g_k}U} \|(L_g - L_{g_k})V_\psi\psi\|_{L^1_w(G)} \\
&\quad \le \sum_k \langle \Psi_k, |F|\rangle_{L^2(G)} w(g_k) \sup_{u\in U} \|(L_u - \mathrm{Id})V_\psi\psi\|_{L^1_w(G)} \\
&\quad \le C\|F\|_{L^1_w(G)} \sup_{u\in U} \|(L_u - \mathrm{Id})V_\psi\psi\|_{L^1_w(G)} \to 0.
\end{aligned}$$

Proof: Banach Frames on Weighted Spaces (cont.)

- R is invertible for sufficiently small U because Id_X is invertible and invertible operators are open.
- Any $F \in X$ can be written as

$$F = RR^{-1}F = \sum_{k\in\mathbb{N}} \langle \Psi_k, R^{-1}F\rangle_{L^2(G)} L_{g_k} V_\psi \psi.$$

- Thus, the desired Banach frame for X with respect to ℓ^1_w is

$$e_k := L_{g_k} V_\psi \psi \in X, \quad c_k := \langle \Psi_k, R^{-1}(\cdot)\rangle_{L^2(G)} \in X^*, \quad k \in \mathbb{N}. \qquad \square$$

Banach Frames for Coorbit Spaces

Corollary

If U is a sufficiently small neighborhood of $e \in G$ and (g_k) is a U-dense and relatively separated sequence in G, then $(\pi_{g_k}\psi)_{k\in\mathbb{N}}$ is a Banach frame for $H_{p,m}$ with respect to ℓ^p_m.

Proof: Apply the isomorphism $V_\psi^{-1}\colon X \to H_{p,m}$. □

Harmonic Analysis and Neural Networks

- Let G be a sub-group of the affine group $GL(\mathbb{R}^d) \ltimes \mathbb{R}^d$, and define

$$\pi \colon G \to U(L^2(\mathbb{R}^d)), \qquad \pi_{(A,b)}(f)(x) = \det(A)^{-1/2} f(A^{-1}(x-b)).$$

- Then coorbit theory provides continuous and discrete representations

$$\begin{aligned} f(x) &= \int_G F(A,b) \det(A)^{-1/2} \psi(A^{-1}(x-b)) dA db \\ &= \sum_k c_k \det(A_k)^{-1/2} \psi(A_k^{-1}(x-b_k)), \end{aligned}$$

  where ψ is a suitable analyzing vector, with an equivalence of norms

$$\|F\|_{L^p_m(G)} \simeq \|c_k\|_{\ell^p_m} \simeq \|f\|_{H_{p,m}}.$$

- These representations can be interpreted as infinite-width multi-layer perceptrons with activation function ψ.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How are Banach frames of weighted spaces and coorbit spaces constructed?
- Background: Refresh your memory of the definition and construction of partitions of unity.
- Check: Why is the set of invertible operators open in the set of bounded linear operators?
- Discussion: How could coorbit theory be used to derive approximation bounds of neural networks?

Mathematics of Deep Learning, Summer Term 2020

Week 5, Video 8

Wrapup

Philipp Harms Lars Niemann

University of Freiburg

Outlook on this week's discussion and reading session

- Reading:
  - Feichtinger Groechenig (1988): A unified approach to atomic decompositions
  - Dahlke, De Mari, Grohs, Labatte (2015): Harmonic and Applied Analysis

- Numerical Example:
  - Some wavelet transforms in image analysis.

Summary by learning goals

Having heard this lecture, you can now. . .

- Describe bases and frames in Hilbert and Banach spaces.
- Build signal representations from group representations.
- Interpret such representations as multi-layer perceptrons.

Mathematics of Deep Learning, Summer Term 2020

Week 6

Signal Analysis

Philipp Harms Lars Niemann

University of Freiburg

Overview of Week 6

1. Coorbit Theory, Signal Analysis, and Deep Learning
2. Heisenberg Group
3. Modulation Spaces
4. Affine Group
5. Wavelet Spaces
6. Shearlet Group
7. Shearlet Coorbit Spaces
8. Wrapup

Acknowledgement of Sources

Sources for this lecture:

- Christensen (2016): An introduction to frames and Riesz bases
- Dahlke, De Mari, Grohs, Labatte (2015): Harmonic and Applied Analysis
- Feichtinger Gröchenig (1988): A unified approach to atomic decompositions
- Folland (2016): A course in abstract harmonic analysis

Mathematics of Deep Learning, Summer Term 2020

Week 6, Video 1

Coorbit Theory, Signal Analysis, and Deep Learning

Philipp Harms Lars Niemann

University of Freiburg

Harmonic Analysis

- Setting: $\pi\colon G \to U(H)$ is a strongly continuous irreducible unitary representation of a locally compact group G on a Hilbert space H such that $\int |\langle \pi_g f, f\rangle_H|^2 dg < \infty$ for some $\psi \in H$.

- Voice transform: For any $\psi \in H$, the voice transform is the linear map

$$V_\psi\colon H \to C(G), \qquad V_\psi f(g) = \langle f, \pi_g \psi\rangle_H.$$

- Admissibility: the voice transform V_ψ is isometric for all $\psi \in D(A)$ with $\|A\psi\|_H = 1$, where A is the Duflo–Moore operator. These ψ are called admissible.

- Reproducing kernel spaces: for any admissible ψ, the voice transform is an isometric isomorphism onto the space

$$\{F \in L^2(G) : F * V_\psi\psi = F\}$$

with reproducing kernel $V_\psi\psi$.

Coorbit Theory

- Weighted spaces: for exponents $p \in [1, \infty]$ and w-moderate weight functions $m \colon G \to \mathbb{R}_+$, one defines weighted spaces $L^p_w(G)$ and $L^p_m(G)$, respectively.
- Analyzing vectors are defined as admissible ψ with $V_\psi \psi \in L^1_w(G)$.
- Coorbit spaces $H_{p,m}$ are constructed by requiring the voice transform to be an isomorphism for some (equivalently, all) analyzing vectors ψ:

$$V_\psi \colon H_{p,m} \xrightarrow{\cong} \{F \in L^p_m(G) : F * V_\psi \psi = F\}.$$

- Banach frames: for suitable analyzing vectors $\psi \in D(A)$ and group elements $(g_k)_{k \in \mathbb{N}}$, one obtains a Banach frame $(\pi_{g_k} \psi)_{k \in \mathbb{N}}$ for the coorbit space $H_{p,m}$ with respect to a weighted sequence space ℓ^p_m.
- Proof by correspondence principle: $(L_{g_k} V_\psi \psi)_{k \in \mathbb{N}}$ is a Banach frame for $\{F \in L^p_m(G) : F * V_\psi \psi = F\}$ with respect to ℓ^p_m.

Abelian Groups are not Interesting for Coorbit Theory

Theorem

Abelian groups have only one-dimensional irreducible representations.

Lemma (Schur)

$\pi : G \to U(H)$ is irreducible if and only if its centralizer is trivial, i.e.,

$$\{T \in L(H) : \pi_g T = T\pi_g \text{ for all } g \in G\} = \operatorname{span}\{\mathrm{Id}_H\}.$$

Proof of the Theorem:

- The centralizer of π is trivial because π is irreducible.
- The operators π_g belong to the centralizer because G is Abelian.
- Thus, the operators π_g are multiples of the identity.
- Thus, all one-dimensional subspaces are invariant. □

Signal analysis and Deep Learning

Signal Analysis:

- There are many different group representations with associated voice transforms.
- These have a variety of applications in signal analysis such as time-frequency analysis, multi-resolution analysis, and edge detection.
- The interpretation varies strongly from case to case.

Deep learning inherits many of the strengths of signal analysis:

- Many voice transforms are implementable via shallow nets with activation function equal to the analyzing function.
- Alternatively, via dictionary learning, they are implementable via deep nets with other activation functions.
- In this case, deep learning can adaptively select (i.e., learn) a suitable analyzing function.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Refresh your memory of the voice transform and the construction of coorbit spaces.
- Check: As the translation group is Abelian, its representation on $L^2(\mathbb{R}^d)$ must be reducible—can you find a subrepresentation?
- Check: Same question for the modulation group. Hint: apply the Fourier transform.
- Check: How can dictionary learning be applied to implement signal transforms via deep networks?
- Background: Look up the proof of Schur's lemma. For instance, in [Christensen], [Dahlke e.a.], or [Folland].

Mathematics of Deep Learning, Summer Term 2020

Week 6, Video 2

Heisenberg Group

Philipp Harms Lars Niemann

University of Freiburg

Structure

Definition

The Heisenberg group is the set $G := \mathbb{R}^d \times \mathbb{R}^d \times S^1$ equipped with the product topology and the composition

$$(a_1, b_1, t_1) \cdot (a_2, b_2, t_2) := (a_1 + a_2, b_1 + b_2, t_1 t_2 e^{2\pi i b_1 a_2}) .$$

Properties:

- The Heisenberg group is not Abelian.
- The Haar measure is the product measure of the three involved Lebesgue measures.
- The Heisenberg group is unimodular.

Representation

Definition

The Schrödinger representation $\pi\colon G \to U(L^2(\mathbb{R}^d))$ is defined as

$$\pi(a,b,t)f(x) := te^{2\pi ib(x-a)}f(x-a)\,,$$

where $f \in L^2(\mathbb{R}^d)$, $(a,b,t) \in G$, and $x \in \mathbb{R}^d$.

Remark:

- π can be expressed in terms of translation and modulation as

$$\pi(a,b,t)f = te^{-2\pi iab}E_bT_af\,.$$

  Translations are time shifts, and modulations are frequency shifts.
- π is irreducible and integrable.
- All unit vectors in $L^2(\mathbb{R}^d)$ are admissible because G is unimodular.

Gabor Transform

Remark:

- The Gabor transform or short-time Fourier transform is the voice transform of the Schrödinger representation.
- The torus component $t \in S^1$ can (and will) be ignored for all practical purposes.

Definition

For any admissible $\psi \in L^2(\mathbb{R}^d)$, the Gabor transform $V_\psi \colon L^2(\mathbb{R}^d) \to L^2(\mathbb{R}^{2d})$ is given by

$$V_\psi f(a,b) := \int_{\mathbb{R}^d} f(x)\psi(x-a)e^{-2\pi i x b}dx = \langle f, E_b T_a \psi \rangle_{L^2(\mathbb{R}^d)},$$

where $f \in L^2(\mathbb{R}^d)$ and $a, b \in \mathbb{R}^d$.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Describe the Schrödinger representation of the Heisenberg group. Think about a way of memorizing the group structure.
- Check: Why can the torus component be ignored for the purpose of signal analysis?

Mathematics of Deep Learning, Summer Term 2020

Week 6, Video 3

Modulation Spaces

Philipp Harms Lars Niemann

University of Freiburg

Analyzing Functions

Setting: We consider the Schrödinger representation π of the Heisenberg group G on $L^2(\mathbb{R}^d)$.

Lemma

Let w be a weight function on G. A function $\psi \in L^2(\mathbb{R}^d)$ is an analyzing vector for w if and only if $\|\psi\| = 1$ and

$$\int_{\mathbb{R}^d} \int_{\mathbb{R}^d} |\langle \psi, E_b T_a \psi \rangle| \, w(a,b) \, da \, db < \infty \, .$$

Remark:

- The Feichtinger algebra $\mathcal{S}_0$ is defined as the subspace of $L^2(\mathbb{R}^d)$ described by the above integrability condition with $w \equiv 1$.
- The Gauss function is analyzing[1] for all polynomial weight functions $w(a,b) := (1 + \|b\|)^{|s|}$, $s \in \mathbb{R}$.

[1] See [Feichtinger Gröchenig 1988, Section 7.1].

Gabor coorbit spaces

Remark: Gabor coorbit spaces are called modulation spaces:

Definition

Let $d \in \mathbb{N}$, let m be a w-moderate weight, and let ψ be an analyzing vector for w. For any $1 \leq p, q \leq \infty$, the modulation space $M_m^{p,q}$ consists of all tempered distributions $f \in \mathcal{S}'$ such that

$$\int \left(\int |\langle f, E_b T_a \psi \rangle|^p \, m(a,b)^p da \right)^{q/p} db < \infty \,,$$

with the usual modifications for $p, q \in \{\infty\}$.

Remark:

- This definition is independent of the choice of w and ψ.
- For $p = q$, we write $M_m^p := M_m^{p,p}$.

Properties and Examples

The Feichtinger algebra provides a rich repertoire of analyzing vectors because it

- Contains all $f \in C_c(\mathbb{R}^d)$ with $\mathcal{F}f \in L^1(\mathbb{R}^d)$.
- Contains the Schwartz space of rapidly decreasing functions.
- Is invariant under the Heisenberg group and the Fourier transform.

Modulation spaces with constant weights $m \equiv 1$:

- M^1_m is the Feichtinger algebra $\mathcal{S}_0$.
- M^2_m is the space $L^2(\mathbb{R}^d)$.

Modulation spaces with polynomial weights $m(a,b) := (1 + \|b\|)^s$:

- M^2_m is the Sobolev (aka. Bessel potential) space $H^s(\mathbb{R}^d)$, for any $s \in \mathbb{R}$. This follows from the respective characterization via frames.

Gabor Frames

Theorem

Let $p \in [1, \infty)$, let $s \in \mathbb{R}$, let $w(a,b) := (1 + \|b\|)^{|s|}$, and let $m(a,b) := (1 + \|b\|)^s$. For any $\psi \in M^{1,1}_w \setminus \{0\}$ and sufficiently small $\alpha, \beta > 0$, the vectors $(E_{\beta b} T_{\alpha a} \psi)_{a,b \in \mathbb{Z}^d}$ form a Banach frame for M^p_m with respect to the sequence space

$$\ell^p_m := \left\{ (\lambda_{a,b})_{a,b \in \mathbb{Z}^d} : \|\lambda\|^p_{\ell^p_m} := \sum_{a,b \in \mathbb{Z}^d} |\lambda_{a,b}|^p (1 + \|b\|)^{sp} < \infty \right\}.$$

Proof: For this choice of weight function, no further conditions[2] on the analyzing vector ψ are needed. □

Remark: The result is independent of the enumeration of $a, b \in \mathbb{Z}^d$ because the sum in the ℓ^p_m norm converges unconditionally.

[2] See Theorem 3.19 in Dahlke, De Mari, Grohs, Labatte (2015).

Gabor Frames for Time-Frequency Analysis

Remark: Gabor frames (equivalently, the short-time Fourier transform) define a uniform tiling of the time-frequency domain:

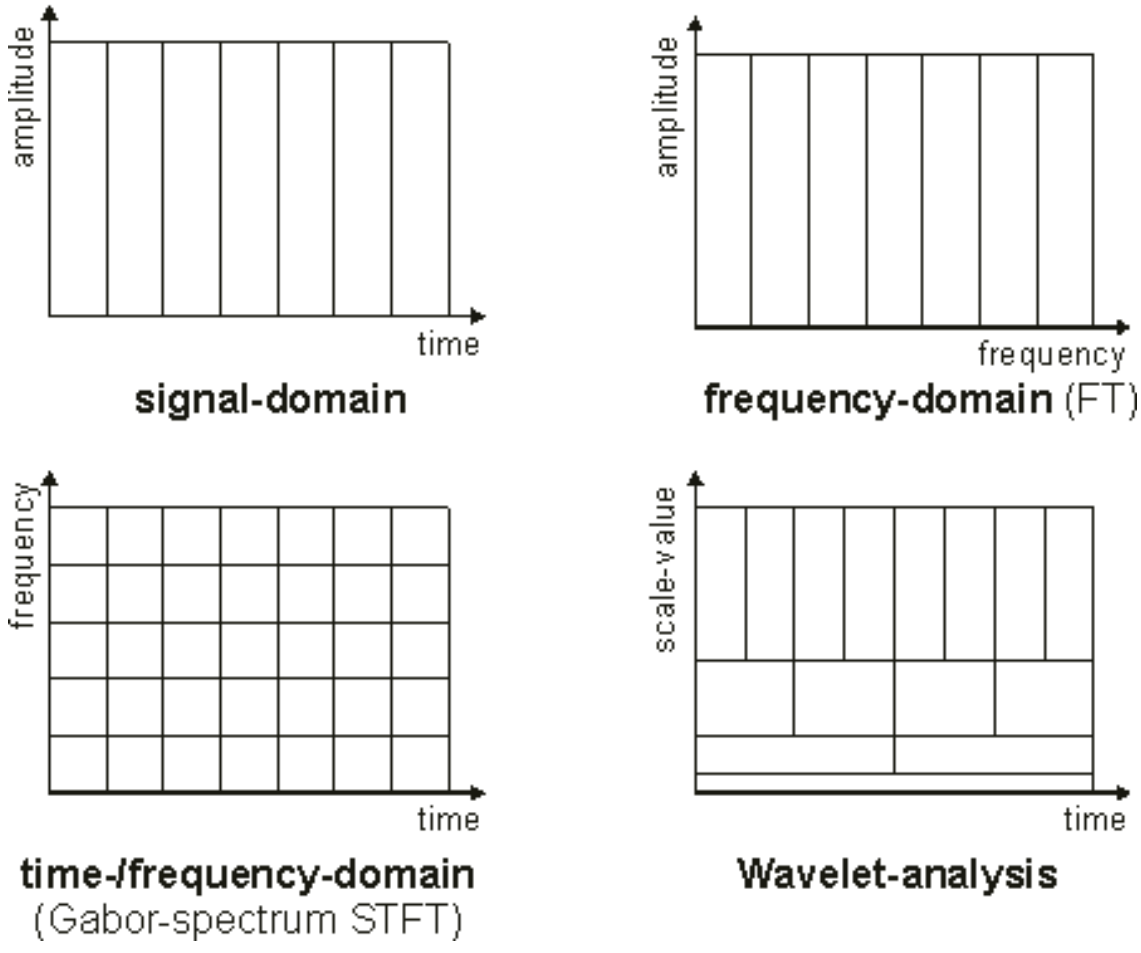


Figure: [www.ndt.net/article/v07n09/08]

