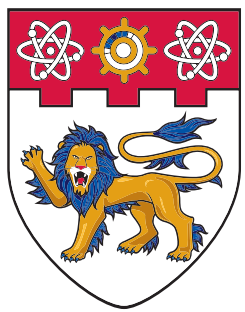
MH3520:Mathematics of Deep Learning

Zhongjian Wang

Nanyang Technological University Singapore



NANYANG
TECHNOLOGICAL
UNIVERSITY
SINGAPORE

Welcome to the course on mathematics of deep learning

- Instructor: Zhongjian Wang
- Contact: zhongjian.wang@ntu.edu.sg or Teams
- Office Hour: Thursday 15:30 - 16:20
- Office Location: SPMS-MAS-05-23
- Special thanks for the previous instructor Dr. Phillip Harms who suggested me to significantly adapt the difficulty of the course. I am pursuing it on a running basis.

Overview of the course

1. Introduction to deep learning
2. Basics of numerical optimization
3. Training neural networks
4. Basics of probability theory
5. Statistical learning theory
6. Basics of functional analysis
7. Universality of multi-layer perceptrons
8. Basics of approximation theory
9. Splines and approximation by wide networks
10. Analyticity and approximation by deep networks

The following are optional chapters for further reading.

- Harmonic analysis and approximation by wide networks
- Coding theory and best approximation rates

Contents of the course

This course presents several mathematical perspectives on deep learning:

- Numerical optimization: The training of neural networks is a high-dimensional nonlinear optimization problem, which is solved by stochastic gradient descent.
- Statistical Learning Theory: The goal in supervised learning is to estimate an unknown function from noisy data. Statistical learning theory provides a general error analysis for problems of this kind.
- Function Approximation: Under suitable conditions, any function can be approximated with arbitrary accuracy by neural networks. Function approximation theory provides upper bounds for the approximation error in relation to the size of the network.
- Coding theory (Optional Reading): A trained neural network can be seen as an encoder for the learned function. Coding theory provides lower bounds for the approximation error in relation to the size of the network.

Intended Learning Outcomes

This course should help you...

- Understand why and under what conditions deep learning methods can be expected to work well—or not.
- Dive into the diverse mathematical theories which have a say on this matter.
- Appreciate the many open questions and challenges on the road towards a more complete understanding of deep learning.
- Get started with numerical implementations of simple learning tasks in Python.

Beyond the scope of this course

The following topics are not covered in the course:

- Generalization error: This course focusses on bounds for the approximation error and merely touches upon the generalization error. Bounds on both errors are needed for a complete error analysis.
- Unsupervised learning: This course focusses on supervised learning and does not cover unsupervised learning via deep networks (e.g. autoencoders or reinforcement learning).
- Efficient Numerics: Despite their importance, advances in software and hardware implementations of deep learning algorithms are beyond the scope of this course. (e.g. GPU acceleration)
- Applications: While deep learning methods have been adopted in a plethora of diverse applications, this course focusses on the underlying theoretical principles. (e.g. AI, genAI)

Prerequisites

The following courses are prerequisites:

- MH2100 Calculus III: Differential calculus in multiple dimensions is needed for continuous optimization methods such as gradient descent.
- MH3500 Statistics: Some basic notions and results of probability theory and asymptotic statistics are needed for statistical learning theory.
- MH3600 Introduction to Topology: Point-set topology and functional analysis are needed for function approximation theory.
- PS0001 Introduction to Computational Thinking: You should be able to derive simple algorithms and code them in Python.

Please let me know if you feel that your background in any of these areas is insufficient. There will be opportunities for getting everyone on track.

Related classes at NTU

In the Division of Mathematics:

- MH4510 Statistical Learning and Data Mining: This course treats a variety of machine learning methods besides deep learning.
- MH4517 Data Applications in Natural Sciences This course covers various methods in geometric and topological data analysis.
- MHXXXX Statistical learning theory: There might be plans for a new course on this topic, which would complement the present one by focussing on the generalization error.

At the School of Computer Science and Engineering:

- CE7454 Deep Learning for Data Science: Mathematical foundations and best engineering practices of modern deep learning.
- CE7455 Deep Learning for Natural Language Processing: From Theory to Practice

Walking the way together

This course offers:

- Beautiful but challenging mathematics: some old and some new, all related to why and how deep learning works, often abstract and sometimes hard to digest.

My role:

- Fetch you where you stand, provide good content, adapt to your feedback, and help you overcome obstacles
- Share my passion and curiosity and learn from you

Your responsibilities:

- Do your homework! Continuous exercise is key, in mathematics as in sport. Repetition forges and consolidates skills.
- Collaborate and help each other! Math is more than numbers.
- Make the best of your time here. Don't shy away from asking questions. Give me feedback.

Workload and assessments

Direct contact:

- Lecture: 3 hours per week
- Tutorial: 1 hour per week

Homework and independent study:

- Homework: 1.5 hours per week (but no mid-term exam)
- Reworking the lecture, further reading: 3 hours per week (according to AU-ETCS conversion guide)

Assessments:

- Written homework: 20% (point-based)
- Oral presentations: 20% (rubric for quality of solution/presentation)
- Written final exam: 60% (point-based)

Homework

Problem sets:

- Problem sets are available on NTULearn around a week ahead of time.

Solutions:

- Before Thursday's lecture, hand in your solution as a scanned `.pdf` or `.ipynb` on NTULearn.
- There, you will also have to indicate what problems you have solved (more on this later).

Collaboration:

- It is recommended that you work in pairs.
- If you do so, name your collaborator and submit individually.

Tutorials

Presentations of solutions:

- For any problem marked as solved in NTULearn, you may be requested to present your solution in the tutorial.
- You will be asked to present at least twice, depending on class size.
- You will be assessed on the quality of your solution and presentation.
- You should be able to explain your solution and the relevant mathematical background.

Contact

During the lectures and tutorials:

- Questions are always welcome; just raise your hand.
- Conversely, I will also ask questions to you.

On NTULearn:

- This is the preferred way of written contact.
- There is a forum for questions and discussions.
- Please answer your colleagues' questions if you know the answer.

Per Email:

- Only as a last resort: `zhongjian.wang@ntu.edu.sg`

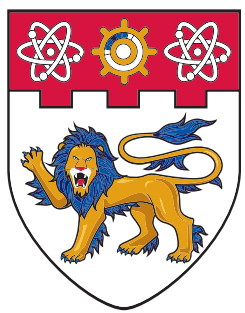
Question to Answer for Yourself / Discuss with Friends

- After the lecture: Find a buddy for solving the homework together. You may also use the forum on NTULearn for this.
- Reflection: What would you, personally, like to learn in this course?
- Reflection: How much time can you set aside for homework and independent study? Try to integrate this into your weekly schedule.

MH3520:Mathematics of Deep Learning

Chapter 1

Introduction to deep learning



NANYANG TECHNOLOGICAL UNIVERSITY

SINGAPORE

Overview of Chapter 1

1. Deep learning in the news
2. Brief history of deep learning
3. Multilayer Perceptrons
4. Deep learning as a way of programming (Optional)
5. Deep learning as representation learning
6. Towards a mathematical theory of deep learning
7. Classification by Machine Learning (Optional)

Sources for this chapter:

- Hutter and Boedecker: Course on Deep Learning. University of Freiburg, Germany.
- Goodfellow, Bengio, and Courville: Deep learning. MIT Press, 2016.