Gabor Frames for Time-Frequency Analysis

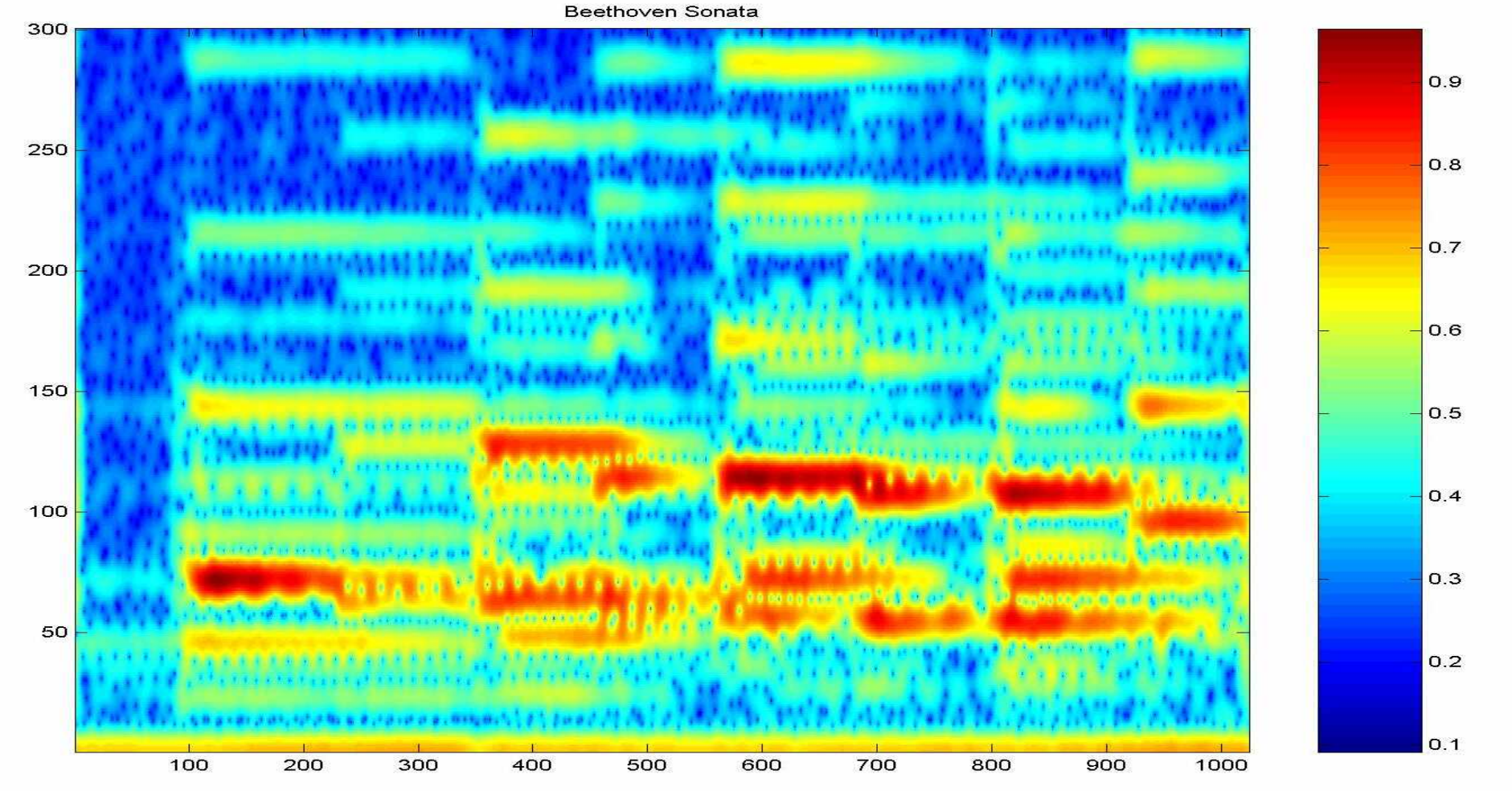


Figure: Intensity (color-coded) of an audio signal, plotted over time (horizontal) and frequency (vertical). [Feichtinger (2015): Wiener Amalgams and Gabor Analysis]

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Describe the Gabor transform, modulation spaces, and their role in signal analysis.
- Check: Compute the analyzing condition more explicitly. Hint: express the integral da by a convolution and apply the Fourier transform; see [Feichtinger Gröchenig (1988), Section 7.1].
- Background: Read up on the Gabor transform and short-time Fourier transform.

Mathematics of Deep Learning, Summer Term 2020

Week 6, Video 4

Affine Group

Philipp Harms Lars Niemann

University of Freiburg

Structure

Definition

The affine group is the set $G := (\mathbb{R} \setminus \{0\}) \times \mathbb{R}$ equipped with the product topology and the composition

$$(a', b') \cdot (a, b) := (a'a, a'b + b') .$$

Properties:

- This corresponds to the composition of affine maps.
- The affine group is not Abelian.
- The left Haar measure is $\frac{1}{|a|^2} da\, db$, and the right Haar measure is $\frac{1}{|a|} da\, db$, where $da\, db$ denotes the Lebesgue measure on $\mathbb{R}^2$.
- In particular, the group is not unimodular.

Representation

Definition

The affine representation $\pi : G \to U(L^2(\mathbb{R}))$ is defined as

$$\pi(b,a)f(y) := \frac{1}{\sqrt{|a|}} f\Big(\frac{y-b}{a}\Big), \quad f \in L^2(\mathbb{R}), \quad (b,a) \in G, \quad y \in \mathbb{R}\,.$$

Remark:

- π can be expressed in terms of translation and dilation as

$$\pi(a,b)f = T_b D_a f\,.$$

- The representation π is irreducible and integrable.[1]

[1] Irreducibility fails for the connected subgroup $\mathbb{R}_{>0} \times \mathbb{R}$.

Admissibility

Lemma

The Duflo–Moore operator associated to π is given by

$$Af(\xi) := \frac{\mathcal{F}f(\xi)}{\sqrt{|\xi|}}, \qquad \xi \in \mathbb{R},$$

and is defined for all f in

$$D(A) := \left\{ f \in L^2(\mathbb{R}) : \int_{\mathbb{R}} \frac{|\mathcal{F}f(\xi)|^2}{|\xi|}\, d\xi < \infty \right\}.$$

Remark: Thus, a function $\psi \in L^2(\mathbb{R})$ is admissible if and only if it satisfies the Calderón equation[2]

$$\int_{\mathbb{R}} \frac{|\mathcal{F}\psi(\xi)|^2}{|\xi|} d\xi = 1.$$

[2] See [Dahlke e.a., Example 2.48.]

Wavelet Transform

Remark:

- Admissible vectors are called wavelets.
- The wavelet transform is the voice transform of the affine representation.

Definition

For any admissible $\psi \in L^2(\mathbb{R})$, the wavelet transform $V_\psi \colon L^2(\mathbb{R}) \to L^2(G)$ is given by

$$V_\psi f(a,b) := \frac{1}{\sqrt{|a|}} \int_{\mathbb{R}} f(x) \overline{\psi\big(\frac{x-b}{a}\big)}\, dx \ .$$

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Describe the representation of the affine group.
- Background: Read the computation of the Duflo–Moore operator. See [Dahlke e.a. (2015), Example 2.48].
- Check: What goes wrong when the affine group is replaced by the connected subgroup $\mathbb{R}_{>0} \times \mathbb{R}$? Hint: see the computation of the Duflo–Moore operator.
- Check: What goes wrong for affine groups in higher dimension. Hint: see the computation of the Duflo–Moore operator.
- Discussion: Can you think of a sub-group of the affine group which has an integrable representation in higher dimension? Hint: restrict to scalar multiples of orthogonal matrices.

Mathematics of Deep Learning, Summer Term 2020

Week 6, Video 5

Wavelet Spaces

Philipp Harms Lars Niemann

University of Freiburg

Analyzing functions

Setting: We consider the representation π of affine group G on $L^2(\mathbb{R})$.

Lemma

Let w be a weight function on G. A function $\psi \in L^2(\mathbb{R})$ is an analyzing vector for w if and only if $\|A\psi\| = 1$ and

$$\int_G |\langle \psi, T_b D_a \psi \rangle| \, w(a,b) \, \frac{da \, db}{|a|^2} < \infty \, .$$

Examples:[1]

- Schwartz functions whose Fourier transform is compactly supported in $\mathbb{R} \setminus \{0\}$ are analyzing for any weight function.
- Compactly supported functions with sufficient smoothness and sufficiently many vanishing moments are analyzing for weight functions of the form $w(a,b) := |a|^s + |a|^{-s}$.

[1] See [Dahlke e.a., Theorems 3.24 and 3.35].

Wavelet Coorbit Spaces

Definition

Let m be a w-moderate weight, and let ψ be an analyzing vector for w. For any $p \in [1, \infty]$, the wavelet coorbit space $H_{p,m}$ consists of all tempered distributions $f \in \mathcal{S}'$ such that

$$\int_G |\langle f, T_b D_a \psi \rangle|^p \, m(a,b)^p \frac{da\, db}{|a|^2} < \infty \,,$$

with the usual modification for $p = \infty$.

Remark:

- This definition is independent of the choice of w and ψ.
- The main example is $m(a,b) = |a|^{-s}$ with $s \in \mathbb{R}$, and in this case $H_{p,m}$ coincides[2] with the homogeneous Besov space $\dot{B}_{p,p}^{s-1/2-1/p}$.

[2] See [Feichtinger Gröchenig 1998] or [Dahlke e.a. 2015]

Wavelet Frames

Theorem

Let $p \in [1, \infty)$, $s \in \mathbb{R}$, $w(a,b) := |a|^s + |a|^{-s}$, and $m(a,b) := |a|^{-s}$. For any w-admissible symmetric ψ subject to some further conditions[3] *and sufficiently small $\alpha > 1$ and $\beta > 0$, the vectors $(T_{\alpha^a \beta b} D_{\alpha^a} \psi)_{a,b \in \mathbb{Z}}$ form a Banach frame for $H_{p,m}$ with respect to the sequence space*

$$\ell^p_m := \left\{ (\lambda_{a,b})_{a,b \in \mathbb{Z}} : \|\lambda\|^p_{\ell^p_m} := \sum_{a,b \in \mathbb{Z}} |\lambda_{a,b}|^p \alpha^{-asp} < \infty \right\}.$$

Proof: For any given U and sufficiently small $\alpha > 1$ and $\beta > 0$, the sequence $(\epsilon \alpha^a, \epsilon \alpha^a \beta b)_{\epsilon \in \{-1,1\}, a \in \mathbb{Z}, b \in \mathbb{Z}}$ is U-dense and relatively separated. □

[3] See Theorem 3.19 in Dahlke, De Mari, Grohs, Labatte (2015).

Wavelet Frames for Multi-Resolution Analysis

Remark: Wavelet frames define a non-uniform tiling of the time-frequency domain, which corresponds to fast sampling of high frequencies and slow sampling of low frequencies.

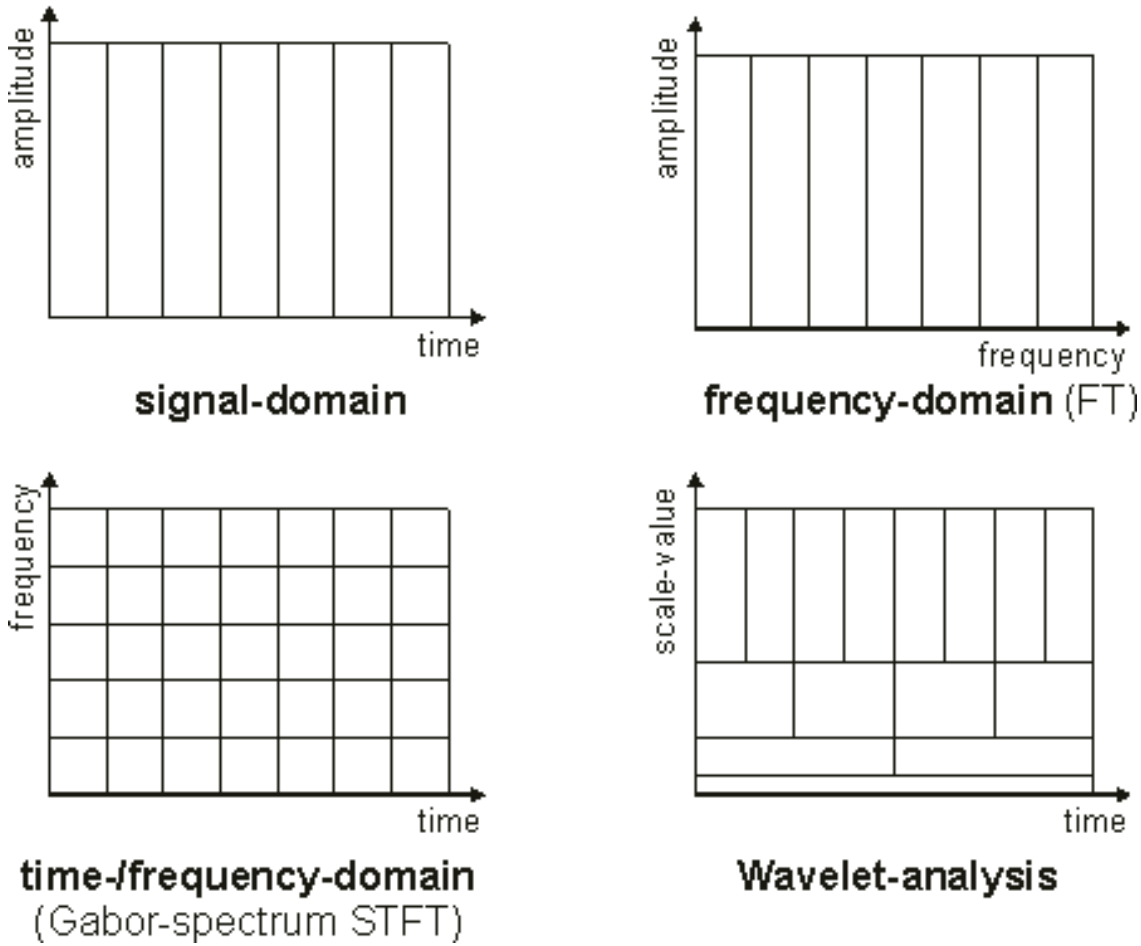


Figure: [www.ndt.net/article/v07n09/08]

Wavelet Frames for Multi-Resolution Analysis

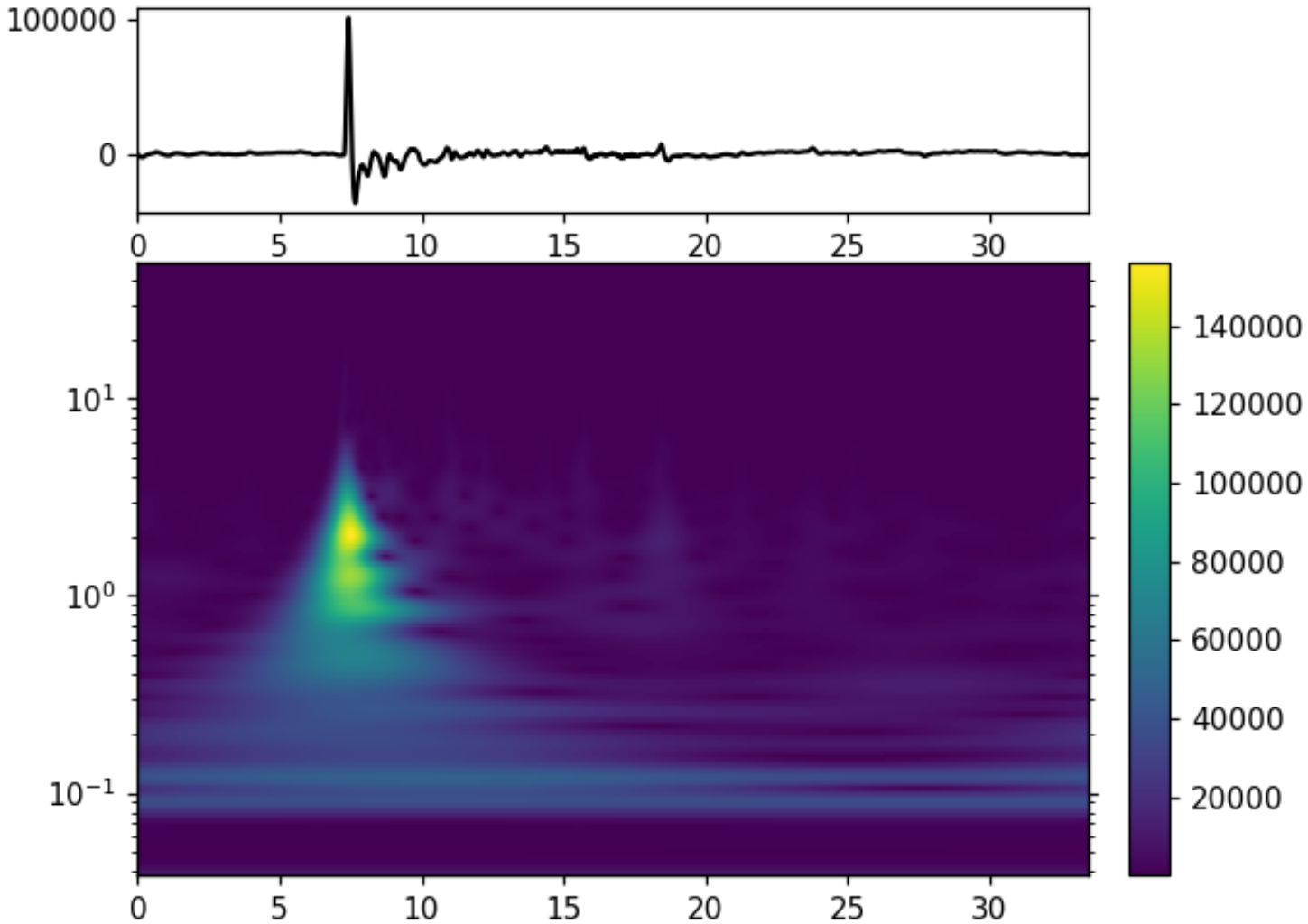


Figure: Top: A seismic signal. Bottom: The signal intensity (color-coded) plotted over time (horizontal) and scale (vertical). From `obspy.org`

Application to Image Analysis

Remark: The JPEG2000 standard uses lossy compression based on Cohen–Daubechies–Feauveau (CDF) wavelets.