MH3520 Chapter 1

Part 1
Deep learning in the news

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

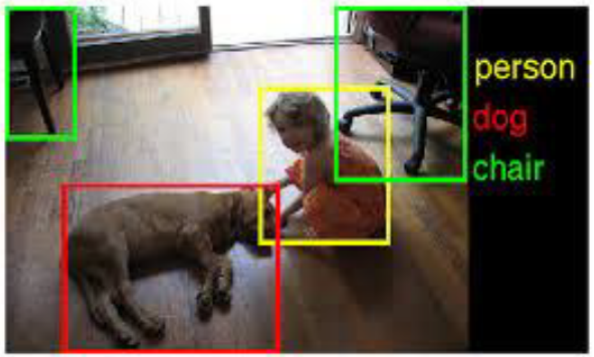
Deep Learning

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

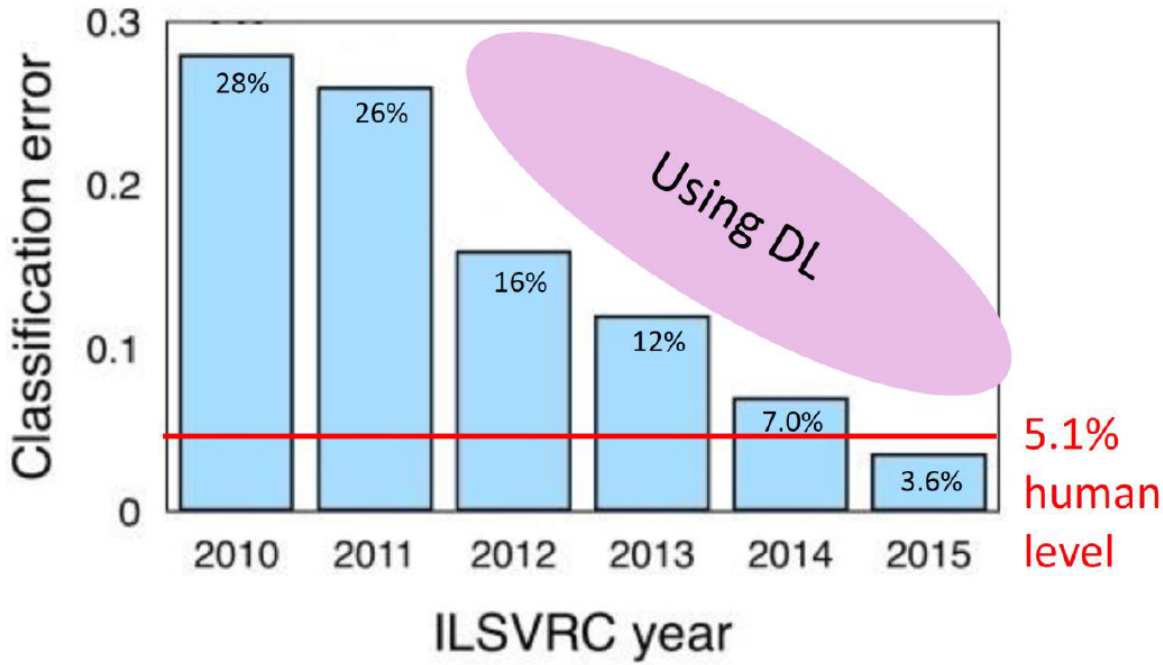
Deep learning revolutionized computer vision

Excellent empirical results via convolutional networks:

Object recognition



Self-driving cars

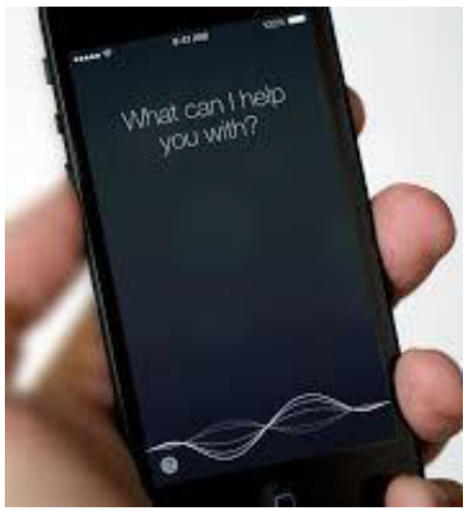


ILSVRC: ImageNet Large-Scale Visual Recognition Challenge

Deep learning revolutionized speech recognition

Excellent empirical results using LSTM and attention-based networks:

Speech recognition



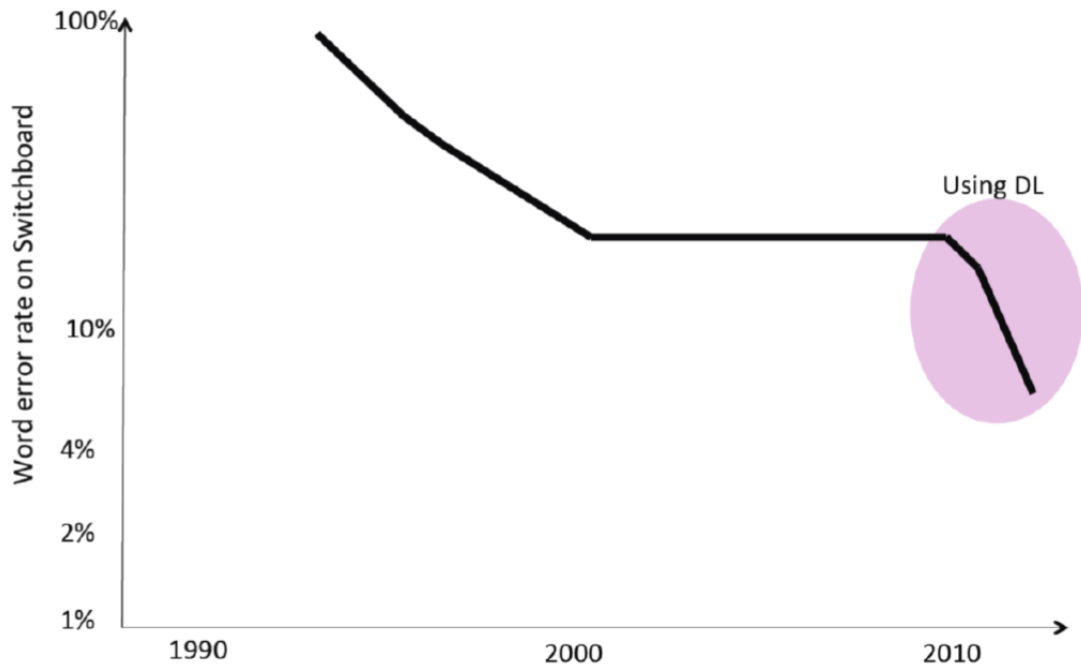


Image credit: Yoshua Bengio (data from Microsoft speech group)

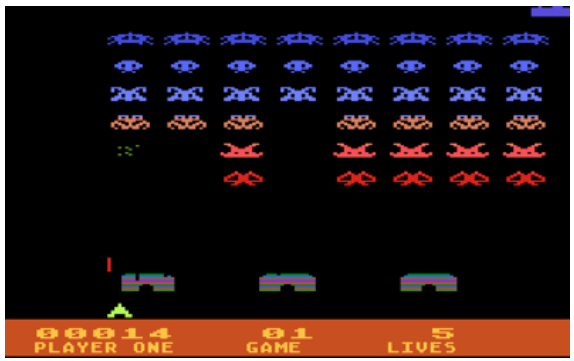
Auto-Translator

Deep learning revolutionized optimal control

Excellent empirical results via 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 in other news

- Failures: Sand dunes mistaken for nudes in software used by the British police. [Telegraph, 18 Dec. 2017]
- Adversarial attacks: Manipulated stop signs mistaken for speed limits [Eykholt e.a., 2018]

- Deepfakes: fake news, hoaxes, fake porn, bullying, financial fraud, etc. [Wikipedia, Deepfake]

Some criticism of deep learning

- Greedy: Requires too much data.
- Unreliable: Transfers badly, generalizes badly.
- Opaque: Lacks interpretability.
- Naive: Cannot reason, lacks genuine knowledge and common sense.
- Biased: Reproduces biases inherent in the training data, confounds correlation and causation.
- Unsafe: Prone to adversarial attacks.
- Dangerous: Can be misused.

[Marcus, 2018] [Tsimenidis, 2012]

Questions to answer for yourself / discuss with friends

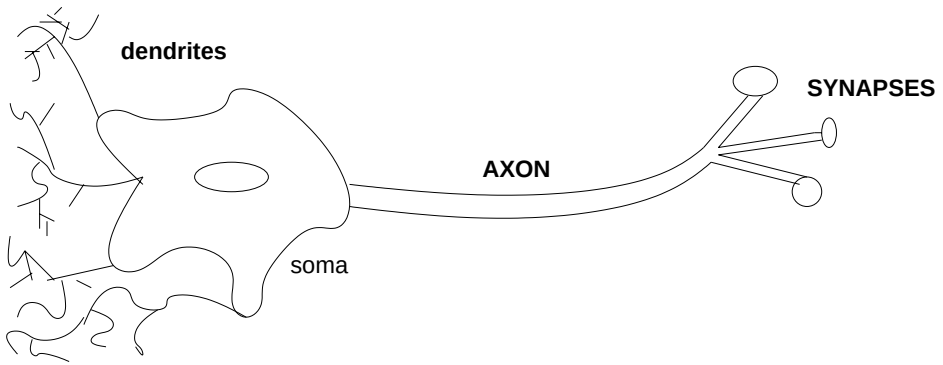
- Repetition: What areas have been revolutionized by deep learning?
- Repetition: Name some actual or potential problems with deep learning.
- Discussion: What other areas have been impacted by deep learning, for good or bad?
- Discussion: How would you describe the difference between artificial intelligence, machine learning, and deep learning?

MH3520 Chapter 1

Part 2

Brief history of deep learning

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: first wave

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: second wave

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 Long short-term memory (LSTM) networks [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., support vector machines (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: third wave

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

- Repetition: For each of the ups and downs in the history of deep learning, describe why it happened.

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

MH3520 Chapter 1

Part 3

Multilayer Perceptrons

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 composition

$$f = \rho \circ T : \mathbb{R}^d \to \mathbb{R},$$

where $T : \mathbb{R}^d \to \mathbb{R}$ is an affine function and $\rho = \mathbb{1}_{\mathbb{R}_+} : \mathbb{R} \to \mathbb{R}$.