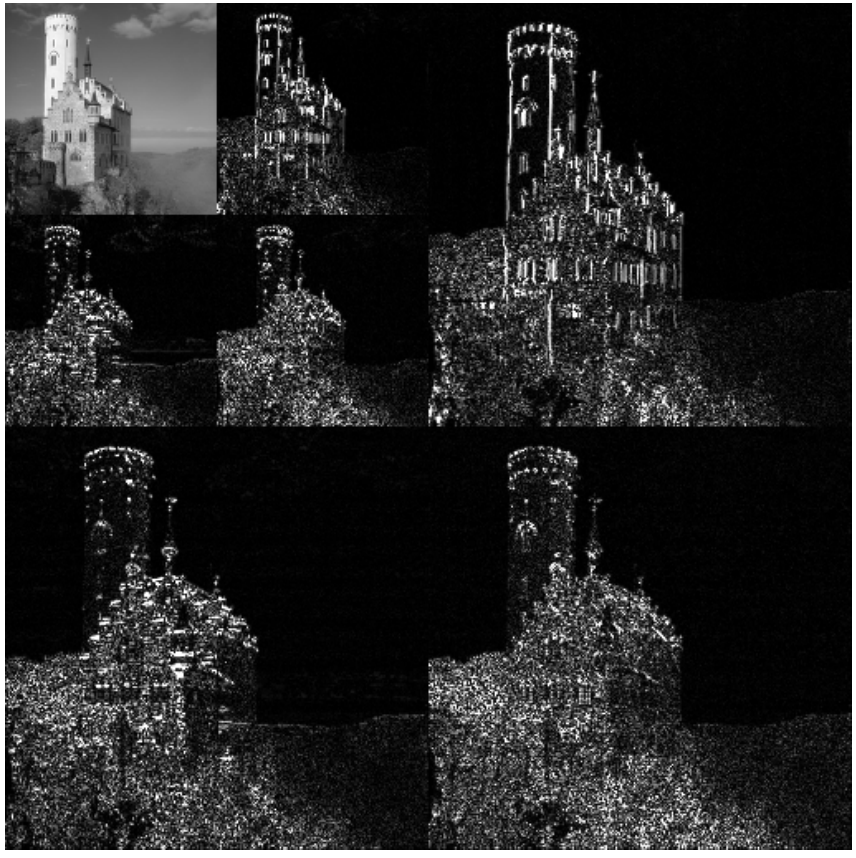


Figure: Wavelet coefficients at scale $a = 1$ (top left), differences to scale $a = 1/2$ (neighboring squares), and differences to scale $a = 1/4$ (neighboring squares).
From en.wikipedia.org/wiki/JPEG_2000

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Describe wavelet spaces and the wavelet transform.
- Check: Draw the locations of the group elements in the definition of wavelet frames.
- Check: These group elements accumulate near $a = 0$; why are they still relatively separated?
- Check: Verify that $m(a, b) := |a|^s$ is moderate for $w(a, b) := |a|^s + |a|^{-s}$.
- Background: Read up on wavelets and multi-resolution analysis.

Mathematics of Deep Learning, Summer Term 2020

Week 6, Video 6

Shearlet Group

Philipp Harms Lars Niemann

University of Freiburg

Structure

Notation: For $a \in \mathbb{R}^* := \mathbb{R} \setminus \{0\}$ and $b \in \mathbb{R}$, let

$$A_a = \begin{pmatrix} a & 0 \\ 0 & \operatorname{sign}(a)\sqrt{|a|} \end{pmatrix} \quad \text{and} \quad S_b = \begin{pmatrix} 1 & b \\ 0 & 1 \end{pmatrix}$$

denote the parabolic scaling matrix and the shear matrix, respectively.

Definition

The full shear group is the set $G := \mathbb{R}^* \times \mathbb{R} \times \mathbb{R}^2$ equipped with the product topology and the composition

$$(a_1, b_1, t_1) \cdot (a_2, b_2, t_2) := (a_1 a_2, b_1 + b_2\sqrt{|a_1|}, t_1 + S_{b_1} A_{a_1} t_2)\,.$$

Properties:

- The full shearlet group is not Abelian.
- The left Haar measure is given by $|a|^{-3}\, da\, db\, dt$.

Representation

Definition

The shearlet representation $\pi : G \to U(L^2(\mathbb{R}^2))$ is defined as

$$\pi(a,b,t)f(x) := |a|^{-\frac{3}{4}} f(A_a^{-1}S_b^{-1}(x-t)),$$

where $f \in L^2(\mathbb{R}^2)$, $(a,b,t) \in G$, and $x \in \mathbb{R}^2$.

Remark:

- It can be written in terms of translations and the left-regular representation of parabolic scaling and shear matrices:

$$\pi(a,b,t)f(y) = T_t L_{S_b A_a} f .$$

- The representation π is irreducible and square-integrable.
- However, as an aside, the representation of the reduced shear group $\mathbb{R}_+ \times \mathbb{R} \times \mathbb{R}^2$ is reducible.

Admissibility

Lemma

The Duflo–Moore operator associated to π is given by

$$Af(\xi,\eta) := \frac{\mathcal{F}f(\xi,\eta)}{|\xi|}, \qquad (\xi,\eta) \in \mathbb{R}^2,$$

and is defined for all f in

$$D(A) := \left\{ f \in L^2(\mathbb{R}^2) : \int_{\mathbb{R}^2} \frac{|\mathcal{F}f(\xi,\eta)|^2}{|\xi|^2} < \infty \right\}.$$

Remark: Thus, a function $\psi \in L^2(\mathbb{R}^2)$ is admissible if and only if

$$\int_{\mathbb{R}^2} \frac{|\mathcal{F}\psi(\xi,\eta)|^2}{|\xi|^2} d\xi\, d\eta = 1.$$

Shearlet Transform

Remark:

- Admissible vectors are called shearlets.
- The shearlet transform is the voice transform of the shearlet representation.

Definition

For any admissible $\psi \in L^2(\mathbb{R}^2)$, the shearlet transform $V_\psi \colon L^2(\mathbb{R}^2) \to L^2(G)$ is given by

$$V_\psi f(g) = \langle f, \pi_g f \rangle \, .$$

Remark: Generalizations to higher dimensions are possible.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Describe the shearlet group and its representation.
- Check: Draw the action of a shear matrix on a rectangle.
- Background: Skim through the computation of the Haar measure and the admissibility condition. Hint: this can be found in [Dahlke e.a. (2015), Lemma 3.27 and Proposition 3.30].

Mathematics of Deep Learning, Summer Term 2020

Week 6, Video 7

Shearlet Coorbit Spaces

Philipp Harms Lars Niemann

University of Freiburg

Analyzing Functions

Setting: We consider the representation of the shearlet group G on $L^2(\mathbb{R}^2)$.

Examples of analyzing functions:[1]

- Schwartz functions whose Fourier transform is compactly supported in $\mathbb{R}^2 \setminus (\{0\} \times \mathbb{R})$ are analyzing for every locally integrable weight function $w(a, b, t) = w(a, b)$.
- Compactly supported functions with sufficient smoothness and sufficiently many vanishing moments are analyzing for weight functions $w(a, b, t) = w(a) = |a|^r + |a|^{-r}$ with $r \in \mathbb{R}$.

[1] See [Dahlke e.a., Theorems 3.33 and 3.35]

Shearlet Coorbit Spaces

Definition

Let m be a w-moderate weight, and let ψ be an analyzing vector for w. For any $p \in [1, \infty]$, the shearlet coorbit space $H_{p,m}$ consists of all tempered distributions $f \in \mathcal{S}'$ such that

$$\int_G |\langle f, \pi_g \psi \rangle|^p \, m(g)^p dg < \infty \,,$$

with the usual modification for $p = \infty$.

Remark:

- This definition is independent of the choice of w and ψ.
- In the most important case $m(a, b, t) = |a|^{-s}$ with $s \in \mathbb{R}$, there are comparison results to Besov spaces.

Shearlet Frames

Theorem

Let $p \in [1, \infty)$, $s \in \mathbb{R}$, $w(a,b,t) = |a|^s + |a|^{-s}$, and $m(a,b,t) = |a|^{-s}$. For suitable[2] ψ and sufficiently small $\alpha > 1$, $\beta > 0$, and $\tau > 0$, the vectors

$$\left(\pi_g \psi : g = (\alpha^a, \alpha^{a/2}\beta b, S_{\alpha^{a/2}\beta b} A_{\alpha^a} \tau t)\right)_{a \in \mathbb{Z}, b \in \mathbb{Z}, t \in \mathbb{Z}}$$

form a Banach frame for $H_{p,m}$ with respect to the sequence space

$$\ell^p_m := \left\{ (\lambda_{a,b,t})_{a,b,t \in \mathbb{Z}} : \|\lambda\|^p_{\ell^p_m} := \sum_{a,b,t \in \mathbb{Z}} |\lambda_{a,b,t}|^p \alpha^{-asp} < \infty \right\}.$$

Proof:[2] For any given U and sufficiently small $\alpha > 1$, $\beta > 0$, and $\tau > 0$, the following group elements are U-dense and relatively separated:

$$(\epsilon \alpha^a, \alpha^{a/2}\beta b, S_{\alpha^{a/2}\beta b} A_{\alpha^a} \tau t)_{\epsilon \in \{-1,1\}, a \in \mathbb{Z}, b \in \mathbb{Z}, t \in \mathbb{Z}}$$ □

[2] See [Dahlke, Theorems 3.36 and 3.38].

Frequency Localization of Shearlet Frames

Remark: ψ is typically chosen as $\mathcal{F}\psi(\xi_1, \xi_2) = \mathcal{F}\psi_1(\xi_1)\,\mathcal{F}\psi_2(\xi_2/\xi_1)$ with $\operatorname{supp}\mathcal{F}\psi_1 \subseteq [-2, -1/2] \cup [1/2, 2]$ and $\operatorname{supp}\mathcal{F}\psi_2 \subseteq [-1, 1]$.

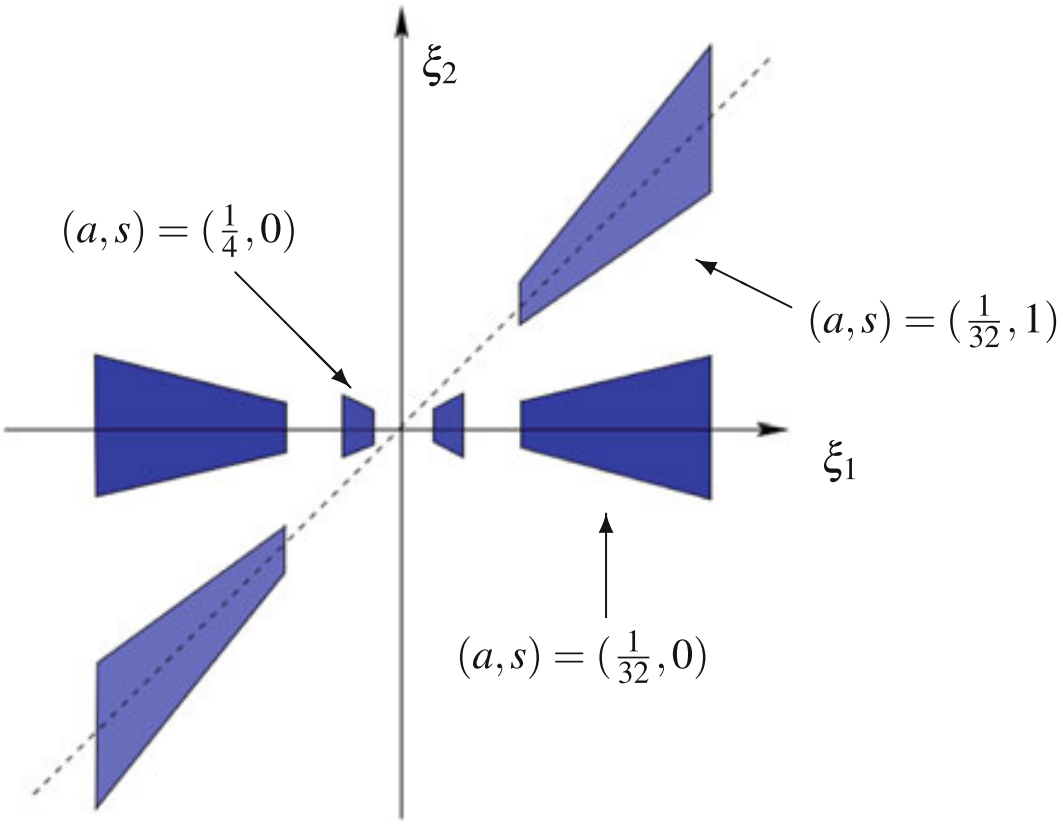


Figure: Support of ψ after scaling by a and shearing by $b := s$. [Dahlke e.a. (2015)]

Shearlet Frames for Edge Detection

Remark: The decay of $V_\psi f(a, b, t)$ for $a \searrow 0$ is

- Fast when t is a regular point of f, and
- Slow when t lies on an edge of f which is normal to $(1, b)$.

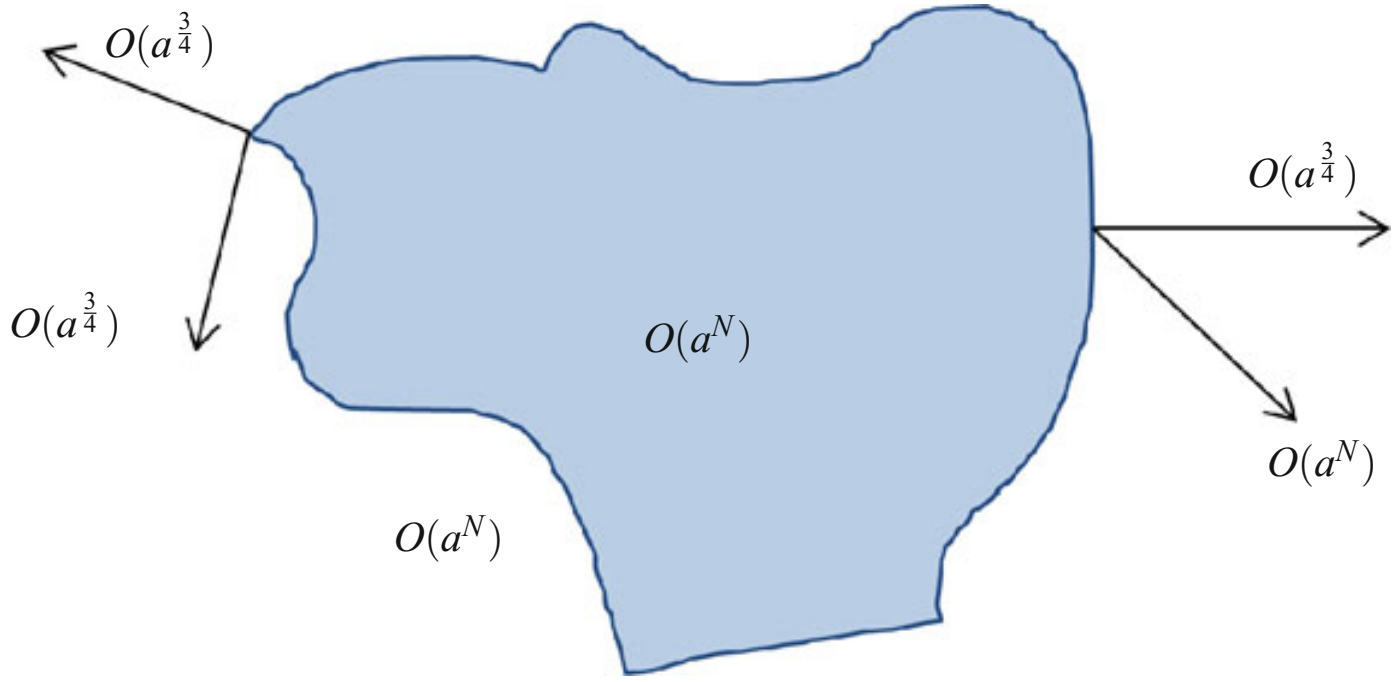


Figure: Indicator function f, points t with attached vectors $(1, b)$, and decay of $V_\psi f(a, b, t)$ for $a \searrow 0$. [Dahlke e.a., 2015]

Shearlet Frames for Edge Detection

Example: edge detection based on shearlet coefficients.