- Recall that an affine function is a linear plus a constant function: $T(x) = Ax + b$ for some weights $A \in L(\mathbb{R}^d, \mathbb{R})$ and some bias $b \in \mathbb{R}$.
- ρ is called activation function. The neuron is said to fire if ρ returns 1, i.e., iff Ax exceeds the threshold $-b$.
- The specific activation function of McCulloch and Pitts is rarely used today.

Multi-layer perceptron

A multi-layer 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

A multi-layer perceptron with L layers and activation function $\rho\colon \mathbb{R} \to \mathbb{R}$ is a function

$$f\colon \mathbb{R}^{N_0} \to \mathbb{R}^{N_L}, \qquad F = T_L \circ \rho \circ T_{L-1} \circ \cdots \circ \rho \circ T_1,$$

where for each $l \in \{1, \ldots, L\}$, N_l is a natural number, $T_l\colon \mathbb{R}^{N_{l-1}} \to \mathbb{R}^{N_l}$ is an affine function, and ρ is applied coordinate-wise.

Remark. We will reserve the name neural network for the parameters of the multi-layer perceptron; see later. This distinction is not commonly made in the literature, and we won't be overly strict about it.

Structure of multilayer perceptrons

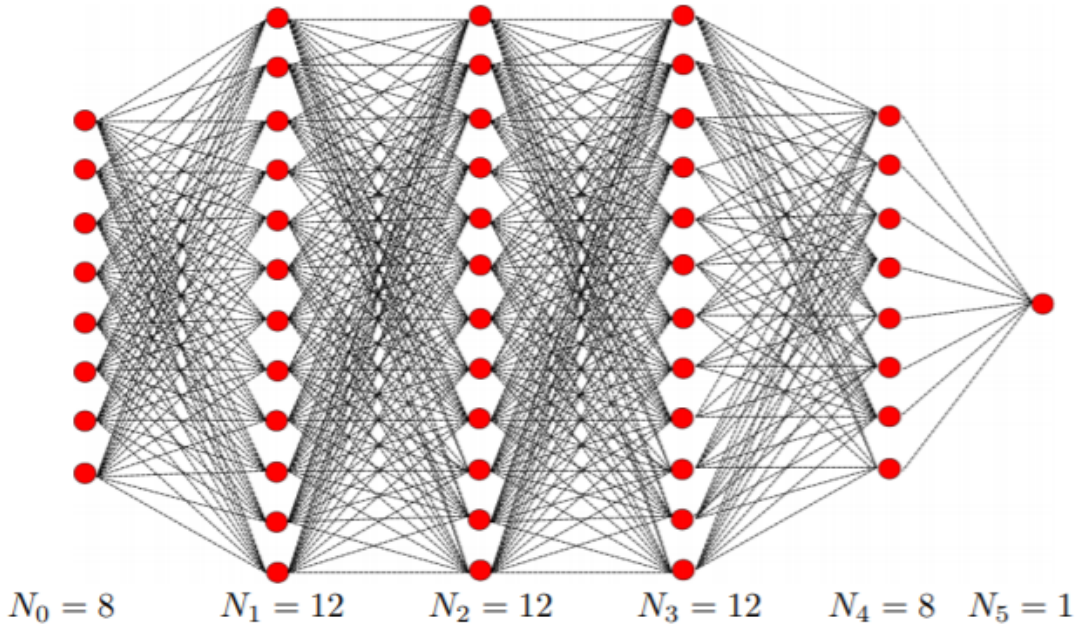


Illustration of a multi-layer perceptron with 5 layers. The red dots correspond to the neurons.

- Multi-layer perceptrons do not allow arbitrary connections between neurons, but only between those in adjacent layers, and only from lower layers to higher layers.
- All layers share the same activation function. This can be relaxed.

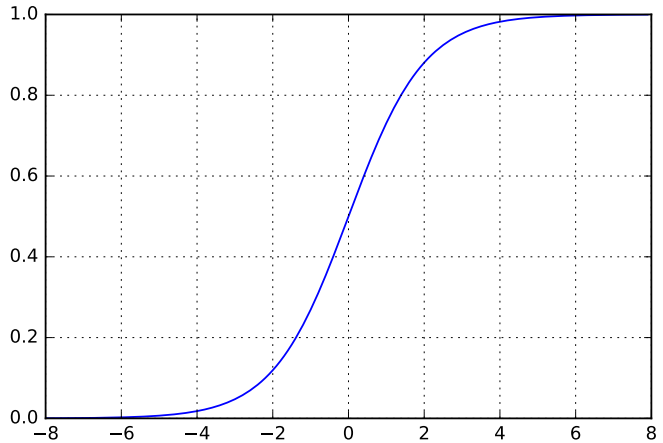
[figure from Petersen, Ch. 1]

Examples of activation functions

- Logistic (sigmoid) function:

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

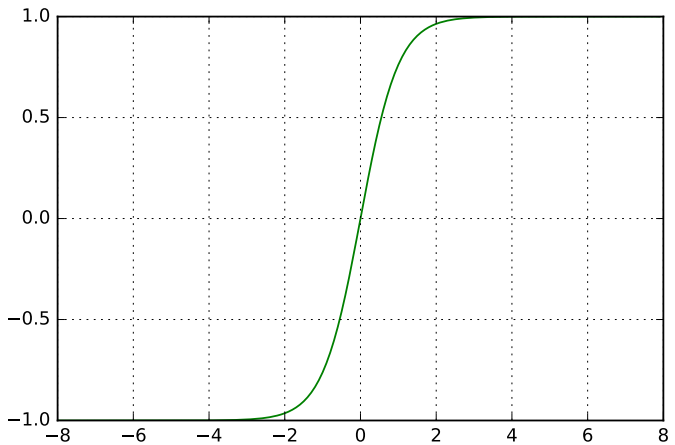
Used in top layer for binary classification.



- Hyperbolic tangent:

$$\begin{aligned}\rho(z) &= \tanh(z) \\ &= \frac{\exp(z) - \exp(-z)}{\exp(z) + \exp(-z)}\end{aligned}$$

Used in top layer for binary classification.

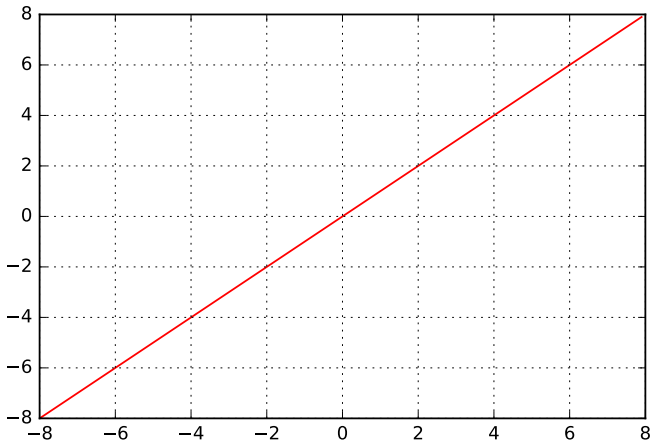


Examples of activation functions (cont.)

- Linear function:

$$\rho(z) = z$$

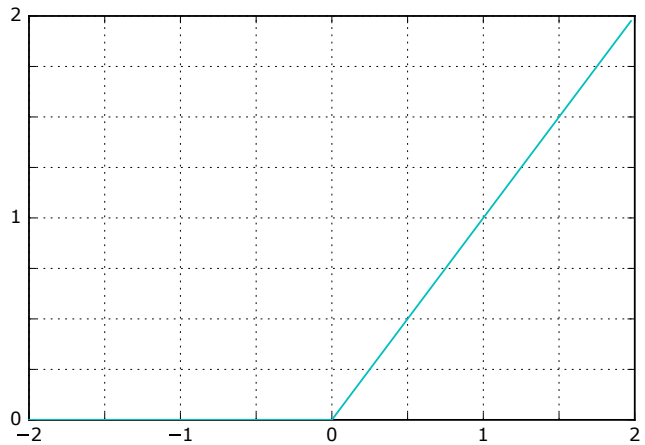
Used in top layer for (generalized) linear regression.



- Rectified linear unit (ReLU):

$$\rho = \max(0, z)$$

Used in all layers for general purpose.



Deep learning

- Deep learning is the use of multi-layer perceptrons in learning tasks.
- We will focus on supervised learning: given input-output pairs (x_i, y_i), the goal is to find a function f such that $f(x_i)$ is close to y_i.
- A loss function ℓ is used to quantify the meaning of proximity.

Definition (Supervised deep learning)

Given input-output pairs $(x_1, y_1), \ldots, (x_n, y_n)$, the task is to find a multi-layer perceptron f which minimizes $\sum_i \ell(f(x_i), y_i)$.

Loss functions

The choice of loss function depends on what kind of data is represented by $\hat{y} = f(x)$ and y:

- Quadratic loss when $\hat{y}, y$ belong to a normed vector space:

$$\ell(\hat{y}, y) = \|\hat{y} - y\|^2.$$

- Binary cross-entropy loss when $\hat{y}, y \in [0, 1]$ are probability densities on the set $\{0, 1\}$ of binary outcomes:

$$\ell(\hat{y}, y) = -\big(y \log \hat{y} + (1 - y) \log(1 - \hat{y})\big).$$

- Categorical cross-entropy loss when $\hat{y}, y$ are probability densities on a finite set $\{1, \ldots, k\}$:

$$\ell(\hat{y}, y) = -\sum_{j=1}^{k} y_j \log \hat{y}_j.$$

Questions to answer for yourself / discuss with friends

- Repetition: What is a multi-layer perceptron?
- Repetition: What kind of activation and loss functions are commonly used?
- Discussion: Can you think of any network architectures which do not satisfy our definition of multi-layer perceptrons?

MH3520 Chapter 1

Part 4

Deep learning as a way of programming (Optional)

Understanding versus learning from experience

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 deep-learn these tasks from data/experience:

- If the task changes, we simply re-train.

The resulting systems may be too complex for us to understand:

- For example, deep neural networks such as AlphaGo have millions of weights.
- David Silver, lead author of AlphaGo cannot explain its moves.
- Paraphrased: "You would have to ask a Go expert."

Deep learning as a way of programming

It is very quick to get good results for some problems:

- Deep learning can handle raw data directly (images, speech, text, etc).
- Well-engineered libraries such as Tensorflow or Pytorch handle the complex underpinnings.
- Very little machine learning knowledge is required to get started.

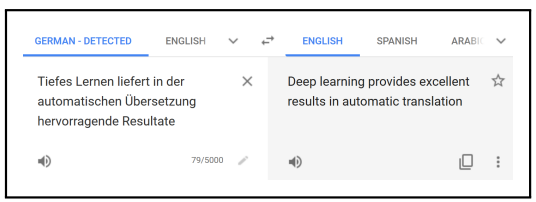
Deep learning is widely applicable:

- Neural networks are very flexible models—this is the main content of the lecture.
- The same general techniques apply to many types of structured data (images, speech, text, etc).
- There is an amazing potential for cross-fertilization.
- Fields that drifted apart for decades have converged again: computer vision, speech recognition, natural language processing, robotics, ...

Wide applicability of deep learning: example tasks

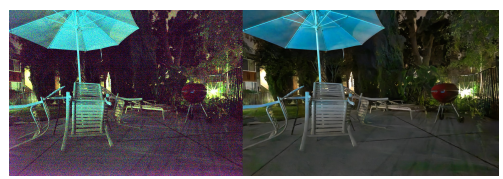
- Classification: input $\mathbb{R}^n \rightarrow$ categories (e.g., {cat, dog, human})
- Regression: input $\mathbb{R}^n \rightarrow \mathbb{R}$ (e.g., house prizes)
- Structured output prediction: structured output (vector, graph, etc)
  - Pixel-wise segmentation: image $\rightarrow$ image
  - Machine translation: sequence $\rightarrow$ sequence
  - Parsing: text $\rightarrow$ tree whose nodes are verbs, nouns, adverbs, etc
  - Image captioning: image $\rightarrow$ sentence
  - Denoising: corrupted example $\tilde{x} \in \mathbb{R}^n \rightarrow$ clean example $x \in \mathbb{R}^n$
  - Generation: sentence $\rightarrow$ image, text

[Valada et al, 2018]



[Google Translate]

[T. Chen et al, 2017]



[C. Chen et al, 2018]

Question to answer for yourself / discuss with friends

- Repetition: From a programming perspective, what are some advantages and disadvantages of deep learning?
- Discussion: Are you aware of any data types that do not easily lend themselves to deep learning?
- Discussion: Do you agree with the distinction of understanding versus learning from experience?

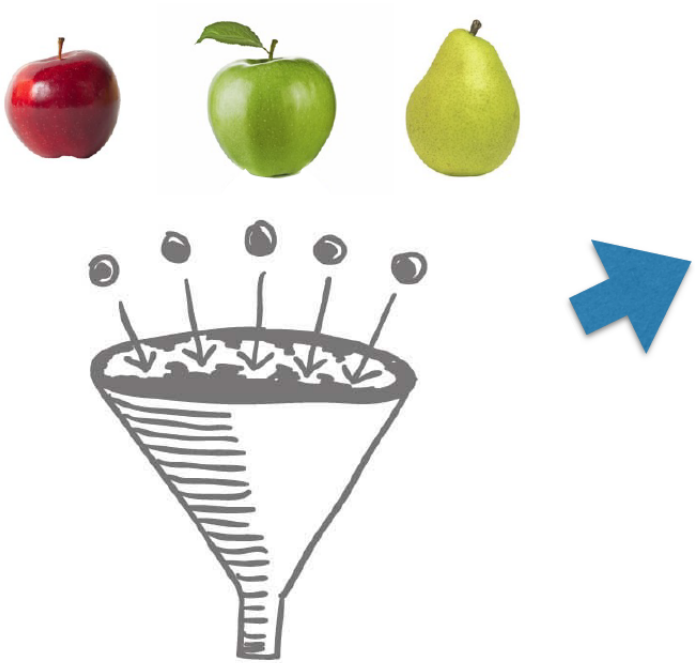
MH3520 Chapter 1

Part 5

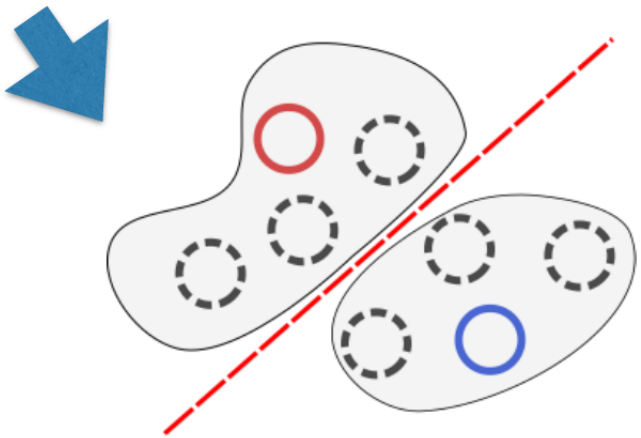
Deep learning as representation learning

Feature engineering

- Standard machine learning algorithms are based on high-level attributes or features of the data
- E.g., the binary attributes we would use for decisions trees
- This requires (often substantial) feature engineering