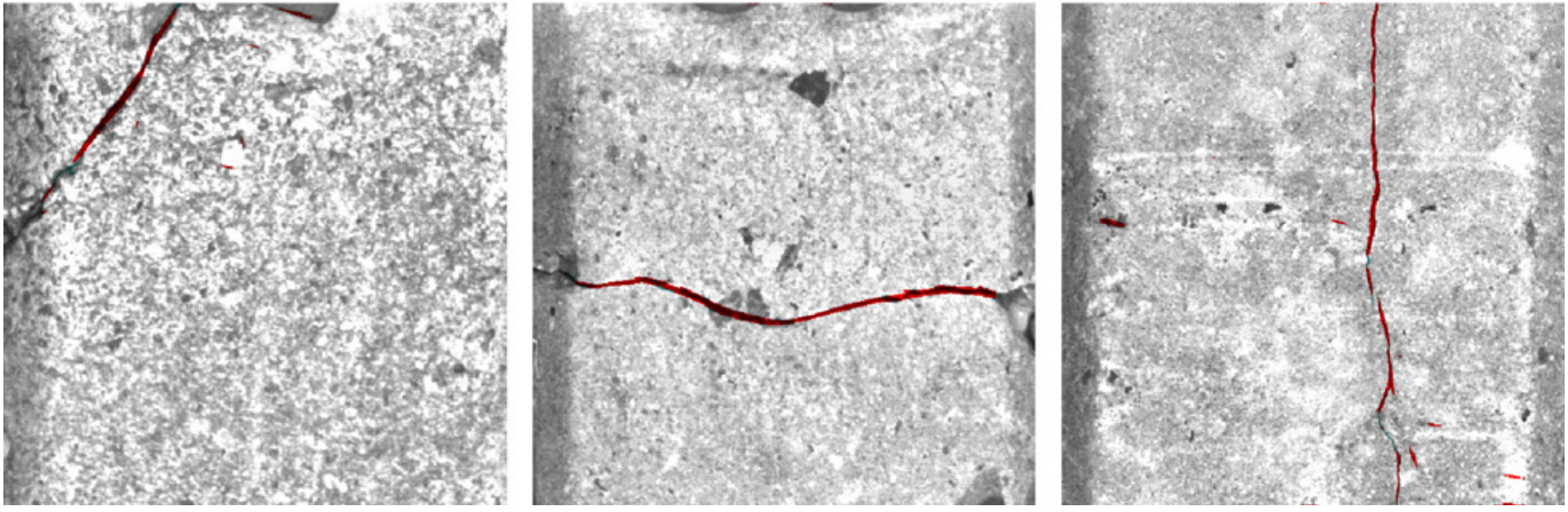


Figure: [Gibert (2014): Discrete Shearlet Transform on GPU with applications in anomaly detection and denoising]

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Describe the construction of shearlet coorbit spaces.
- Check: Draw the locations of the scaling and shearing coefficients of the shearlet frame.
- Discussion: How could one redefine shearlets to achieve symmetry with respect to the horizontal and vertical axes in $\mathbb{R}^2$? Hint: define horizontal and vertical shearlets.
- Discussion: Are shearlets directional wavelets? In what sense?
- Background: Find out about ridgelets and curvelets and compare them to shearlets.

Mathematics of Deep Learning, Summer Term 2020

Week 6, Video 8

Wrapup

Philipp Harms Lars Niemann

University of Freiburg

Outlook on this week's discussion and reading session

- Reading:
  - Gröchenig (2001): Foundations of Time-Frequency Analysis
  - Mallat (2009): A Wavelet Tour of Signal Processing
  - Kutyniok and Labate (2012): Shearlets - Multiscale Analysis for Multivariate Data

Summary by learning goals

Having heard this lecture, you can now ...

- Describe Schrödinger, wavelet, and shearlet representations and the associated modulation, wavelet, and shearlet spaces.
- Explain the time-frequency tilings of the associated signal transforms.
- Implement these signal transforms by neural networks.

Mathematics of Deep Learning, Summer Term 2020

Week 7

Sparse Data Representation

Philipp Harms Lars Niemann

University of Freiburg

Overview of Week 7

1. Rate-Distortion Theory
2. Hypercube Embeddings and Ball Coverings
3. Dictionaries as Encoders
4. Frames as Dictionaries
5. Networks as Encoders
6. Dictionaries as Networks
7. Wrapup

Acknowledgement of Sources

Sources for this lecture:

- Bölcskei, Grohs, Kutyniok, Petersen (2017): Optimal approximation with sparsely connected deep neural networks. In: SIAM Journal on Mathematics of Data Science 1.1, pp. 8–45
- Dahlke, De Mari, Grohs, Labatte (2015): Harmonic and Applied Analysis. Birkhäuser.

Mathematics of Deep Learning, Summer Term 2020

Week 7, Video 1

Rate-Distortion Theory

Philipp Harms Lars Niemann

University of Freiburg

Encoding, Decoding, and Distortion

Definition

Let $\mathcal{H}$ be a normed space, let $\mathcal{C} \subseteq \mathcal{H}$ be a signal class, and let $l \in \mathbb{N}$.

- The set of binary encoders of $\mathcal{C}$ with runlength l is defined as

$$\mathcal{E}^l := \{E : \mathcal{C} \to \{0,1\}^l\} .$$

- The set of binary decoders with runlength l is defined as

$$\mathcal{D}^l := \{D : \{0,1\}^l \to \mathcal{H}\} .$$

- The distortion of an encoder-decoder pair $(E, D) \in \mathcal{E}^l \times \mathcal{D}^l$ is defined as

$$\delta(E, D) := \sup_{f \in \mathcal{C}} \|f - D(E(f))\|_{\mathcal{H}} .$$

Remark: Alternatively, in probabilistic settings, one can consider the expected distortion $\mathbb{E}[\|f - D(E(f))\|_{\mathcal{H}}]$.

Encoding Rate

Definition

The optimal encoding rate of a signal class $\mathcal{C}$ in a normed space $\mathcal{H}$ is defined as

$$s^*_{\text{enc}}(\mathcal{C}) := \sup\left\{ s > 0 \middle| \inf_{(E,D)\in\mathcal{E}^l\times\mathcal{D}^l} \delta(E,D) = \mathcal{O}(l^{-s}) \right\}.$$

Remark:

- The optimal encoding rate quantifies the complexity of a signal class.
- The interpretation is information-theoretic: for any $s < s^*_{\text{enc}}(\mathcal{C})$, one can compress signals $f \in \mathcal{C}$ using l-bit encodings with distortion l^{-s}.
- Rate-distortion theory is the mathematical branch of information theory which studies data compression problems by analyzing the trade-off between compression rates and distortion.

Examples: Signal Classes

- Continuously differentiable functions:
  $\mathcal{C}_K^k(C) := \{f \in L^2(\mathbb{R}^d) \mid f \in C^k,\ \|f\|_{C^k} \leq K,\ \operatorname{supp} f \subseteq C\}$, where $C \subseteq \mathbb{R}^d$ is a smooth bounded domain.
- Piecewise continuously differentiable functions:
  $\mathcal{C}_K^{k,pw}(I) := \{f_1 \mathbb{1}_{[0,c)} + f_2 \mathbb{1}_{[c,1)} \mid c \in I,\ f_1, f_2 \in \mathcal{C}_K^k(I)\}$, where $I = (a, b)$ is an open interval.
- Star-shaped images:
  $\mathrm{STAR}_K^2 := \{\mathbb{1}_B \mid B$ is interior of Jordan curve $\rho \in C^2,\ \|\rho\|_{C^2} \leq K\}$.
- Cartoon images:
  $\mathrm{CART}_K^2 := \{f_1 \mathbb{1}_B + f_2 \mid \mathbb{1}_B \in \mathrm{STAR}_K^2,\ f_1, f_2 \in \mathcal{C}_K^2([0,1]^2)\}$.
- Textures: $\mathrm{TEXT}_{K,M}^k := \{\sin(Mf)g \mid f, g \in \mathcal{C}_K^k([0,1]^2)\}$.
- Mutilated functions: $\mathrm{MUTIL}_K^k := \{g(u\,\cdot)h \mid g \in \mathcal{C}_K^{k,pw}(\mathbb{R}),\ h \in \mathcal{C}_K^k([0,1]^d),\ u \in \mathbb{R}^d,\ \|u\| = 1\}$.

Remark: All introduced signal classes are relatively compact in $L^2(\mathbb{R}^d)$.

Examples: Optimal Encoding Rates

Remark: The main goal of this week's lecture is to establish the following optimal encoding rates and to show that they are achieved by deep neural networks.

Theorem

- $s^*_{\text{enc}}(\mathcal{C}^k_K(C)) = k/d.$
- $s^*_{\text{enc}}(\mathcal{C}^{k,pw}_K(I)) = k.$
- $s^*_{\text{enc}}(\text{STAR}^2_K) = 1.$
- $s^*_{\text{enc}}(\text{CART}^2_K) = 1.$
- $s^*_{\text{enc}}(\text{TEXT}^k_{K,M}) = k/2.$
- $s^*_{\text{enc}}(\text{MUTIL}^k_K) = k/d.$

Sketch of Proof:

- Upper bounds on encoding rates: Hypercubes are difficult to encode. If $\mathcal{C}$ contains hypercubes, then $\mathcal{C}$ is difficult to encode. See Video 2.
- Lower bounds on encoding rates: If signals in $\mathcal{C}$ have Banach frame coefficients with fast decay, then picking the n largest among the first n^k frame coefficients defines a good encoder. See Video 4. □

Paradigm: Analysis by Synthesis

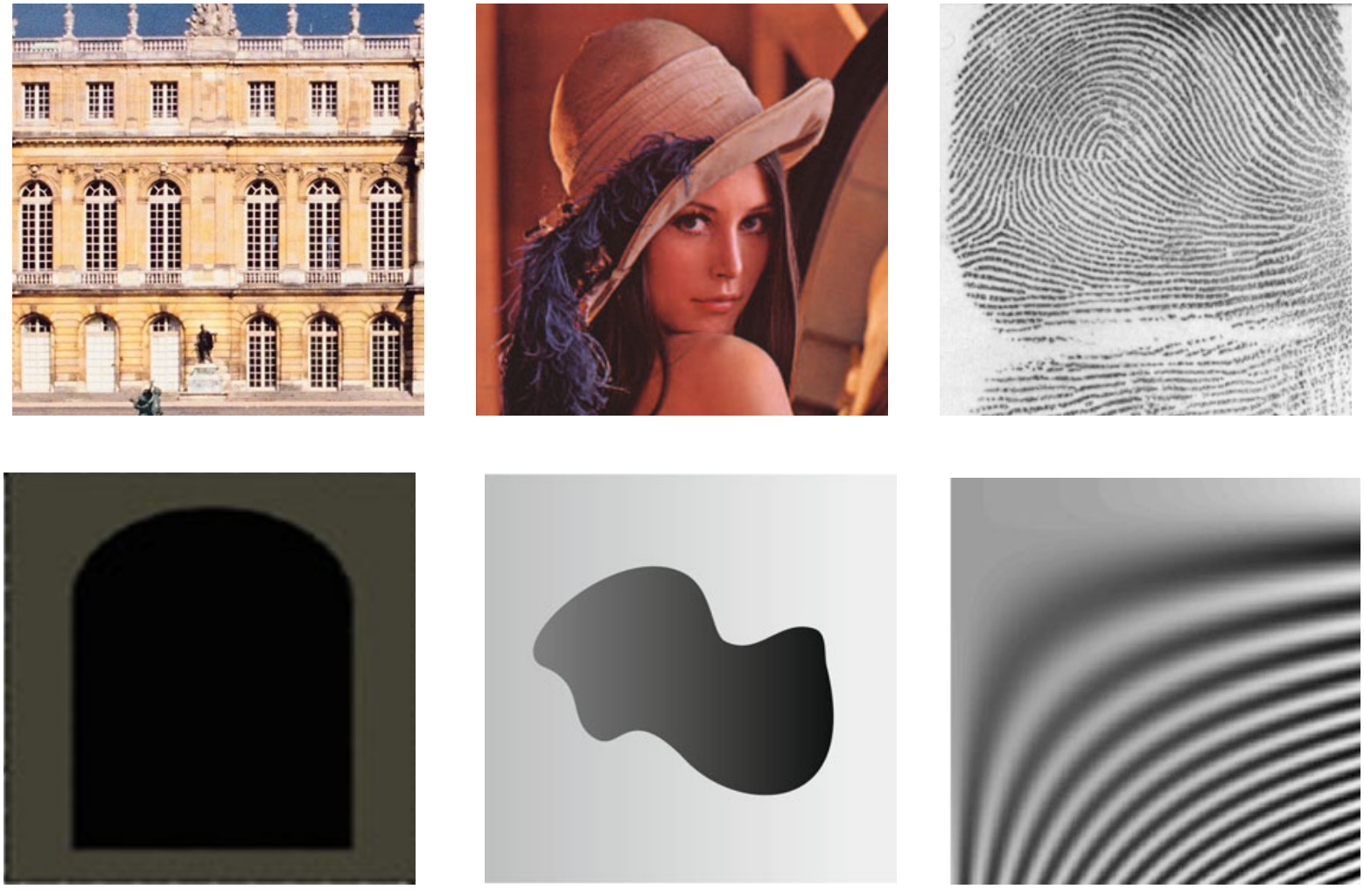


Figure: Real-world images (top) can be analyzed by synthesizing them from simpler image elements (bottom) such as star-shaped domains, cartoons, or textures. Additional benefits are compression and denoising. [Dahlke, Fig. 5.1–3]

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What is an endoding-decoding pair, and how are optimal encoding rates defined?
- Check: How many bits are needed to encode a natural number in $\{1, \ldots, n\}$?
- Background: The definition of star-shaped images involves Jordan curves—can you recall their definition and main properties?
- Context: Read some introductory articles (e.g. on Wikipedia) on data compression and rate-distortion theory.

Mathematics of Deep Learning, Summer Term 2020

Week 7, Video 2

Hypercube Embeddings and Ball Coverings

Philipp Harms Lars Niemann

University of Freiburg

Hypercube Embeddings

Definition (Donoho 2001)

Let $\mathcal{C}$ be a signal class in $\mathcal{H}$, and let $p > 0$.

- A hypercube of dimension $m \in \mathbb{N}$ and side-length $\delta > 0$ is a set of the form

$$\left\{ f + \sum_{i=1}^{m} \epsilon_i \psi_i \middle| \epsilon_i \in \{0, 1\} \right\},$$

  where $f \in \mathcal{C}$, and ψ_i are orthogonal functions in $\mathcal{H}$ with $\|\psi_i\|_{\mathcal{H}} \geq \delta$.
- The signal class $\mathcal{C}$ is said to contain a copy of ℓ_0^p if it contains for each $k \in \mathbb{N}$ a hypercube with dimension m_k and side-length δ_k such that

$$\delta_k \to 0 \quad \text{and} \quad m_k^{-1/p} = \mathcal{O}(\delta_k) \quad \text{as } k \to \infty.$$

Remark: A ball of radius r in ℓ^p contains hypercubes of dimension $m \in \mathbb{N}$ with side-length $rm^{-1/p}$.

Hypercube Embeddings and Encoding Rates

Remark: For many signal classes, hypercube embeddings are easy to construct and provide (sharp) upper bounds on the encoding rate.

Theorem

If a signal class $\mathcal{C}$ in $\mathcal{H}$ contains a copy of ℓ_0^p for some $p \in (0, 2]$, then

$$s^*_{\mathrm{enc}}(\mathcal{C}) \leq \frac{1}{p} - \frac{1}{2} \, .$$

Proof: Hypercube Embeddings and Encoding Rates

Idea of proof: (See [Dahlke e.a., Theorem 5.12] for a full proof.)

- Hypercubes of dimension m can be identified with bit streams in $\{0,1\}^m$.
- Recall that the Hamming distance (aka. ℓ^1 or Manhattan distance) between two bit streams is the number of unequal bits.
- Chernoff's bounds imply that for any compression rate $\alpha \in (0,1)$, there exists $C > 0$ such that for any $m \in \mathbb{N}$ and encoder-decoder

$$E\colon \{0,1\}^m \to \{0,1\}^{\lfloor \alpha m \rfloor}, \qquad D\colon \{0,1\}^{\lfloor \alpha m \rfloor} \to \{0,1\}^m,$$

  the distortion in the Hamming distance is lower-bounded by Cm.
- This translates into a lower bound on the encoding rate of a hypercube as well as its containing signal class. □

Examples: Upper Bounds on Optimal Encoding Rates

Remark: The following are special cases of the above theorem.

Corollary

The following upper bounds on encoding rates are achieved via hypercube embeddings:

- $s^*_{\text{enc}}(\mathcal{C}^k_K(C)) \leq k/d$ *via embedding of* $\ell_0^{1/(\frac{k}{d}+\frac{1}{2})}$
- $s^*_{\text{enc}}(\mathcal{C}^{k,pw}_K(I)) \leq k$ *via embedding of* $\ell_0^{1/(k+\frac{1}{2})}$
- $s^*_{\text{enc}}(\text{STAR}^2_K) \leq 1$ *via embedding of* $\ell_0^{2/3}$
- $s^*_{\text{enc}}(\text{CART}^2_K) \leq 1$ *via embedding of* $\ell_0^{2/3}$
- $s^*_{\text{enc}}(\text{TEXT}^k_{K,M}) \leq k/2$ *via embedding of* $\ell_0^{2/(k+1)}$
- $s^*_{\text{enc}}(\text{MUTIL}^k_K) \leq k/d$ *via embedding of* $\ell_0^{1/(\frac{k}{d}+\frac{1}{2})}$

Examples: Upper Bounds on Optimal Encoding Rates

Idea of proof: For a fixed bump function ψ, one uses hypercubes of the following forms:

- $\sum_{i=0}^{n-1} \epsilon_i \psi(nx - i)$ for piece-wise continuously differentiable functions,
- $\mathbb{1}_{\{\|x\| \leq 1\}} + \sum_{i=0}^{n-1} \epsilon_i \left(\mathbb{1}_{\{\|x\| \leq i/n\}} - \mathbb{1}_{\{\|x\| \leq 1\}} \right)$ for star-shaped images, or
- $\sum_{i,j=1}^{n-1} \epsilon_{i,j} \sin\left(n^{-k} \psi(nx - i) \psi(ny - j) \right)$ for textures, etc.

See [Dahlke e.a., Theorem 5.17] for a full proof. □

Digression: Kolmogorov Entropy

Remark:

- Encoding rates are closely related to covering numbers and Kolmogorov entropy.
- We have already encountered the Kolmogorov entropy in the context of statistical learning theory.
- Unfortunately, covering numbers are often difficult to compute and therefore of rather theoretical interest.

Definition

Let $\mathcal{H}$ be a metric space, and let $\mathcal{C} \subseteq \mathcal{H}$ be a relatively compact subset.

- The covering number of $\mathcal{C}$ is defined for any $\epsilon > 0$ as the smallest number $N_\epsilon(\mathcal{C})$ of ϵ-balls required to cover $\mathcal{C}$.
- The Kolmogorov entropy of $\mathcal{C}$ is defined as $H_\epsilon(\mathcal{C}) := \log_2(N_\epsilon(\mathcal{C}))$.

Digression: Kolmogorov Entropy and Encoding Rates

Lemma

Let $\mathcal{C} \subseteq \mathcal{H}$ be a relatively compact signal class in a normed space $\mathcal{H}$. Then the optimal encoding rate $s^*_{\text{enc}}(\mathcal{C})$ is related to the Kolmogorov entropy $H_\epsilon(\mathcal{C})$ by

$$s^*_{\text{enc}}(\mathcal{C}) = \sup\left\{s > 0 : H_\epsilon(\mathcal{C}) = \mathcal{O}(\epsilon^{-\frac{1}{s}})\right\}.$$

Proof:

- Given a pair (E, D) of length l that achieves distortion ϵ, the ϵ-balls centered at $D(\xi)$, $\xi \in \{0,1\}^l$, cover $\mathcal{C}$.
- Conversely, given $\epsilon > 0$, we can find $N_\epsilon := 2^{H_\epsilon(\mathcal{C})}$ centers whose ϵ-neighborhoods cover $\mathcal{C}$. Encode $\mathcal{C}$ using the binary representation of the nearest center, and decode by reversing this process. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How are upper bounds on the encoding rate obtained from hypercube embeddings?
- Check: Show that relatively compact signal classes have finite covering numbers.
- Background: Skim through the construction of hypercube embeddings for specific signal classes in [Dahlke e.a., Theorem 5.17].
- Transfer: The upper bounds on the optimal encoding rates decay inversely proportional to the dimension—an instance of the curse of dimensionality.

Mathematics of Deep Learning, Summer Term 2020

Week 7, Video 3

Dictionaries as Encoders

Philipp Harms Lars Niemann

University of Freiburg

Repetition: Approximation Rates of Dictionaries

Definition

A dictionary $(\phi_\lambda)_{\lambda\in\Lambda}$ in $\mathcal{H}$ achieves an approximation rate of $(h_n)_{n\in\mathbb{N}}$ if

$$\sigma(\Sigma_n(\phi),\mathcal{C}) := \sup_{f\in\mathcal{C}} \inf_{g\in\Sigma_n(\phi)} \|f-g\|_{\mathcal{H}} = \mathcal{O}(h_n) \quad \text{as } n\to\infty,$$

where $\Sigma_n(\phi)$ denotes the set of n-term linear combinations in ϕ.

Remark:

- A dense dictionary ϕ in $\mathcal{H}$ achieves any approximation rate for any signal class. Nevertheless, it is ill-suited for efficient encoding of functions.
- This motivates the requirement of polynomial-depth search, which is described next.
- We restrict ourselves to polynomial rates $h_n = n^{-s}$, $s>0$, as these are most relevant.

Dictionary Approximation with Polynomial-Depth Search

Definition (Donoho 2001)

Let $\phi = (\phi_i)_{i\in\mathbb{N}}$ be a dictionary, π a univariate polynomial, $\mathcal{C}$ a signal class in $\mathcal{H}$, and $n \in \mathbb{N}$.

- The set of n-term linear combinations in ϕ with polynomial-depth search is defined as

$$\Sigma_n^{\pi}(\phi) = \left\{ \sum_{i=1}^{\pi(n)} c_i \phi_i \middle| c_i \in \mathbb{R} \text{ with } \|c\|_0 \leq n \right\} .$$

- The approximation rate of ϕ with polynomial-depth search is defined as

$$s^*_{\text{dict}}(\mathcal{C}, \phi) := \sup \left\{ s > 0 \middle| \exists \pi : \sup_{f\in\mathcal{C}} \inf_{g\in\Sigma_n^{\pi}(\phi)} \|g - f\|_{\mathcal{H}} = \mathcal{O}(n^{-s}) \right\}$$

Remark: Here, the dictionary needs to be ordered, i.e., indexed over $\mathbb{N}$.

Encoding via Dictionaries

Remark: Polynomial-depth search leads to the desired link between dictionary approximation rates and encoding rates:

Theorem

For any dictionary ϕ and bounded signal class $\mathcal{C}$ in $\mathcal{H}$,

$$s^*_{\mathrm{enc}}(\mathcal{C}) \geq s^*_{\mathrm{dict}}(\mathcal{C}, \phi) \ .$$

Remark:

- A dictionary ϕ is called rate-optimal if equality holds above.
- Explicit dictionary approximation rates can be obtained for Hilbert or Banach frames, as shown in the next video.

Proof: Encoding via Dictionaries

Proof:

- We start by constructing an encoder. For any $s < s^*_{\mathrm{dict}}(\mathcal{C}, \phi)$, there exists a polynomial π and a constant $C > 0$ such that for all $n \in \mathbb{N}$ and $f \in \mathcal{C}$, there exist coefficients $c_i \in \mathbb{R}$ with $\|c\|_0 \leq n$ such that

$$\left\| f - \sum_{i=1}^{\pi(n)} c_i \phi_i \right\|_{\mathcal{H}} \leq C n^{-s} .$$

- The set $\Lambda_n := \{i \in \mathbb{N} : c_i \neq 0\}$ can be encoded using $\mathcal{O}(n \log n)$ bits thanks to the assumption of polynomial-depth search.
- Applying the Gram-Schmidt orthonormalization to $\phi_{\Lambda_n} := (\phi_\lambda)_{\lambda \in \Lambda_n}$ yields an orthonormal set $\tilde{\phi}_{\Lambda_n} := (\tilde{\phi}_\lambda)_{\lambda \in \Lambda_n}$. Some $\tilde{\phi}_\lambda$ may be zero.

Proof: Encoding via Dictionaries (cont.)

- Determine coefficients $\tilde{c}_\lambda$ uniquely by

$$\sum_{\lambda\in\Lambda_n} \tilde{c}_\lambda \tilde{\phi}_\lambda = \sum_{\lambda\in\Lambda_n} c_\lambda \phi_\lambda, \qquad \tilde{c}_\lambda = 0 \text{ if } \tilde{\phi}_\lambda = 0.$$

- Note that

$$\left\| f - \sum_{\lambda\in\Lambda_n} \tilde{c}_\lambda \tilde{\phi}_\lambda \right\|_{\mathcal{H}} \leq C n^{-s}$$

  and that the sequence $\tilde{c}$ is ℓ^2-bounded uniformly in n and f. (Here enters the boundedness of $\mathcal{C}$.)

- Rounding the coefficients $\tilde{c}_\lambda$ up to multiples of $n^{-(s+\frac{1}{2})}$ encodes them with a bit string of length $\mathcal{O}(n \log n)$.
- Altogether, this gives an encoding procedure $E_l : \mathcal{C} \to \{0,1\}^l$ with length $l = \mathcal{O}(n \log n)$.

Proof: Decoding via Dictionaries

- Decoding is done by reversing this process: starting from a bit string ξ, reconstruct the set Λ_n and the rounded approximations $\hat{c}_\lambda$ of $\tilde{c}_\lambda$, and define the decoder

$$D_n \colon \{0,1\}^l \to \mathcal{H}, \qquad D_l(\xi) := \sum_{\lambda \in \Lambda_n} \hat{c}_\lambda \tilde{\phi}_\lambda \,.$$

- It remains to control the distortion:

$$\begin{aligned}
\|f - D_l(E_l(f))\|_{\mathcal{H}} &= \left\| f - \sum_{\lambda \in \Lambda_n} \hat{c}_\lambda \tilde{\phi}_\lambda \right\|_{\mathcal{H}} \\
&\leq \left\| f - \sum_{\lambda \in \Lambda_n} \tilde{c}_\lambda \tilde{\phi}_\lambda \right\|_{\mathcal{H}} + \left\| \sum_{\lambda \in \Lambda_n} \hat{c}_\lambda \tilde{\phi}_\lambda - \sum_{\lambda \in \Lambda_n} \tilde{c}_\lambda \tilde{\phi}_\lambda \right\|_{\mathcal{H}} \\
&\leq C n^{-s} + \max_{\lambda \in \Lambda_n} |\tilde{c}_\lambda - \hat{c}_\lambda| \, n^{\frac{1}{2}} \leq C n^{-s}.
\end{aligned}$$

□

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How are lower bounds on encoding rates obtained from dictionary approximation rates?
- Check: The approximation rate of a dense dictionary is arbitrarily high—what about the approximation rate with polynomial-depth search?
- Check: Verify that the coefficients $\tilde{c}$ after Gram–Schmidt orthogonalization are ℓ^2-bounded uniformly in $n \in \mathbb{N}$ and $f \in \mathcal{C}$. Hint: $\|\tilde{c}\|_{\ell^2} = \|\sum_\lambda \tilde{c}_\lambda \tilde{\phi}_\lambda\|_{\mathcal{H}}$.
- Transfer: Nonlinear approximation spaces $\mathcal{C}$ are *defined* by the requirement that $s^*(\mathcal{C}, \phi) = s$ for given $s \in \mathbb{R}$.

Mathematics of Deep Learning, Summer Term 2020

Week 7, Video 4

Frames as Dictionaries

Philipp Harms Lars Niemann

University of Freiburg

Repetition: Hilbert Frames

Remark: Recall that Hilbert frames are Banach frames in Hilbert spaces with respect to the sequence space ℓ^2; this boils down to the following:

Definition

- A Hilbert frame in a Hilbert space $\mathcal{H}$ is a dictionary $\phi = (\phi_\lambda)_{\lambda\in\Lambda}$ s.t.

$$\forall f \in \mathcal{H}: \qquad \|f\|_H^2 \lesssim \sum_{\lambda\in\Lambda} |\langle f, \phi_\lambda\rangle_{\mathcal{H}}|^2 \lesssim \|f\|_{\mathcal{H}}^2.$$

- A dual frame for ϕ is a complementary dictionary $\tilde{\phi} = (\tilde{\phi}_\lambda)_{\lambda\in\Lambda}$ s.t.

$$\forall f \in \mathcal{H}: \qquad f = \sum_{\lambda\in\Lambda} \langle f, \tilde{\phi}_\lambda\rangle_{\mathcal{H}} \phi_\lambda = \sum_{\lambda\in\Lambda} \langle f, \phi_\lambda\rangle_{\mathcal{H}} \tilde{\phi}_\lambda.$$

Remark: Every Hilbert frame has a dual frame, for instance the canonical one, which is determined by $\phi_\mu = \sum_\lambda \langle \tilde{\phi}_\mu, \phi_\lambda\rangle_{\mathcal{H}} \phi_\lambda$, or the one from the definition of Banach frames.

Weak ℓ^p Spaces

Remark: Recall that a quasi-norm is a norm without a triangle inequality.

Definition

The weak ℓ^p-quasinorm of a sequence $c := (c_k)_{k\in\mathbb{N}}$ is defined for any $p > 0$ as

$$\|c\|_{w\ell^p}^p := \sup_{t>0} t^p \ \#\{k \in \mathbb{N} : |c_k| > t\},$$

and the space $w\ell^p$ consists of all sequences with finite weak ℓ^p-quasinorm.

Remark:

- For any $p \geq 1$, the space ℓ^p embeds continuously in $w\ell^p$ because

$$\|c\|_{\ell^p}^p \geq \sum_k t^p \mathbb{1}_{\{k:|c_k|>t\}} + \sum_k |c_k|^p \mathbb{1}_{\{k:|c_k|\leq t\}} \geq t^p \#\{k : |c_k| > t\}.$$

- The space $w\ell^p$ coincides with the Lorentz space $\ell^{p,\infty}$, is complete, and is normable for $p > 1$. Weak L^p spaces are defined similarly.

Approximation via Frames

Remark: We next show that weak ℓ^p bounds on Hilbert frame coefficients translate into dictionary approximation rates.

Theorem

Let $(\phi_n)_{n\in\mathbb{N}}$ be a Hilbert frame with dual frame $(\tilde{\phi}_n)_{n\in\mathbb{N}}$ in a Hilbert space $\mathcal{H}$, and let $\mathcal{C}$ be a signal class in $\mathcal{H}$ which satisfies the weak ℓ^p bound

$$\sup_{f\in\mathcal{C}} \left\| (\langle f, \tilde{\phi}_n \rangle_{\mathcal{H}})_{n\in\mathbb{N}} \right\|_{w\ell^p} < \infty$$

and, for some $\alpha > 0$, the ℓ^2 tail bound

$$\sup_{f\in\mathcal{C}} \sum_{i\geq n} |\langle f, \tilde{\phi}_i \rangle|^2 = \mathcal{O}(n^{-\alpha}).$$

Then $s^*_{\mathrm{dict}}(\mathcal{C}, \phi) \geq \frac{1}{p} - \frac{1}{2}$.

Proof: Approximation via Frames

Proof: Claim 1: The $w\ell^p$ bound implies that $\sigma(\Sigma_n(\phi), \mathcal{C}) = \mathcal{O}(n^{-s})$.

- For any signal $f \in \mathcal{C}$, picking the n largest frame coefficients defines an n-term approximation

$$f_n := \sum_{i \leq n} c_{k_i} \phi_{k_i} \,,$$

  were c_{k_i} is a non-increasing rearrangement of $c_k := \langle f, \tilde{\phi}_k \rangle_{\mathcal{H}}$.
- The definition of the $w\ell^p$ norm implies $|c_{k_i}| \lesssim i^{-1/p}$ because

$$|c_{k_i}|^p \, i \leq |c_{k_i}|^p \, \#\{k \in \mathbb{N} : |c_k| \geq |c_{k_i}|\} \leq \|c\|_{w\ell^p}^p \,.$$

- Together with the frame property of ϕ this yields

$$\|f - f_n\|^2 \lesssim \sum_{i>n} |c_{k_i}|^2 \lesssim \sum_{i>n} i^{-2/p} \leq n^{-2s}, \quad \text{where } s := \tfrac{1}{p} - \tfrac{1}{2} \,,$$

  where the last inequality follows from an elementary calculation. This proves Claim 1.

Proof: Approximation via Frames

Claim 2: The ℓ^2 tail bound implies $\sigma(\Sigma_n^\pi(\phi), \mathcal{C}) = \mathcal{O}(n^{-s})$ for suitable π.

- Define $\pi(n) := n^{\lceil 2s/\alpha \rceil}$.
- For any signal $f \in \mathcal{C}$, picking the first $\pi(n)$ frame coefficients defines an approximation $\tilde{f}_n$ with

$$\|f - \tilde{f}_n\|_{\mathcal{H}}^2 \lesssim \sum_{i > \pi(n)} \left|\langle f, \tilde{\phi}_i \rangle_{\mathcal{H}}\right|^2 \leq \big(\pi(n)\big)^{-\alpha} \leq n^{-2s}.$$

- By the previous claim, picking the n largest frame coefficients of $\tilde{f}_n$ defines an approximation f_n with

$$\|\tilde{f}_n - f_n\|_{\mathcal{H}}^2 \lesssim n^{-2s}.$$

- Taken together, this implies

$$\|f - f_n\|_{\mathcal{H}} \lesssim n^{-s},$$

which proves Claim 2 and establishes the theorem. □

Examples: Lower Bounds on Optimal Encoding Rates

Remark: The following lower bounds are sharp and are obtained as special cases of the previous theorem:

Corollary

The following lower bounds on encoding rates are achieved via frames:

- $s^*_{\mathrm{enc}}(\mathcal{C}^k_K(C)) \geq k/d$ *via wavelets, shearlets, and many more*
- $s^*_{\mathrm{enc}}(\mathcal{C}^{k,pw}_K(I)) \geq k$ *via wavelets*
- $s^*_{\mathrm{enc}}(\mathrm{STAR}^2_K) \geq 1$ *via curvelets and shearlets*
- $s^*_{\mathrm{enc}}(\mathrm{CART}^2_K) \geq 1$ *via curvelets and shearlets*
- $s^*_{\mathrm{enc}}(\mathrm{TEXT}^k_{K,M}) \geq k/2$ *via wave atoms*
- $s^*_{\mathrm{enc}}(\mathrm{MUTIL}^k_K) \geq k/d$ *via ridgelets*

Proof: Verify the conditions of the previous theorem for the specified frames; see [Dahlke e.a., Theorem 5.51]. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How are dictionary approximation rates obtained from weak ℓ^p bounds on Hilbert frame coefficients?
- Background: Find the definition of wave atoms and have a look at some pictures of wave atoms. Hint: [Demanet and Ying (2007): Wave atoms and sparsity of oscillatory patterns]
- Discussion: Are the encoders/decoders obtained via frame approximations constructive and numerically implementable?
- Discussion: How could the theory be generalized to Banach frames, and what kind of results would you expect from this?

Mathematics of Deep Learning, Summer Term 2020

Week 7, Video 5

Networks as Encoders

Philipp Harms Lars Niemann

University of Freiburg

Neural Network Approximation Rates

Remark: Neural networks with constrained memory can be seen as encoders.

Definition

Let $\mathcal{C}$ be a signal class in a normed function space $\mathcal{H}$ on $\mathbb{R}^d$, let $M \in \mathbb{N}$, let π be a univariate polynomial, and let A be a subset of $\mathbb{R}$.

- The set $\mathcal{NN}_M^A$ of neural networks with quantized weights is defined as the set of neural networks Φ with input dimension d, output dimension 1, and at most M non-zero weights belonging to A.
- The effective network approximation rate of $\mathcal{C}$ is defined as

$$s^*_{\mathcal{NN}}(\mathcal{C}) := \sup \Big\{ s > 0 \Big| \exists \pi, \exists (A_M)_{M\in\mathbb{N}} : \#A_M = \mathcal{O}(\pi(M)), \sup_{f\in\mathcal{C}} \inf_{\Phi\in\mathcal{NN}_M^{A_M}} \|\mathrm{R}(\Phi) - f\|_{\mathcal{H}} = \mathcal{O}(M^{-s}) \Big\},$$

where R is defined using some fixed activation function $\rho \in C(\mathbb{R})$.