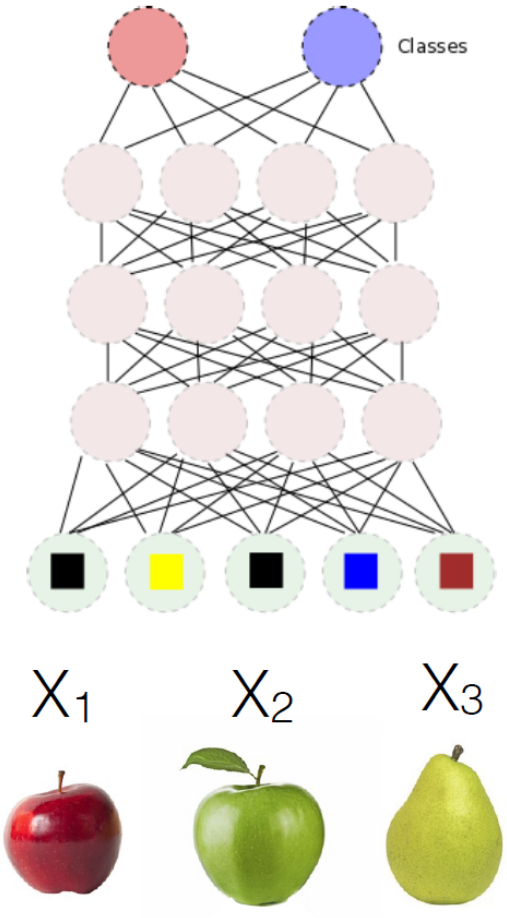


Features		Label
red	3.5 cm	apple
green	4 cm	apple
yellow	5cm	pear

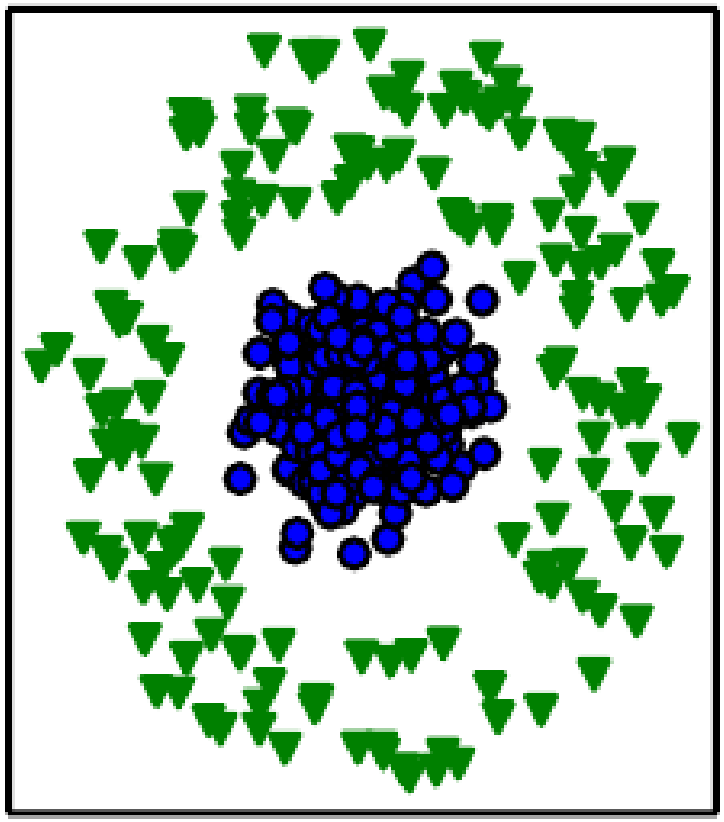


Representation learning

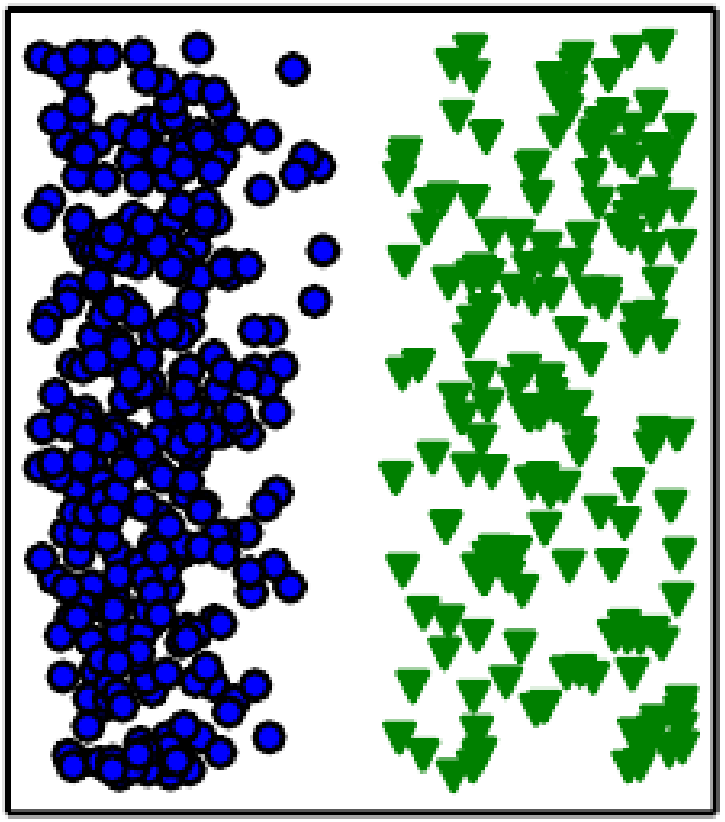
"a set of methods that allows a machine to be fed with raw data and to automatically discover the representations needed for detection or classification" [LeCun et al., 2015]



Example: representations via change of coordinates



Euclidean coordinates
Not linearly separable



Polar coordinates
Linearly separable

[Figure from Goodfellow e.a., p. 4, 2016]

Example: representations of natural numbers

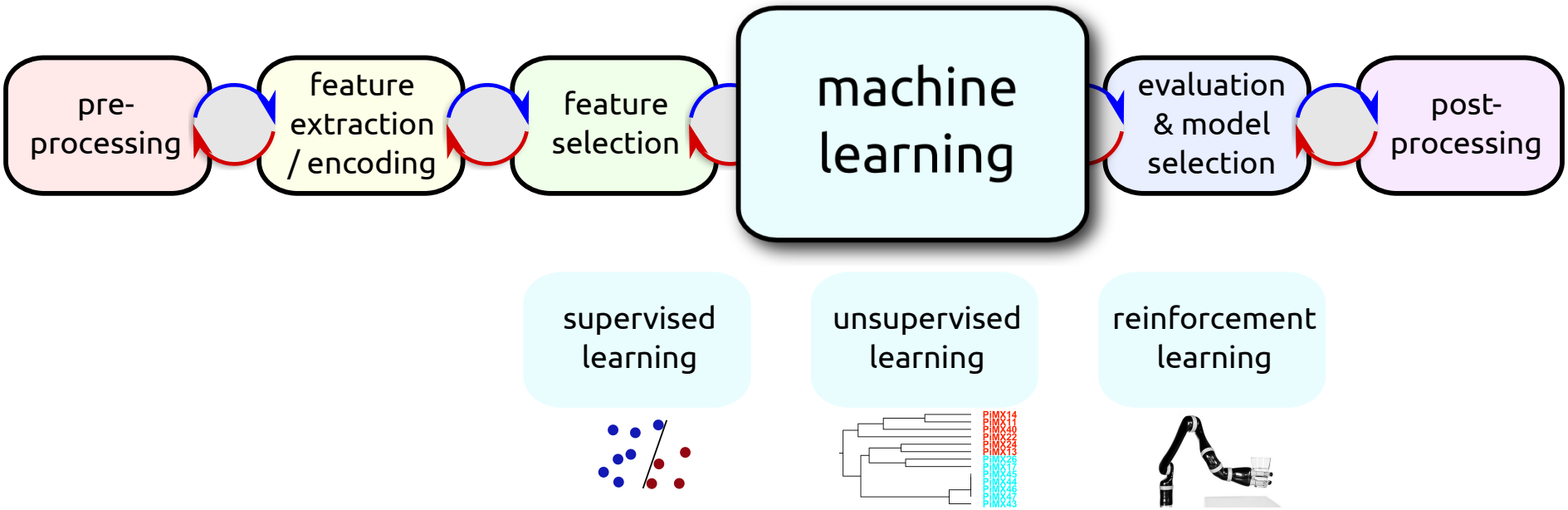
The Roman number representation is poor for the task of addition.
E.g., perform CCCLXIX + DCCCXLV (369 + 845)

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: $369 + 845 = 1214$.

Machine learning design cycle

The classical machine learning design cycle consists of:



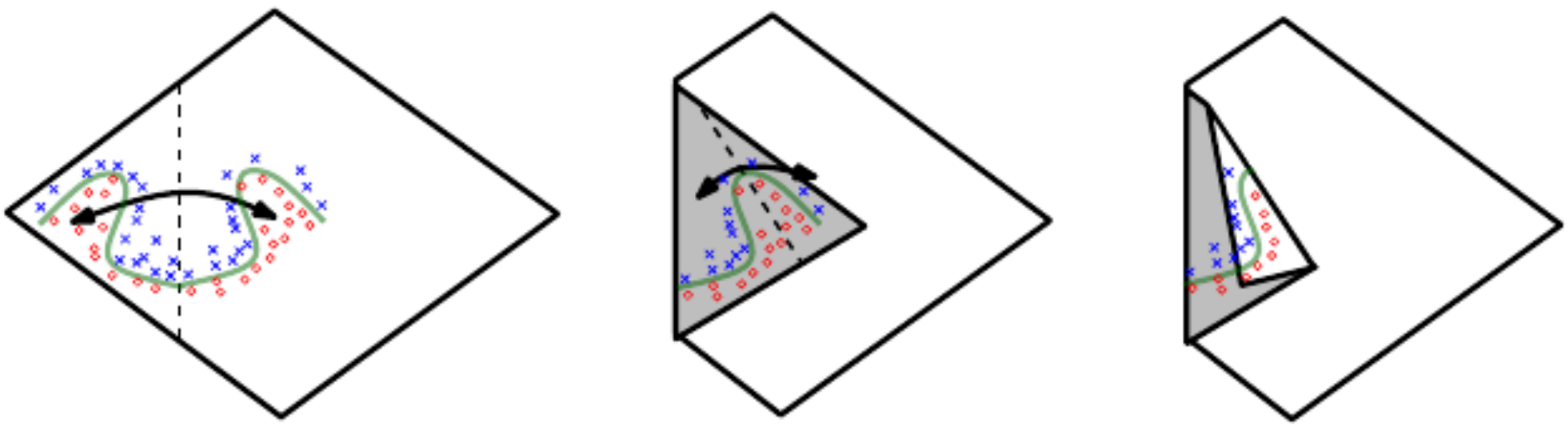
Deep learning facilitates end-to-end learning:

- The individual steps are replaced by a single optimization problem.
- There is far less pre-processing and no manual feature engineering (extraction & selection).
- The useful features (or data representations) are learned automatically.