Encoding via Neural Networks

Remark: The memory constraint imposed via weight quantization yields the desired link between network approximation rates and encoding rates:

Theorem

For any signal class $\mathcal{C}$,

$$s^*_{\text{enc}}(\mathcal{C}) \geq s^*_{\mathcal{NN}}(\mathcal{C}).$$

Remark:

- Neural networks are called rate-optimal for $\mathcal{C}$ if equality holds above.
- The theorem implies a lower bound on the network connectivity, namely, an approximation error of ϵ requires approximately $\epsilon^{1/s^*_{\text{enc}}(\mathcal{C})}$ non-zero network weights.

Proof: Encoding via Neural Networks

Proof:

- Let $s < s^*_{\mathcal{NN}}(\mathcal{C})$, and choose π, $(A_M)_{M\in\mathbb{N}}$, and C such that

$$\forall M \in \mathbb{N}: \quad \sup_{f\in\mathcal{C}} \inf_{\Phi\in\mathcal{NN}_M^{A_M}} \|\mathrm{R}(\Phi) - f\|_{\mathcal{H}} < CM^{-s}, \quad \#A_M \leq \pi(M)\,.$$

- Thus, for any given $f \in \mathcal{C}$ and $M \in \mathbb{N}$, there exists a network $\Phi \in \mathcal{NN}_M^{A_M}$ with $\|\mathrm{R}(\Phi) - f\|_{\mathcal{H}} < CM^{-s}$.
- We write $E \leq M$ for the number of edges, $L \leq M$ for the number of layers, $N_0 := d$ for the input dimension, $N_1, \ldots, N_L$ for the numbers of neurons per layer, and $N := \sum_{\ell=0}^{L} N_\ell \leq 2E$.
- We will show that Φ can be encoded in a bit string of length $\mathcal{O}(M \log M)$. This yields an encoder-decoder pair with distortion

$$\|D(E(F)) - f\| = \|\mathrm{R}(\Phi) - f\| = \mathcal{O}(M^{-s})$$

thereby establishing the theorem.

Proof: Encoding via Neural Networks (cont.)

- We encode the architecture of Φ in a bit string:
  - The number E of edges is encoded by a string of E 1's, followed by a single 0.
  - The number L of layers is encoded by a string of $\lceil \log_2 E \rceil$ bits, namely, by the binary representation of $L-1$ with left-padded zeros.
  - Then $(N_0, \ldots, N_L)$ is encoded in a string of $(L+1)\lceil \log_2 E + 1 \rceil$ bits.
- We encode the topology of Φ in a bit string:
  - To each neuron, we associate a unique index $i \in \{1, \ldots, N\}$, noting that this index can be encoded in a string b_i of $\lceil \log_2 E \rceil + 1$ bits.
  - For each neuron i, we output the concatenation of the bit strings b_j of all children j, followed by a zero string of length $2\lceil \log_2 E \rceil + 2$ to signal the transition to neuron $i+1$.
- We encode the weights of Φ in a bit string:
  - Each weight requires $\lceil \log_2 \pi(M) \rceil$ bits.
  - The nodal weights are encoded in $(N_1 + \cdots + N_l)\lceil \log_2 \pi(M) \rceil$ bits.
  - The edge weights are encoded in $E\lceil \log_2 \pi(M) \rceil$ bits.
- Overall, this requires $\mathcal{O}(M \log_2 M)$ bits, as claimed. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What is the effective network approximation rate, and why is it upper-bounded by the encoding rate?
- Check: Why can the logarithmic factors in the rate computations be ignored?
- Check: In the last proof we constructed an encoder—what does the corresponding decoder look like?
- Discussion: What does the result say about deep learning? What are limitations of the result?

Mathematics of Deep Learning, Summer Term 2020

Week 7, Video 6

Dictionaries as Networks

Philipp Harms Lars Niemann

University of Freiburg

Representation of Dictionaries by Neural Networks

Setting: $\mathcal{H} = L^2(\Omega)$ for some $\Omega \subseteq \mathbb{R}^d$, and $\rho\colon \mathbb{R} \to \mathbb{R}$ is globally Lipschitz continuous or differentiable with polynomially bounded first derivative.

Definition

A dictionary $\phi = (\phi_i)_{i\in\mathbb{N}}$ in $\mathcal{H}$ is said to be effectively representable by neural networks if there exists $L, M \in \mathbb{N}$ and a bi-variate polynomial π such that for every $\epsilon \in (0, 1/2)$ and $i \in \mathbb{N}$ there exists a neural network Φ with $\mathrm{M}(\Phi) \leq M$, $\mathrm{L}(\Phi) \leq L$, and weights bounded by $\pi(i, \epsilon^{-1})$, such that

$$\|\phi_i - \mathrm{R}(\Phi)\|_{\mathcal{H}} \leq \epsilon.$$

Remark:

- The crucial point, also compared to our former setting for dictionary learning, is the requirement of polynomially bounded weights.
- For affine systems, i.e., dictionaries of affine transformations of a mother function ψ, it suffices to check effective representability of ψ.

Quantization of Neural Networks

Remark: We will need a seemingly stronger property, namely effective representation by quantized networks:

Lemma

In the definition of effective representability, it can be assumed without loss of generality that the weights of Φ are quantized in the sense that they belong to the set

$$\pi(i,\epsilon)\mathbb{Z} \cap [-\pi(i,\epsilon^{-1}), \pi(i,\epsilon^{-1})].$$

Proof: Quantization of Neural Networks

Sketch of proof for Lipschitz activation functions ρ:

- For single-layer networks $x \mapsto A_1 x + b_1$, which by definition are just affine maps, the quantization error of the network is proportional to the quantization error of the weights.
- For double-layer networks $x \mapsto A_2 \rho(A_1 x + b_1) + b_2$, the quantization error of the single-layer sub-network is amplified polynomially via the multiplication by A_2.
- By induction, the same holds for multi-layer networks.
- Thus, the quantization error of the network is $\mathcal{O}(\epsilon)$ if the quantization error of the weights is $\mathcal{O}(\epsilon^k)$ for sufficiently high k, with additional polynomial dependence on i.

For activation functions with polynomially bounded first derivative we refer to [Bölcskei e.a., Lemma 3.3]. □

Transfer of Approximation

Remark: Approximation rates for dictionaries transfer to approximation rates for neural networks if the dictionary is effectively represented by neural networks.

Theorem

If ϕ is effectively representable by neural networks and $\mathcal{C}$ is bounded, then

$$s^*_{\mathcal{NN}}(\mathcal{C}) \geq s^*_{\mathrm{dict}}(\mathcal{C}, \phi).$$

Proof: Transfer of Approximation

Proof: Dictionary learning.

- For any $s < s^*_{\text{dict}}(\mathcal{C}, \phi)$, there are approximations f_n of $f \in \mathcal{C}$ s.t.

$$f_n := D_n(E_n(f)) := \sum_{i=1}^{\pi(n)} c_i \phi_i, \qquad \|f_n - f\|_{\mathcal{H}} = \mathcal{O}(n^{-s}).$$

- In the theorem on encoding via dictionaries in Video 3 we have shown that the coefficients c_i can be chosen in a set of cardinality polynomially bounded in n.
- The dictionary functions ϕ_i, $i \in \{1, \dots, \pi(n)\}$, can be effectively represented by neural networks Φ_i, up to an approximation error of order $\mathcal{O}(n^{-s})$, with weights polynomially bounded in n.
- By the quantization lemma, it can be assumed without loss of generality that the weights of the networks Φ_i belong to a set of cardinality polynomially bounded in n.
- Taking linear combinations produces a network approximation of f_n with weights in a set of cardinality polynomially bounded in n and approximation error $\mathcal{O}(n^{-s})$.

□

Rate-Optimal Approximation by Neural Networks

Corollary

If ϕ is a rate-optimal dictionary for $\mathcal{C}$, and ϕ is effectively represented by neural networks, then neural networks are rate-optimal for $\mathcal{C}$.

Proof: The following rates are equal,

$$s^*_{\text{dict}}(\mathcal{C}, \phi) \overset{1}{=} s^*_{\text{enc}}(\mathcal{C}) \overset{2}{\geq} s^*_{\mathcal{NN}}(\mathcal{C}) \overset{3}{\geq} s^*_{\text{dict}}(\mathcal{C}, \phi),$$

because

1. the dictionary ϕ is rate-optimal,
2. quantized neural networks are encoders, as shown in Video 5, and
3. quantized dictionary approximations are quantized neural networks, as shown in the last theorem. □

Remark: This corollary applies to all examples of signal classes and dictionaries discussed so far.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: Why and under what conditions is the effective network approximation rate lower-bounded by the dictionary approximation rate?
- Check: How wide and deep are the approximating networks?
- Check: How does the present transfer-of-approximation result differ from the one of Week 3?
- Discussion: What does the result say about deep learning? What are limitations of the result?

Mathematics of Deep Learning, Summer Term 2020

Week 7, Video 7

Wrapup

Philipp Harms Lars Niemann

University of Freiburg

Outlook on this week's discussion and reading session

- Reading:
  - Bölcskei, Grohs, Kutyniok, Petersen (2017): Optimal approximation with sparsely connected deep neural networks
  - Donoho (2001): Sparse Components of Images and Optimal Atomic Decompositions. In: Constructive Approximation 17, pp. 353–382
  - Shannon (1959): Coding Theorems for a Discrete Source with a Fidelity Criterion. In: International Convention Record 7, pp. 325–350

Summary by learning goals

Having heard this lecture, you can now ...

- Derive lower bounds on effective network approximation rates from harmonic analysis.
- Derive upper bounds on effective network approximation rates from rate-distortion theory.
- Explain why neural networks are optimal descriptors of a wide variety of signal classes.

Mathematics of Deep Learning, Summer Term 2020

Week 8

ReLU Networks and the Role of Depth

Philipp Harms Lars Niemann

University of Freiburg

Overview of Week 8

1. Operations on ReLU Networks
2. ReLU Representation of Saw-Tooth Functions
3. Saw-Tooth Approximation of the Square Function
4. ReLU Approximation of Multiplication
5. ReLU Approximation of Analytic Functions
6. Wrapup

Acknowledgement of Sources

Sources for this lecture:

- Philipp Christian Petersen (Faculty of Mathematics, University of Vienna): Course on Neural Network Theory.
- Perekrestenko, Grohs, Elbrächter, Bölcskei (2018): The universal approximation power of finite-width deep ReLU Networks. `arXiv:1806.01528`
- E, Wang (2018): Exponential convergence of the deep neural approximation for analytic functions. `arXiv:1807.00297`
- Yarotsky (2017): Error bounds for approximations with deep ReLU networks. Neural Networks 94, pp. 103–114.

Mathematics of Deep Learning, Summer Term 2020

Week 8, Video 1

Operations on ReLU Networks

Philipp Harms Lars Niemann

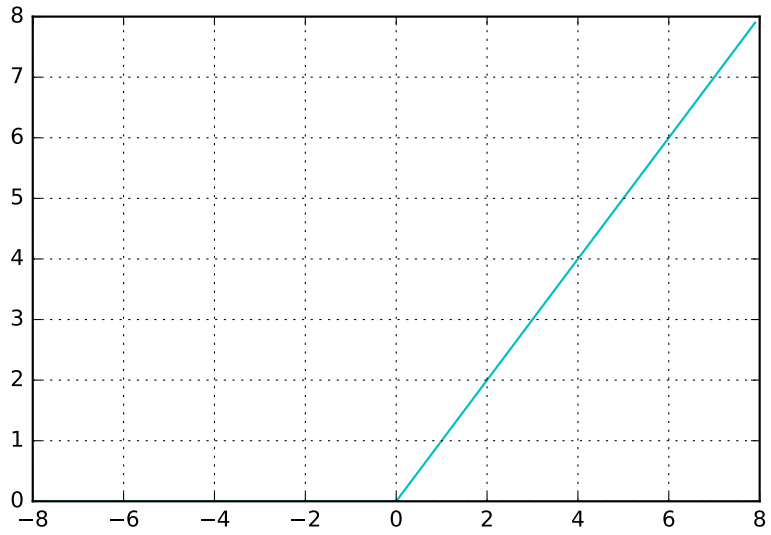
University of Freiburg

Repetition: ReLU Activation Function

Definition

The rectified linear unit (ReLU) activation function is defined as

$$\rho_R(x) = \max(0, x), \qquad x \in \mathbb{R}.$$



Remark: The ReLU function is not sigmoidal but discriminatory.

Networks of Bounded Width with Bounded Weights

Remark:

- Previously, the focus was on wide networks of bounded depth.
- For ReLU networks, we focus on deep networks of bounded width.

Definition

Let $\Phi = ((A_1, b_1), \ldots, (A_L, b_L))$ be a neural network with architecture $(N_0, N_1, \ldots, N_L)$.

- The width of Φ is defined as $\mathrm{W}(\Phi) := \max_i N_i$.
- The weight bound of Φ is defined as

$$\mathrm{B}(\Phi) := \max\{\max_i \|A_i\|_{\infty,\infty}, \max_i \|b_i\|_\infty\},$$

where the norms $\|\cdot\|_{\infty,\infty}$ and $\|\cdot\|_\infty$ are the maxima of the absolute values of the matrix or vector entries, respectively.

ReLU Representation of the Identity

Lemma

For each $d \in \mathbb{N}$ and $L \in \mathbb{N}$, the identity on $\mathbb{R}^d$ can be realized as $\mathrm{Id}_{\mathbb{R}^d} = \mathrm{R}(\Phi^{\mathrm{Id}}_{d,L})$ for a ReLU network $\Phi^{\mathrm{Id}}_{d,L}$ with $\mathrm{B}(\Phi^{\mathrm{Id}}_{d,L}) = 1$, $\mathrm{W}(\Phi^{\mathrm{Id}}_{d,L}) = 2d$, and $\mathrm{L}(\Phi^{\mathrm{Id}}_{d,L}) = L$.

Proof: For $L = 1$ we use $\Phi^{\mathrm{Id}}_{d,1} := ((\mathrm{Id}_{\mathbb{R}^d}, 0))$, and for $L \geq 2$, the network

$$\Phi^{\mathrm{Id}}_{d,L} := \left(\left(\begin{pmatrix} \mathrm{Id}_{\mathbb{R}^d} \\ -\mathrm{Id}_{\mathbb{R}^d} \end{pmatrix}, 0 \right), (\mathrm{Id}_{\mathbb{R}^{2d}}, 0), \ldots (\mathrm{Id}_{\mathbb{R}^{2d}}, 0), ((\mathrm{Id}_{\mathbb{R}^d}, -\mathrm{Id}_{\mathbb{R}^d}), 0) \right)$$

has the desired properties thanks to the algebraic relations

$$\rho_R(x) - \rho_R(-x) = x, \qquad \rho_R(\rho_R(x)) = \rho_R(x).$$

□

Problem: Lack of Sparsity in Network Concatenations

Example: Lack of sparsity in network concatenations.

- Let $n \in \mathbb{N}$ and define the neural network Φ by

$$\Phi := ((A_1, 0), (A_2, 0)),$$

  where $A_1 = (1, \ldots, 1)^\top \in \mathbb{R}^{n \times 1}$ and $A_2 = (1, \ldots, 1) \in \mathbb{R}^{1 \times n}$.

- Φ realizes the map

$$\mathbb{R} \ni x \mapsto (x, \ldots, x) \mapsto (x_+, \ldots, x_+) \mapsto x_+ + \cdots + x_+ = n x_+ \in \mathbb{R}.$$

- Then $\mathrm{M}(\Phi) = 2n$ but $\mathrm{M}(\Phi \bullet \Phi) = 2n + n^2$ because

$$\Phi \bullet \Phi = ((A_1, 0), (A_1 A_2, 0), (A_2, 0)).$$

- Hence, the number of weights of a concatenated network scales quadratically in the number of weights of the individual networks.

Solution: Sparse Concatenation

Remark: The lack of sparsity of concatenations motivates the following definition:

Definition

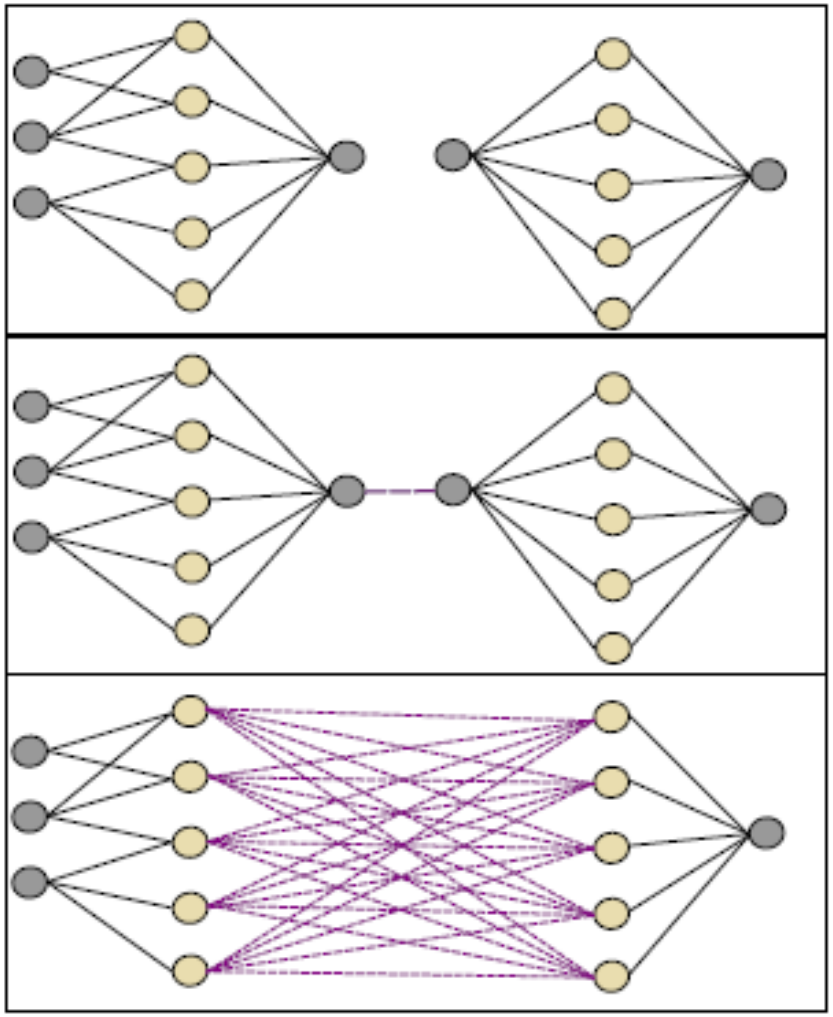
The sparse concatenation of a neural network Φ^1 with input dimension d and neural network Φ_2 with output dimension d is defined as

$$\Phi^1 \odot \Phi^2 := \Phi^1 \bullet \Phi^{\mathrm{Id}}_{d,2} \bullet \Phi^2,$$

where $\Phi^{\mathrm{Id}}_{d,2}$ is the 2-layer ReLU representation of the identity on $\mathbb{R}^d$.

Remark: Similarly, using $\Phi^{\mathrm{Id}}_{d,L}$ with $L > 2$, one can define sparse concatenations of increased depth.

Concatenation versus Sparse Concatenation



Top: Two neural networks, Middle: Sparse Concatenation, Bottom: Concatenation. [Figure from Petersen, Ch. 3]

Properties of Sparse Concatenation

Lemma

If Φ^1 has input dimension d and Φ_2 has output dimension d, then the sparse concatenation $\Phi^1 \odot \Phi^2$ satisfies

$$\begin{aligned}
\mathrm{R}(\Phi^1 \odot \Phi^2) &= \mathrm{R}(\Phi^1) \circ \mathrm{R}(\Phi^2), \\
\mathrm{L}(\Phi^1 \odot \Phi^2) &= \mathrm{L}(\Phi^1) + \mathrm{L}(\Phi^2), \\
\mathrm{M}(\Phi^1 \odot \Phi^2) &\leq 2(\mathrm{M}(\Phi^1) + \mathrm{M}(\Phi^2)), \\
\mathrm{W}(\Phi^1 \odot \Phi^2) &\leq \max(\mathrm{W}(\Phi^1), \mathrm{W}(\Phi^2), 2d), \\
\mathrm{B}(\Phi^1 \odot \Phi^2) &\leq \max(\mathrm{B}(\Phi^1), \mathrm{B}(\Phi^2)).
\end{aligned}$$

Remark: Most importantly, the number of weights increases linearly rather than quadratically, and the weights remain bounded.

Proof: Properties of Sparse Concatenation

Proof:

- Sparse concatenation realizes function composition because

$$\mathrm{R}(\Phi^1 \bullet \Phi^{\mathrm{Id}}_{d,2} \bullet \Phi^2) = \mathrm{R}(\Phi^1) \circ \mathrm{R}(\Phi^{\mathrm{Id}}_{d,2}) \circ \mathrm{R}(\Phi^2) = \mathrm{R}(\Phi^1) \circ \mathrm{R}(\Phi^2).$$

- The width, depth, weight bound, and number of weights can be estimated from the following explicit formula:

$$\begin{aligned}
&((A^1_1, b^1_1), \ldots, (A^1_{L_1}, b^1_{L_1})) \odot ((A^2_1, b^2_1), \ldots, (A^2_{L_2}, b^1_{L_2})) \\
&\quad = \left((A^2_1, b^2_1), \ldots, (A^2_{L_2-1}, b^2_{L_2-1}), \left(\begin{pmatrix} A^2_{L_2} \\ -A^2_{L_2} \end{pmatrix}, \begin{pmatrix} b^2_{L_2} \\ -b^2_{L_2} \end{pmatrix} \right), \right. \\
&\qquad\qquad \left. ((A^1_1, -A^1_1), b^1_1), (A^1_2, b^1_2), \ldots, (A^1_{L_1}, b^1_{L_1}) \right).
\end{aligned}$$

□

Skip Connections

Remark: Recall that a network $\Phi = ((A_1, b_1), \ldots, (A_L, b_L))$ can be represented as a computational graph with edges corresponding to the non-zero entries of the matrices A_i.

Definition

A skip connection is an edge between non-adjacent layers in the computational graph of a network.

Remark:

- Networks with skip connections have been highly successful in image recognition.
- The ReLU representation of the identity allows one to rewrite networks with skip connections as networks without skip connections.

Deep Linear Combinations of Networks

Remark:

- The following implementation of linear combinations increases the depth, and not the width, of the networks.
- As scalar multiplication does not affect the network structure, we focus on sums of networks.

Lemma

For any networks $\Phi_1, \ldots, \Phi_k$ *with input dimension* d *and output dimension* n*, there exists a network* Φ *with* $\mathrm{B}(\Phi) \leq \max_i \mathrm{B}(\Phi_i)$, $\mathrm{W}(\Phi) \leq \max_i \mathrm{W}(\Phi) + 2d + 2n$*, and* $\mathrm{L}(\Phi) = \sum_i \mathrm{L}(\Phi_i)$ *such that*

$$\mathrm{R}(\Phi) = \sum_i \mathrm{R}(\Phi_i).$$

Proof: Deep Linear Combinations of Networks

Proof:

- Let Φ^{sum} and Φ^{diag} be the single-layer networks realizing the maps

$$\mathrm{sum}\colon \mathbb{R}^d \times \mathbb{R}^n \times \mathbb{R}^n \ni (x, y, z) \mapsto (x, y + z) \in \mathbb{R}^d \times \mathbb{R}^n,$$
$$\mathrm{diag}\colon \mathbb{R}^d \times \mathbb{R}^n \ni (x, y) \mapsto (x, x, y) \in \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^n.$$

- Then the sum with skip connections

$$\mathbb{R}^d \times \mathbb{R}^n \ni (x, y) \mapsto (x, \mathrm{R}(\Phi_i)(x) + y) \in \mathbb{R}^d \times \mathbb{R}^n$$

is realized by the network

$$\Psi_i := \Phi^{\mathrm{sum}} \bullet \mathrm{FP}\left(\Phi^{\mathrm{Id}}_{d,\mathrm{L}(\Phi_i)}, \Phi_i, \Phi^{\mathrm{Id}}_{N,\mathrm{L}(\Phi_i)}\right) \bullet \Phi^{\mathrm{diag}},$$

which satisfies $\mathrm{B}(\Psi_i) \leq \max\{\mathrm{B}(\Phi_i), 1\}$, $\mathrm{W}(\Psi_i) \leq \mathrm{W}(\Phi_i) + 2d + 10$, $\mathrm{L}(\Psi_i) = \mathrm{L}(\Phi_i)$.

Proof: Deep Linear Combinations of Networks

- Let Φ^{pr} and Φ^{ins} be the single-layer networks realizing the maps

$$\mathrm{pr}\colon \mathbb{R}^d \times \mathbb{R}^n \ni (x, y) \mapsto y \in \mathbb{R}^n,$$
$$\mathrm{ins}\colon \mathbb{R}^d \ni x \mapsto (x, 0) \in \mathbb{R}^d \times \mathbb{R}^n.$$

- Then the network $\Phi := \Phi^{\mathrm{pr}} \bullet \Psi_1 \odot \cdots \odot \Psi_k \bullet \Phi^{\mathrm{ins}}$ has the desired properties. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How can the identity be realized using ReLU networks?
- Repetition: What is sparse concatenation, and how does it differ from non-sparse concatenation?
- Repetition: What are skip connections, what are they good for, and how can they be implemented using ReLU networks?
- Discussion: To what extent are the results of this video limited to ReLU networks?

Mathematics of Deep Learning, Summer Term 2020

Week 8, Video 2

ReLU Representation of Saw-Tooth Functions

Philipp Harms Lars Niemann

University of Freiburg

ReLU Representation of the Hat Function

Lemma

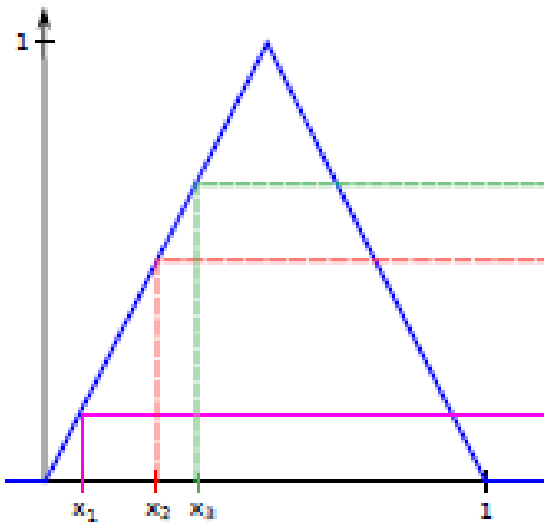
The hat function

$$F(x) := \rho_R(2x) - 2\rho_R(2x-1) + \rho_R(2x-2)$$

equals the ReLU realization of the network $\Phi^{\mathrm{hat}} := ((A_1, b_1), (A_2, 0))$ *with*

$$A_1 := (2,2,2)^\top, \quad b_1 := (0,-1,-2)^\top, \quad A_2 := (1,-2,1)\,.$$

This network satisfies $\mathrm{B}(\Phi^{\mathrm{hat}}) = 2$, $\mathrm{W}(\Phi^{\mathrm{hat}}) = 3$, *and* $\mathrm{L}(\Phi^{\mathrm{hat}}) = 2$.



ReLU Representation of Saw-Tooth Functions

Theorem

For any $n \in \mathbb{N}$, the saw-tooth function F_n given by $F_n(x) = 0$ for $x \notin (0,1)$ and

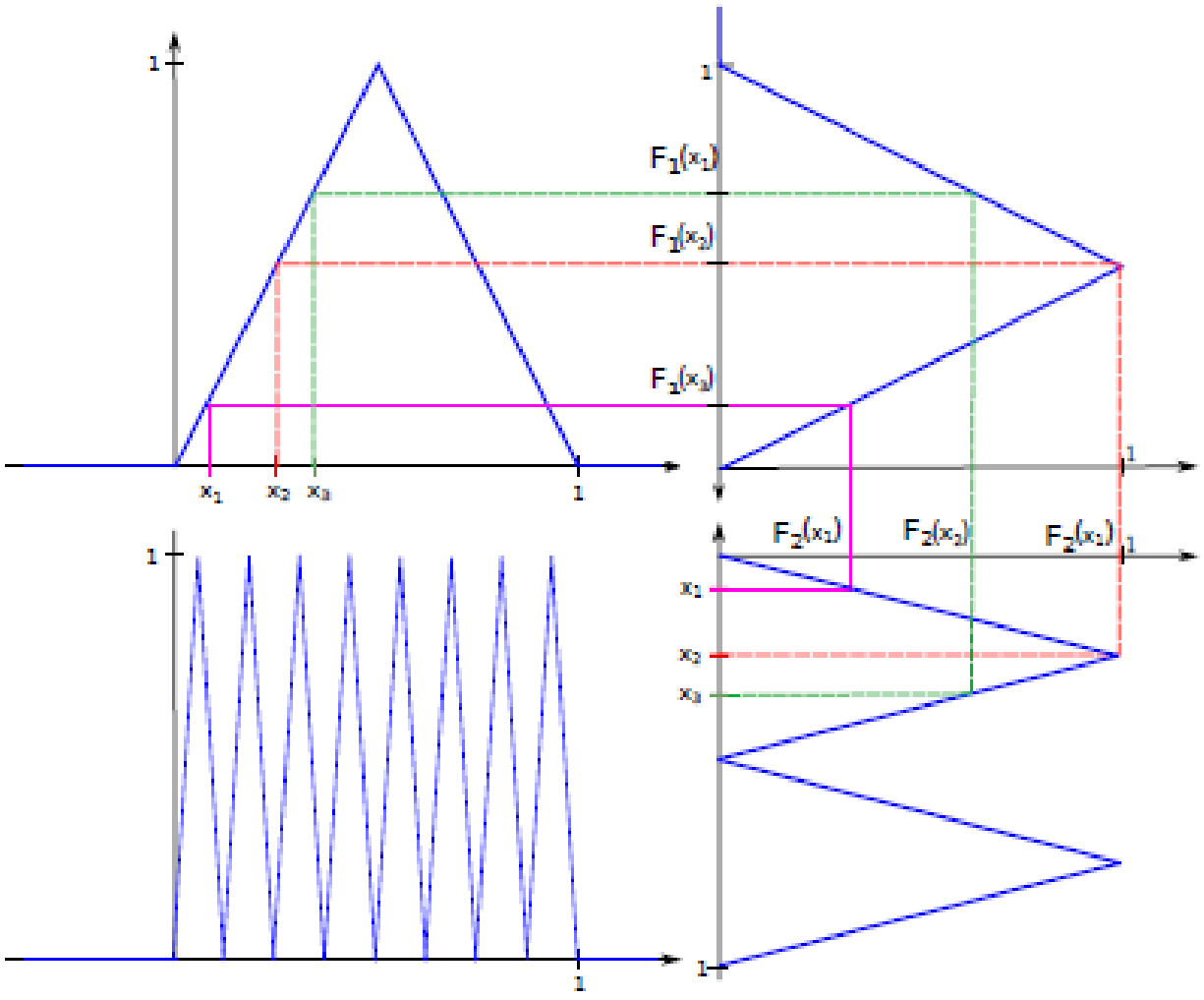
$$F_n(x) := \begin{cases} 2^n(x - i2^{-n}), & x \in [i2^{-n}, (i+1)2^{-n}],\ i \text{ even}, \\ 2^n((i+1)2^{-n} - x), & x \in [i2^{-n}, (i+1)2^{-n}],\ i \text{ odd}, \end{cases}$$

equals the ReLU realization of the concatenated network $\Phi_n := \bullet^n \Phi^{\mathrm{hat}}$ with $\mathrm{B}(\Phi_n) \leq 4$, $\mathrm{W}(\Phi_n) \leq 3$, and $\mathrm{L}(\Phi_n) = n + 1$.

Proof:

- F_n is the n-fold composition of hat functions.
- Thus, the n-fold concatenation $\bullet^n \Phi^{\mathrm{hat}}$ has the desired properties. □

Visualization of Saw-Tooth Functions



Top Left: F_1, Bottom Right: F_2, Bottom Left: F_4.

[Figure from Petersen, Ch. 3]

The Role of Depth

Remark: The theorem is surprising for the following reason:

- The realization of a shallow network Φ with two layers and input dimension 1 is piece-wise linear with at most $\mathrm{W}(\Phi)$ pieces.
- Similarly, networks of depth bounded by L have at most $\mathrm{W}(\Phi)^{L-1}$ pieces.
- In contrast, the previously introduced deep networks realize the saw-tooth function F_n, which has exponentially many pieces in $\mathrm{L}(\Phi)$.
- Thus, saw-tooth functions F_n can be represented very efficiently by deep networks, but not very efficiently by shallow networks.

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How can saw-tooth functions be represented by deep ReLU networks?
- Check: Why can the realization of a two-layer network Φ have at most $\mathrm{M}(\Phi)$ pieces?
- Check: Verify that the saw-tooth function is a composition of hat functions.
- Background: Can you show that the ReLU function is discriminatory?

Mathematics of Deep Learning, Summer Term 2020

Week 8, Video 3

Saw-Tooth Approximation of the Square Function

Philipp Harms Lars Niemann

University of Freiburg

Saw-Tooth Approximation of the Square Function

Setting: Let F_n, $n \in \mathbb{N}$, denote the saw-tooth functions of Video 2.

Lemma

The piece-wise linear functions

$$H_n(x) := x - \sum_{k=1}^{n} F_k(x) 2^{-2k}, \qquad n \in \mathbb{N},\ x \in \mathbb{R},$$

approximate the square function at an exponential rate:

$$\sup_{x \in [0,1]} \left| x^2 - H_n(x) \right| \leq 2^{-2(n+1)}, \qquad n \in \mathbb{N}\,.$$

Remark: This makes us optimistic that, using sufficiently deep networks, we can approximate the square function efficiently.