Deep learning as multi-level representation learning

Deep learning can be understood as:

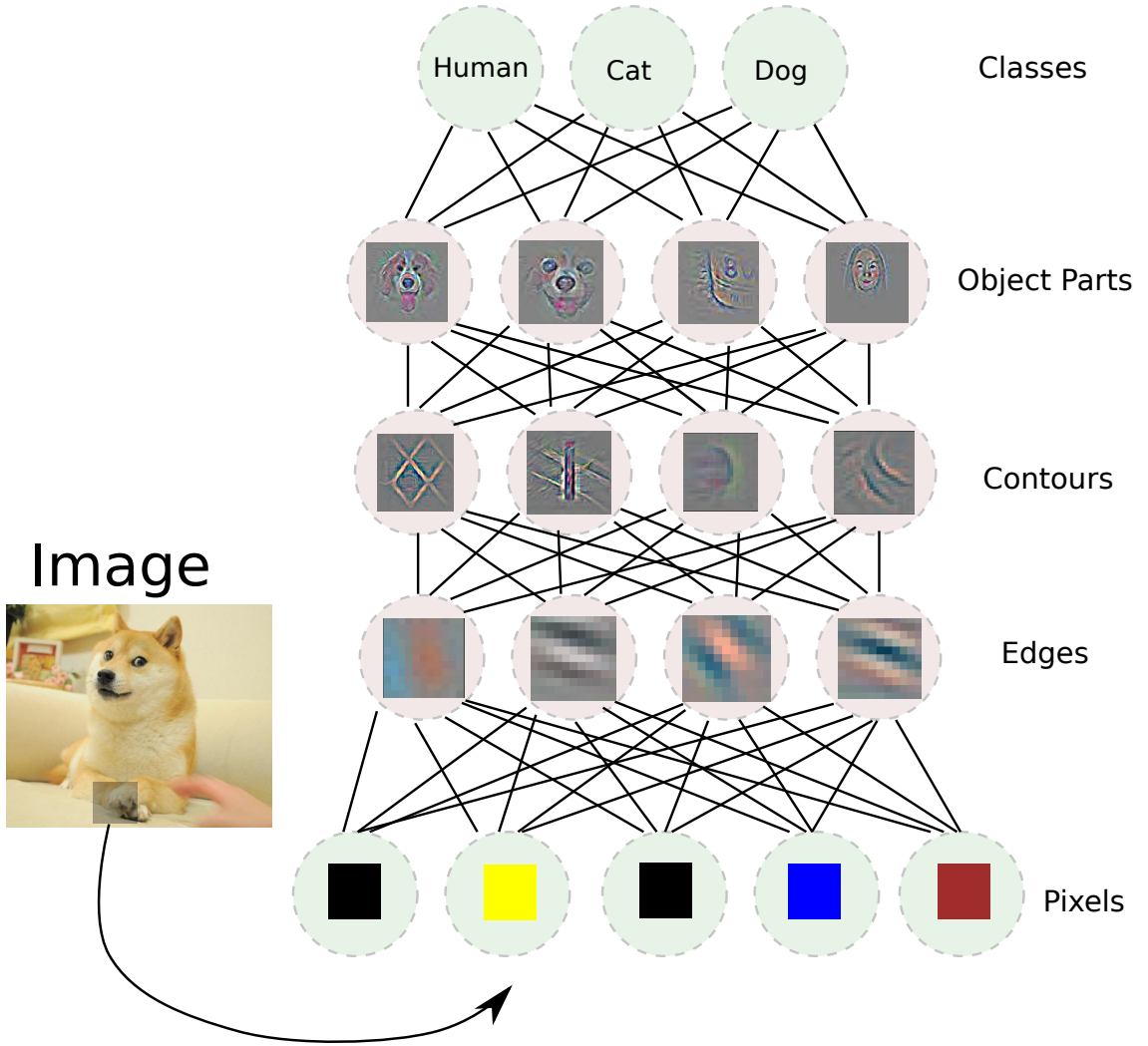
> *"representation learning methods with multiple levels of representation, obtained by composing simple but nonlinear modules that each transform the representation at one level into a [...] higher, slightly more abstract (one)"* *[LeCun et al., 2015]*



[Figure from Goodfellow e.a., p. 194, 2016]

Example: image representations learned by deep networks

A hierarchy of representations, from simple to complex, learned in the layers of a trained neural network.



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

Questions to answer for yourself / discuss with friends

- Repetition: What is representation learning, and how does it relate to deep learning?
- Repetition: Give some examples of data representations.
- Discussion: The visualization of features learned by image nets is a non-trivial task—can you guess how it is done?
- Discussion: Are deep networks always better than shallow ones?

MH3520 Chapter 1

Part 6

Towards a mathematical theory of deep learning

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:

- As Mathematicians, we are ideally prepared for appreciating the abstract issues involved in high-dimensional data analysis [Donoho]

Unreasonable effectiveness of deep learning

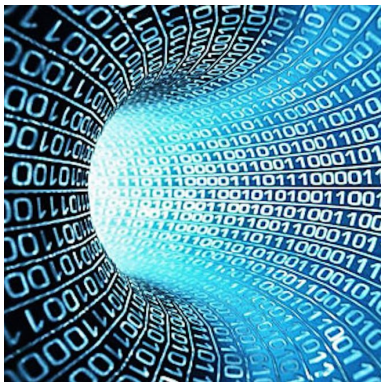
Deep learning performs way better than predicted by theory:

> *Although applications of deep learning networks to real-world problems have become ubiquitous, our understanding of why they are so effective is lacking.*
>
> *These empirical results should not be possible according to sample complexity in statistics and nonconvex optimization theory.*
>
> *[Sejnowski 2020]*

Empirically, it's a miracle that deep learning works at all:

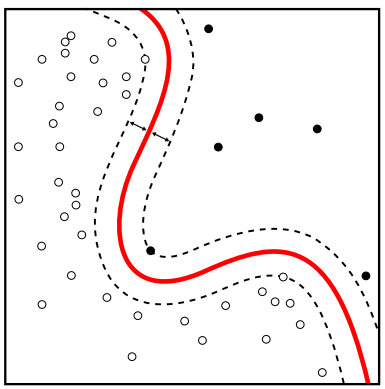
- Generic nonconvex optimization problems with millions of variables are impossible to solve.
- Generic statistical estimation problems in a million dimensions are impossible to solve.

Abstract issues in complex learning tasks



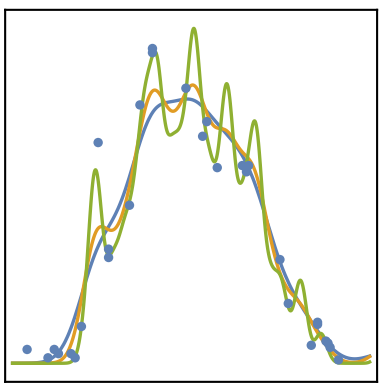
High-dimensionality:

- More and more pixels, voxels, frames per second, etc
- More and more samples (not always)
- Larger and larger networks.



Nonlinearity:

- Nonlinear input-output relations in image recognition, language translation, robotics, etc.
- Tellingly, also in neural networks.



Randomness:

- Inherent in noisy data.
- Used in network initialization and training.

Mathematical ideas and tools

High-dimensionality $\leftrightsquigarrow$ asymptotic methods:

- The curse of dimensionality refers to certain quantities increasing exponentially in the dimension (e.g. the number of points on a regular grid).
- The blessing is that asymptotic methods sometimes allow one to pass to infinite-dimensional settings, which are easier to work with.

Nonlinearity $\leftrightsquigarrow$ geometry:

- For instance, the input data may belong to a low-dimensional manifold, i.e., a nice geometric space.
- Neural networks with a given architecture also form a nonlinear space, which can be described geometrically.

Randomness $\leftrightsquigarrow$ probability and statistics:

- Statistical learning theory provides an error analysis for supervised learning with noisy data.
- Probability theory describes concentration of measure phenomena in high dimensions or limits of infinitely wide or deep random networks.

Coverage in this course

Asymptotic methods will loom large:

- Neural networks shall be seen as elements of an infinite-dimensional function space.
- Approximation theory quantifies how well a given function can be approximated by neural networks.
- (Reading) Coding theory gives information-theoretic bounds on how good the approximation can be.

Geometry and probability will appear only marginally:

- Gradient descent (as e.g. in training neural networks) is a geometric notion, but we will view it foremost in the context of numerical optimization.
- We will encounter some probabilistic methods in statistical learning theory.

Questions to answer for yourself / discuss with friends

- Repetition: What mathematical theories can be used to describe and analyze deep learning?
- Discussion: Are there aspects of deep learning that are *not* captured by these mathematical theories?