Visualizing the Approximation of the Square Function

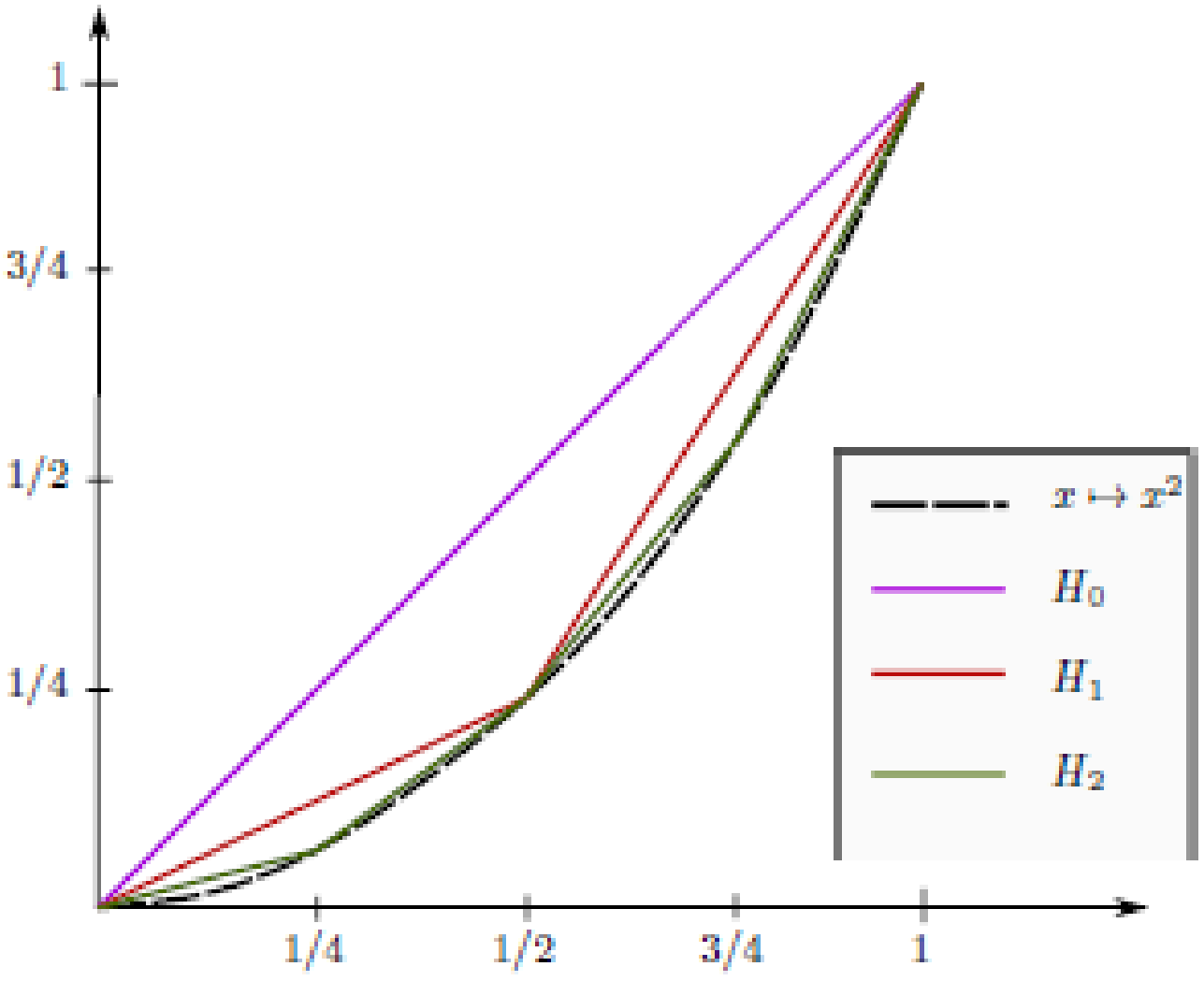


Figure: Approximants $H_n(x) := x - \sum_{k=1}^{n} F_k(x)2^{-2k}$ of the square function x^2.

[Figure from Petersen, Ch. 3]

Proof: Approximating the Square Function by Saw-Tooths

Proof:

- By induction, the function H_n is piecewise linear with breakpoints $k2^{-n}$ for $k \in \{0, \ldots, 2^n\}$, and $H_n(x) = x^2$ at the breakpoints.
- By convexity, $H_n(x) \geq x^2$ for $x \in [0, 1]$.
- For any x between the breakpoints $\ell := k2^{-n}$ and $u := (k+1)2^{-n}$,

$$\left|H_n(x) - x^2\right| = H_n(x) - x^2 = \frac{u-x}{u-\ell}\ell^2 + \frac{x-\ell}{u-\ell}u^2 - x^2.$$

- This quadratic function assumes its maximum at its unique critical point x^*, and one easily verifies that

$$x^* = \frac{u+\ell}{2}, \qquad H_n(x^*) - (x^*)^2 = \left(\frac{u-\ell}{2}\right)^2 = 2^{-2(n+1)}.$$

□

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How can the square function be approximated by linear combinations of saw-tooth functions?

- Check: Verify that a secant approximation of the square function is worst half-way between the abscissas of the intersection.

  Discussion: How could the saw-tooth approximation of the square function be implemented by ReLU networks. Spoiler alert: think about this before you watch the next video.

Mathematics of Deep Learning, Summer Term 2020

Week 8, Video 4

ReLU Approximation of Multiplication

Philipp Harms Lars Niemann

University of Freiburg

Approximating the Square Function

Remark: As an auxiliary result, we will approximate the square function by ReLU networks, building on the saw-tooth approximations of the square function.

Lemma

The square function can be approximated by ReLU networks at an exponential rate:

$$\forall n \in \mathbb{N}\ \exists \Phi : \mathrm{B}(\Phi) \leq 4, \mathrm{W}(\Phi) \leq 5, \mathrm{L}(\Phi) = n + 2,$$

$$\sup_{x \in [-1,1]} \left| x^2 - \mathrm{R}(\Phi)(x) \right| \leq 2^{-2(n+1)}.$$

Attempted Proof: Approximating the Square Function

Attempted proof: Strategy of Yarotsky (2017).

- Approximate the square function by saw-tooth functions: For any $n \in \mathbb{N}$,

$$\sup_{x\in[0,1]} \left|x^2 - H_n(x)\right| \leq 2^{-2(n+1)}, \qquad H_n(x) = x - \sum_{k\leq n} F_k 2^{-2k}.$$

- Represent each saw-tooth function by a network: $F_k = \mathrm{R}(\bullet^k \Phi^\wedge)$.
- Use skip connections to get networks of equal depth: $F_k = \mathrm{R}(\Phi_k)$ with $\Phi_k := \Phi^{\mathrm{Id}}_{1,n-k} \odot \bullet^k \Phi^\wedge$.
- Take linear combinations of $\Phi_1, \ldots, \Phi_n$ to obtain networks of width proportional to n.
- Alternatively, using deep linear combinations, one obtains networks of depth proportional to n^2.
- In any case, this strategy is sub-optimal. □

Proof: Approximating the Square Function

Proof: Strategy of Perekrestenko e.a. (2018).

- As before, approximate the square by saw-tooth functions H_n:

$$\sup_{x\in[0,1]} \left|x^2 - H_n(x)\right| \leq 2^{-2(n+1)}, \qquad H_n(x) = x - \sum_{k\leq n} F_k 2^{-2k}.$$

- Recall that F_n is the n-fold composition of the hat function

$$F(x) := 2\rho_R(x) - 4\rho_R(x - \tfrac{1}{2}) + 2\rho_R(x-1),$$

and note that $H_n(x) = H_{n-1}(x) - 2^{-2n}F_n(x)$.

- This yields the recursion

$$\begin{cases} F_n(x) = 2\rho_R(F_{n-1}(x)) - 4\rho_R(F_{n-1}(x) - \frac{1}{2}) + 2\rho_R(F_{n-1}(x) - 1), \\ H_n(x) = \rho_R(H_{n-1}(x)) - \rho_R(-H_{n-1}(x)) - 2^{-2n}F_n(x), \end{cases}$$

where the term $F_n(x)$ on the right-hand side can be substituted by a term involving the functions $F_{n-1}(x)$ using the first equation.

Proof: Approximating the Square Function (cont.)

- Each recursive step corresponds to a network layer:

$$\begin{pmatrix} F_n \\ H_n \end{pmatrix} = W_1 \rho_R \left(W_2 \begin{pmatrix} F_{n-1} \\ H_{n-1} \end{pmatrix} \right),$$

$$W_1(x) = \begin{pmatrix} 2 & -2^{-2n+1} \\ -4 & 2^{-2n+2} \\ 2 & 2^{-2n+1} \\ 0 & 1 \\ 0 & -1 \end{pmatrix}^\top \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix},$$

$$W_2(x) = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} 0 \\ 1/2 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$

- Thus, using non-sparse concatenation •, the iteration for H_n with $F_0(x) = |x|$ and $H_0(x) = |x|$ can be realized by a ReLU network Φ of depth $n + 2$, width 5, and weights bounded by 4. □

Approximating Multiplication

Remark: The previous lemma on approximation of the square function implies the following theorem:

Theorem

Multiplication can be approximated by ReLU networks at an exponential rate:

$$\forall n \in \mathbb{N}\ \exists \Phi : \mathrm{B}(\Phi) \leq 8, \mathrm{W}(\Phi) \leq 10, \mathrm{L}(\Phi) = n + 2,$$

$$\sup_{x,y \in [-1,1]} |xy - \mathrm{R}(\Phi)(x,y)| \leq 2^{-2n-1}.$$

Remark: On domains $x, y \in [-K, K]$, the weight bound changes to a quadratic polynomial in K.

Proof: Approximating Multiplication

Proof:

- By polarization, we have for $x, y \in [-1, 1]$ that

$$xy = \left(\frac{x+y}{2}\right)^2 - \left(\frac{x-y}{2}\right)^2. \qquad (*)$$

- Approximate the square function on $[-1, 1]$ with precision $2^{-2(n+1)}$ by a neural network Φ_0 with $\mathrm{B}(\Phi_0) \leq 4$, $\mathrm{W}(\Phi_0) \leq 5$, and $\mathrm{L}(\Phi_0) = n + 2$.
- Define neural networks Φ_1 and Φ_2 as

$$\Phi_1 := \left(\left(\frac{1}{2}\begin{pmatrix} 1 & 1 \\ -1 & -1 \\ 1 & -1 \\ -1 & 1 \end{pmatrix}, 0\right)\right), \qquad \Phi_2 := \Big(\big((1, -1), 0\big)\Big).$$

- As the realization of $\Phi := \Phi_2 \bullet \mathrm{FP}(\Phi_0, \Phi_0) \bullet \Phi_1$ equals $(*)$ with squares replaced by $\mathrm{R}(\Phi_0)$, the error is at most 2^{-2n-1}. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How can multiplication be approximated by ReLU networks at an exponential rate?
- Transfer: Compare the ReLU approximation to the sigmoidal approximation of multiplication. See Week 3.
- Discussion: Using harmonic analysis we previously established polynomial upper bounds on network approximation rates—are they in contradiction to the exponential approximation rate established here?

Mathematics of Deep Learning, Summer Term 2020

Week 8, Video 5

ReLU Approximation of Analytic Functions

Philipp Harms Lars Niemann

University of Freiburg

Approximating Monomials

Lemma

Monomials can be approximated by ReLU networks at an exponential rate:

$$\forall d, p, n \in \mathbb{N}\ \forall i_1, \ldots, i_p \in \{1, \ldots, d\}\ \exists \Phi :$$
$$\mathrm{B}(\Phi) \leq 8, \mathrm{W}(\Phi) \leq 2d + 10, \mathrm{L}(\Phi) = p(n+2),$$

$$\sup_{x \in [-1,1]^d} \left| x_{i_1} \cdots x_{i_p} - \mathrm{R}(\Phi) \right| (x) \leq 2^{-2n-1}$$

Remark:

- Via dictionary learning, this leads to optimal polynomial approximation rates for many signal classes.
- More interestingly, in contrast to our previous results, it also leads to exponential approximation rates for real-analytic functions, including e.g. sinusoidal functions and oscillatory textures.

Proof: Approximating Monomials

Proof:

- For any $i \in \{1, \dots, d\}$, the multiplication with skip connections

$$(x_1, \dots, x_d, y) \mapsto (x_1, \dots, x_d, x_i y)$$

can be approximated by a network Ψ_i with $\mathrm{B}(\Psi_i) \leq 8$, $\mathrm{W}(\Psi_i) \leq 2d + 10$, $\mathrm{L}(\Psi_i) = n + 2$, and

$$\sup_{x_1,\dots,x_d,y \in [-1,1]} \|(x_1, \dots, x_d, x_i y) - \mathrm{R}(\Psi_i)(x_1, \dots, x_d, y)\|_\infty \leq 2^{-2n-1}.$$

- As the realizations of Ψ_i are 1-Lipschitz and bounded by 1, the net

$$\Phi := (((0_{(\mathbb{R}^d)^*}, 1), 0)) \bullet \Psi_{i_1} \odot \cdots \odot \Psi_{i_p} \bullet \left(\begin{pmatrix} \mathrm{Id}_{\mathbb{R}^d} \\ 0_{(\mathbb{R}^d)^*} \end{pmatrix}, \begin{pmatrix} 0_{\mathbb{R}^d} \\ 1 \end{pmatrix} \right)$$

satisfies $\mathrm{B}(\Phi) \leq 8$, $\mathrm{W}(\Phi) \leq 2d + 10$, $\mathrm{L}(\Phi) = p(n + 3)$, and

$$\sup_{x_1,\dots,x_d \in [-1,1]} \left| x_{i_1} \cdots x_{i_p} - \mathrm{R}(\Phi)(x_1, \dots, x_d) \right| \leq 2^{-2n-1}. \qquad \square$$

Real-Analytic Functions

Definition

A function $f : (-r, r)^d \to \mathbb{R}$ is real-analytic if it is given by a power series

$$f(x) = \sum_{k \in \mathbb{N}^d} a_k x^k, \qquad x \in (-r, r)^d,$$

for some coefficients $(a_k)_{k \in \mathbb{N}^d}$.

Remark:

- The power series converges absolutely on $(-r, r)^d$.
- Thus, if $r > 1$, then a is summable, i.e., $\|a\|_{\ell^1} := \sum_{k \in \mathbb{N}^d} |a_k| < \infty$.

Approximating Real-Analytic Functions

Theorem

Real-analytic functions can be approximated by ReLU networks:

$$\forall d \in \mathbb{N}_{\geq 2}\ \forall \delta > 0\ \exists \bar{\epsilon} > 0\ \forall \epsilon \in (0, \bar{\epsilon})\ \forall (a_k)_{k \in \mathbb{N}^d} \in \ell^1\ \exists \Phi :$$

$$\mathrm{B}(\Phi) \leq 8 \sum_{k \in \mathbb{N}^d} |a_k|, \mathrm{W}(\Phi) \leq (2d + 10), \mathrm{L}(\Phi) \leq \left(e\left(\tfrac{1}{d\delta} \log_2 \tfrac{1}{\epsilon} + 1 \right) \right)^{2d},$$

$$\sup_{x \in [-1+\delta, 1-\delta]^d} \left| \sum_{k \in \mathbb{N}^d} a_k x^k - \mathrm{R}(\Phi)(x) \right| \leq 2\epsilon \|a_k\|_{\ell^1}.$$

Remark: Note that the error decays exponentially in $L^{1/(2d)}$ because

$$L(\Phi) \leq \left(e\left(\tfrac{1}{d\delta} \log_2 \tfrac{1}{\epsilon} + 1 \right) \right)^{2d} \Leftrightarrow \epsilon \leq \exp(-d\delta(e^{-1} L^{1/(2d)} - 1)).$$

Approximating Real-Analytic Functions

Proof:

- Without loss of generality, $\|a_k\|_{\ell^1} = 1$.
- Truncation: Let $p := \lceil \frac{1}{\delta} \log_2 \frac{1}{\epsilon} \rceil$, $f(x) := \sum_{k \in \mathbb{N}^d} a_k x^k$, $f_p(x) := \sum_{k \in \mathbb{N}^d_{\leq p}} a_k x^k$. Then

$$\sup_{x \in [-1+\delta, 1-\delta]^d} |f(x) - f_p(x)| \leq (1-\delta)^p \leq \epsilon.$$

- Monomial approximation: Let $n := \lceil \frac{1}{2} \log_2 \frac{1}{\epsilon} \rceil$. Approximate each monomial x^k by a network Φ_k with $\mathrm{B}(\Phi) \leq 8$, $\mathrm{W}(\Phi) \leq 2d + 10$, $\mathrm{L}(\Phi_k) = p(n+2)$, and

$$\sup_{x \in [-1,1]^d} \left| x^k - \mathrm{R}(\Phi_k)(x) \right| \leq 2^{-2n-1} \leq \epsilon.$$

Approximating Real-Analytic Functions

- Deep linear combinations of the $\binom{p+d}{d}$ monomials: there is a network Φ with $\mathrm{B}(\Phi) \leq 8$, $\mathrm{W}(\Phi) \leq 2d + 11$, $\mathrm{L}(\Phi) = p(n+2)\binom{p+d}{d}$,

$$\sup_{x \in [-1,1]^d} |f_p(x) - \mathrm{R}(\Phi)(x)| \leq \epsilon.$$

- Depth bound: for sufficiently small $\bar{\epsilon}$ and $\epsilon < \bar{\epsilon}$,

$$\begin{aligned}
\mathrm{L}(\Phi) = p(n+2)\binom{p+d}{d} &= p(n+2)\frac{(p+d)\cdots(p+1)}{d!} \\
&\leq p(n+2)\left(\frac{p+d}{d/e}\right)^d = p(n+2)\left(e(\tfrac{p}{d}+1)\right)^d \\
&\leq \left(e(\tfrac{1}{d\delta}\log_2\tfrac{1}{\epsilon}+1)\right)^{2d},
\end{aligned}$$

where the last inequality follows by an elementary calculation from the definitions of p and n and the assumption $d \geq 2$. □

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How can real-analytic functions be approximated by ReLU networks at an exponential rate?
- Background: What is the difference between smooth, real-analytic, and holomorphic functions?
- Check: Prove the inequality $d! \geq (d/e)^d$, which was used in the last proof. Hint: $d^d/d!$ is a summand in the series expansion of e^d.
- Discussion: Can real-analytic functions be approximated by shallow networks at an exponential rate?
- Transfer: What other assumptions on the signal class besides real analyticity might increase the approximation rate?

Mathematics of Deep Learning, Summer Term 2020

Week 8, Video 6

Wrapup

Philipp Harms Lars Niemann

University of Freiburg

Outlook on this week's discussion and reading session

- Reading:
  - Yarotsky (2017): Error bounds for approximations with deep ReLU networks. Neural Networks 94, pp. 103–114.
  - Perekrestenko, Grohs, Elbrächter, Bölcskei (2018): The universal approximation power of finite-width deep ReLU Networks. `arXiv:1806.01528`
  - E, Wang (2018): Exponential convergence of the deep neural approximation for analytic functions. `arXiv:1807.00297`

Summary by learning goals

Having heard this lecture, you can now ...

- Establish exponential rates for the approximation of real-analytic functions by deep ReLU networks.
- Explain the role of skip connections in this construction.

Review and Outlook

- Topics covered in this lecture series:
  - Statistical learning theory
  - Universal approximation theorems
  - Dictionary learning
  - Kolmogorov–Arnold representation
  - Harmonic analysis
  - Information theory
  - ReLU networks and the role of depth
- Topics not covered in this lecture series: (non-exhaustive)
  - Residual, recurrent, and adversarial networks; auto-encoders
  - Manifold assumptions on the data distribution
  - Generalization capability and implicit regularization
  - Many practical issues