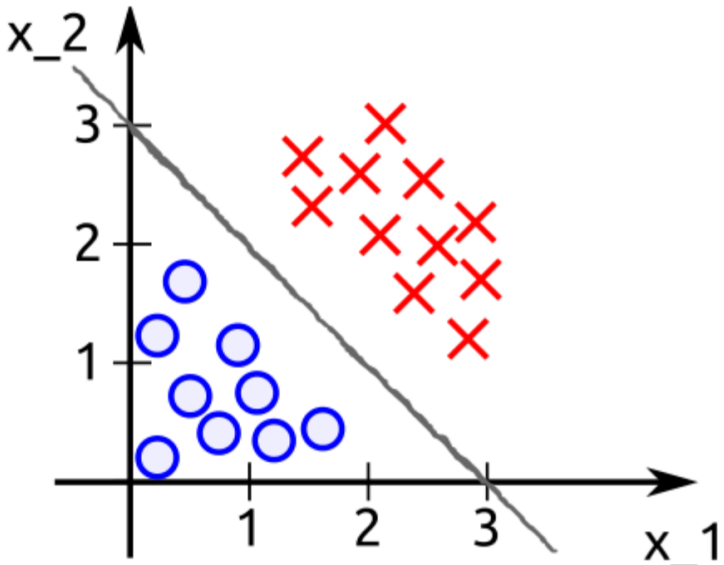
MH3520 Chapter 1

Part 7

Classification by Machine Learning (Optional)

Binary classification

- We want to assign labels 0 or 1 to our observations
- We start simple and use an affine function as decision boundary



$$w_0 + w_1 x_1 + w_2 x_2 = 0$$

$$\mathbf{w} = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

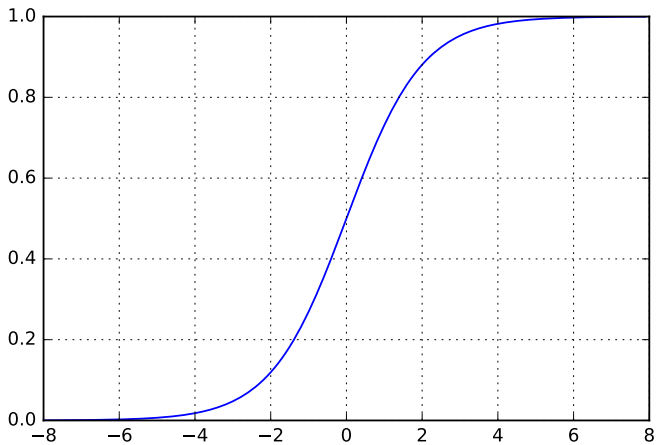
- This is a linear function if we declare $x_0 = 1$
- We classify as "$y = 1$" if $\mathbf{w}^\top \mathbf{x} \geq 0$
- This is linear regression model with basis functions $\{1, x_1, x_2\}$, followed by a binary classifier $\mathbb{1}_{\mathbb{R}_+}$.

Logistic regression

- Instead of a binary outcome, logistic regression yields a probabilistic estimate of how likely a data point $\mathbf{x}$ belongs to class 1.
- Namely, the probability that $\mathbf{x}$ belongs to class 1 is defined as

$$h_{\mathbf{w}}(\mathbf{x}) := \rho(\mathbf{w}^{\mathsf{T}}\mathbf{x}).$$

- Here, ρ is the logistic function $\rho(z) = \frac{1}{1+e^{-z}}$.



- We're maximally uncertain about points on the decision boundary

$$\text{E.g., } \mathbf{w}^{\mathsf{T}}\mathbf{x} = 0 \quad \Leftrightarrow \quad h_{\mathbf{w}}(\mathbf{x}) = 0.5$$
$$\text{E.g., } \mathbf{w}^{\mathsf{T}}\mathbf{x} = 5 \quad \Leftrightarrow \quad h_{\mathbf{w}}(\mathbf{x}) = 0.993$$

Likelihood of predicting the correct label

- Logistic regression asserts that the label of $\mathbf{x}$ is a random variable Y with Bernoulli distribution

$$p_{\text{model}}(Y = 1 \mid \mathbf{x}; \mathbf{w}) = h_{\mathbf{w}}(\mathbf{x}).$$

- In supervised learning, we observe the true label ($y = 0$ or $y = 1$) from the data.
- The likelihood of observing the true label is

$$p_{\text{model}}(Y = y \mid \mathbf{x}; \mathbf{w}) = \begin{cases} h_{\mathbf{w}}(\mathbf{x}) & \text{for} \quad y = 1 \\ 1 - h_{\mathbf{w}}(\mathbf{x}) & \text{for} \quad y = 0 \end{cases}$$

- Cross-entropy loss is the negative logarithm of this likelihood:

$$\ell(h_{\mathbf{w}}(\mathbf{x}), y) = \begin{cases} -\log(h_{\mathbf{w}}(\mathbf{x})) & \text{for} \quad y = 1 \\ -\log(1 - h_{\mathbf{w}}(\mathbf{x})) & \text{for} \quad y = 0 \end{cases}$$

Cross-entropy loss

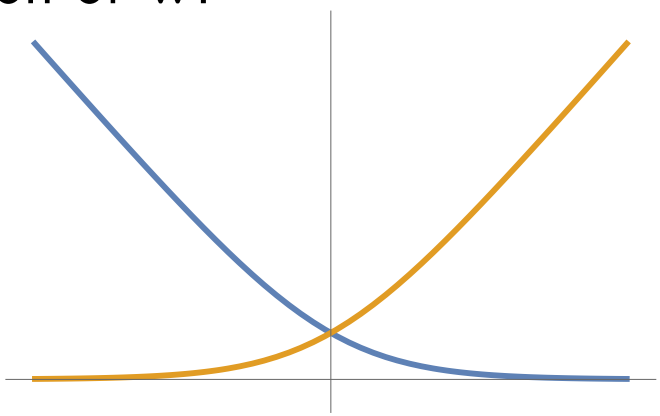
- Likelihoods of independent events are multiplied, and logarithmic likelihoods are added up.
- Therefore, the loss for the entire dataset is the sum of the individual losses:

$$\sum_i \ell(h_{\mathbf{w}}(\mathbf{x}_i), y_i)$$

- Each individual loss is a convex function of $\mathbf{w}$:

$$y = 0: \qquad -\log\left(\frac{1}{1+e^{-z}}\right)$$

$$y = 1: \qquad -\log\left(1 - \frac{1}{1+e^{-z}}\right)$$

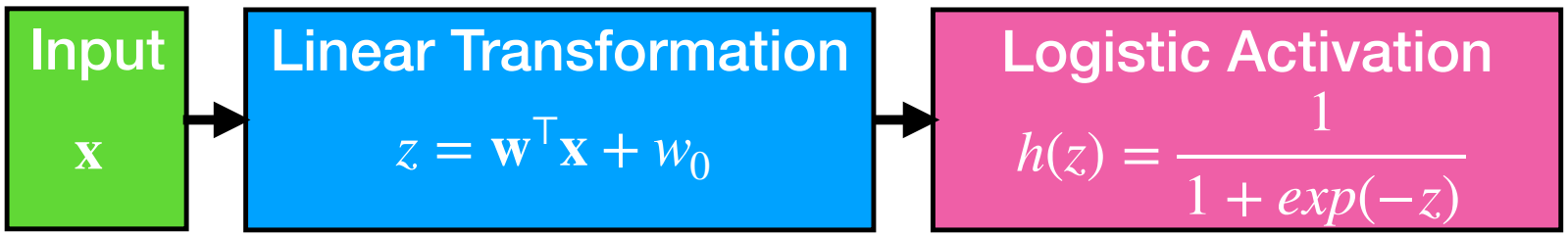


- The aggregated loss is again convex and can be minimized efficiently.
- Then, the resulting logistic regression model can be used to predict the labels of yet unseen data.

Example of logistic regression

$$h_{\mathbf{w}}(\mathbf{x}) = \frac{1}{1 + exp(-(\mathbf{w}^{\top}\mathbf{x} + w_0))} = \frac{1}{1 + exp(-z)}$$

Logistic Regression



$x = [x_1 \quad x_2 \quad x_3 \quad x_4]$

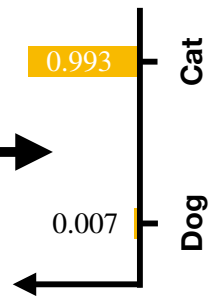
$= [2 \quad 0.25 \quad 1.5 \quad 5]$

$$z = \mathbf{w}^{\top}\mathbf{x} + w_0 = \begin{bmatrix} 0 \\ 4.0 \\ 9.0 \\ -2 \end{bmatrix} \cdot [2.0 \quad 0.25 \quad 1.5 \quad 5] + 0.5$$

$$= 2 * 0 + 0.25 * 4 + 1.5 * 9 - 2 * 5 + 0.5$$

$$= 5$$

$$h = \frac{1}{1 + exp(-z)} = \frac{1}{1 + exp(-5)} = 0.993$$



Prediction

Example of logistic regression

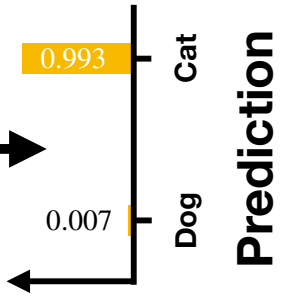
$$x = [x_1 \quad x_2 \quad x_3 \quad x_4] = [2 \quad 0.25 \quad 1.5 \quad 5]$$

$$z = \mathbf{w}^\top \mathbf{x} + w_0 = \begin{bmatrix} 0 \\ 4.0 \\ 9.0 \\ -2 \end{bmatrix} \cdot [2.0 \quad 0.25 \quad 1.5 \quad 5] + 0.5$$

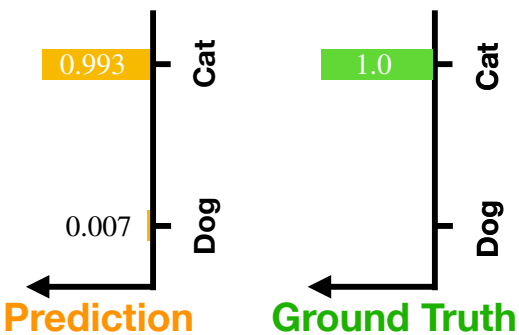
$$= 2 * 0 + 0.25 * 4 + 1.5 * 9 - 2 * 5 + 0.5$$

$$= 5$$

$$h = \frac{1}{1 + exp(-z)} = \frac{1}{1 + exp(-5)} = 0.993$$



$$Cost(h_{\mathbf{w}}(\mathbf{x}), y) = \begin{cases} -log(h_{\mathbf{w}}(\mathbf{x})) & \textbf{for } y = 1 \\ -log(1 - h_{\mathbf{w}}(\mathbf{x})) & \textbf{for } y = 0 \end{cases}$$



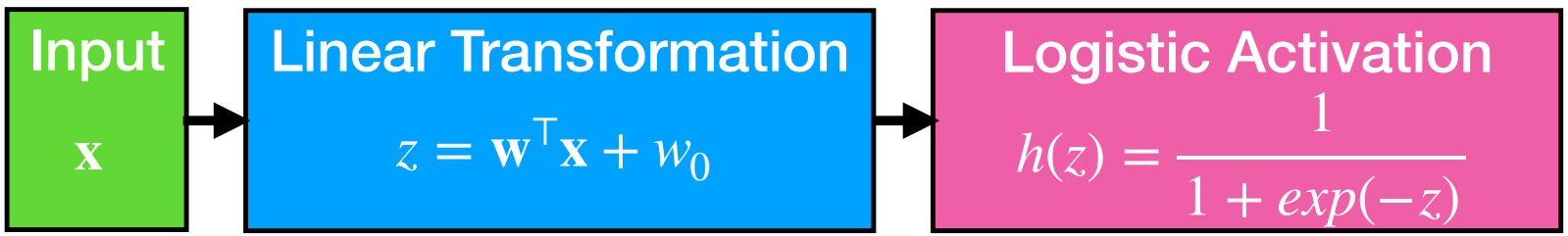
$$Cost(h_{\mathbf{w}}(\mathbf{x}), y) = -log(h_{\mathbf{w}}(\mathbf{x}))$$
$$= -log(0.993)$$
$$= 0.00702$$

Cross Entropy

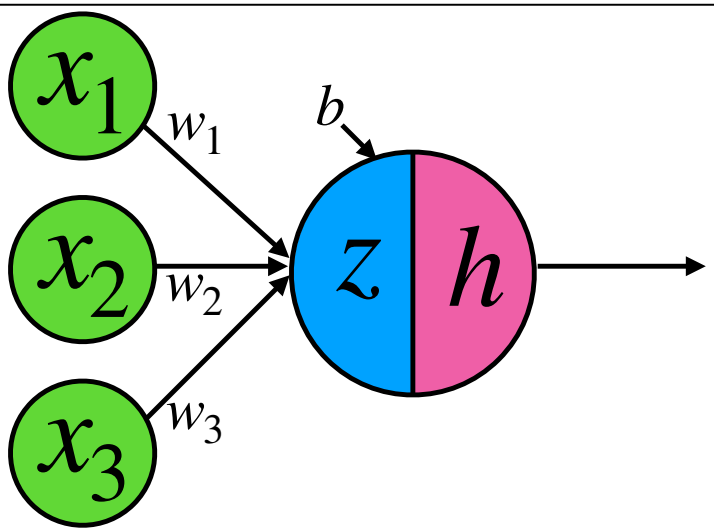
The logistic regression model as a perceptron

$$h_{\mathbf{w}}(\mathbf{x}) = \frac{1}{1 + exp(-(\mathbf{w}^\top \mathbf{x} + w_0))} = \frac{1}{1 + exp(-z)}$$

Logistic Regression



Perceptron



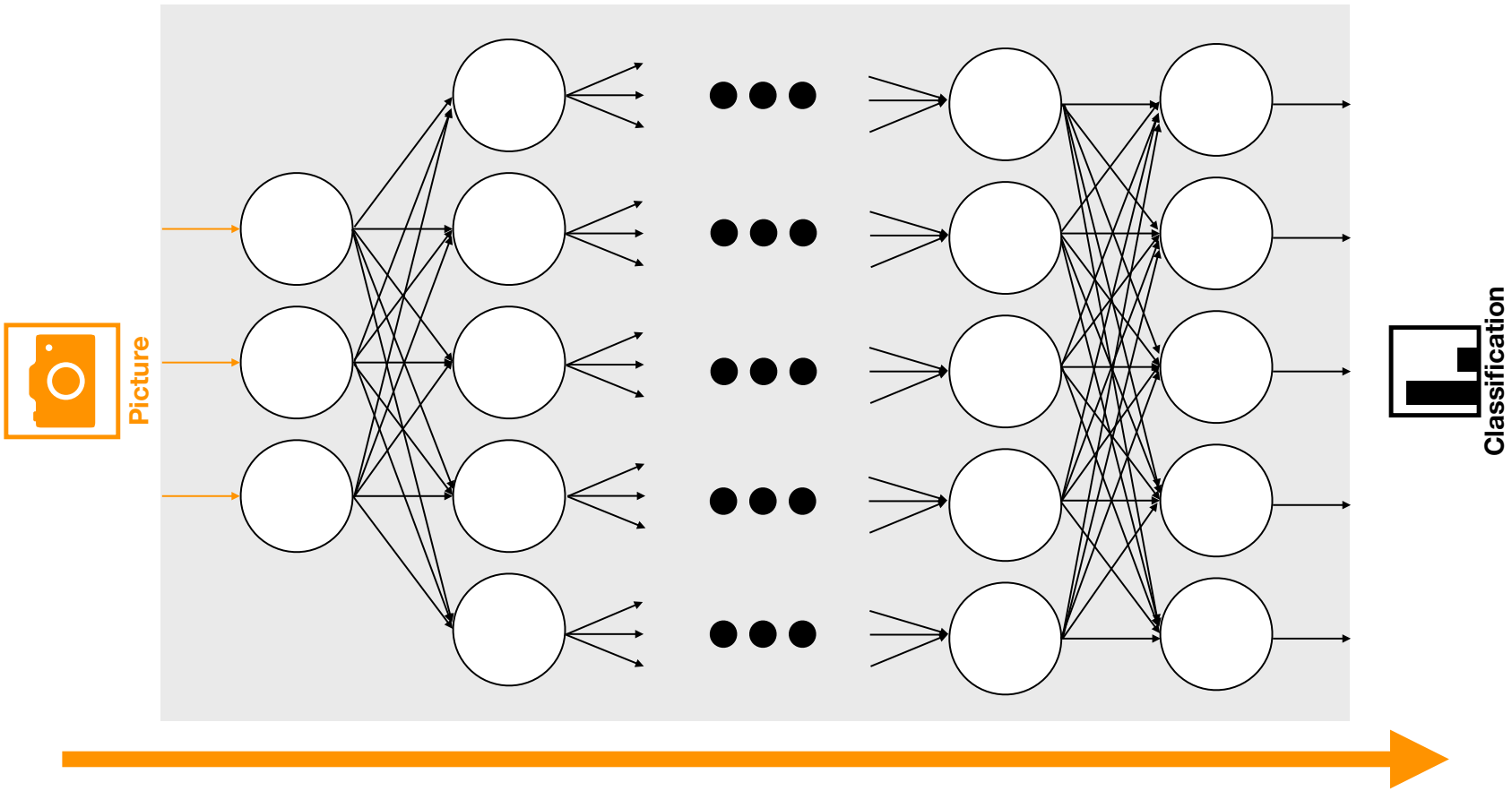
$$z = \mathbf{w}^\top \mathbf{x} + b$$

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

Logistic regression with multi-layer perceptrons

- We now add hidden layers, which learn nonlinear features for the final logistic regression.
- Computation is performed layer-by-layer.
- As before, the top layer has a logistic activation function.
- The bottom layers can have other activation functions.

Information flow through the multi-layer perceptron



[Hutter and Boedecker]

Logistic regression with multiple labels

- For multi-class classification with K classes, use K output neurons.
- Then, the probability of belonging to class $k \in \{1, \ldots, K\}$ is defined as the softmax function

$$p_{\text{model}}(Y = k \mid \mathbf{x}, \mathbf{w}) := (\hat{\mathbf{y}}_\mathbf{1}, \ldots, \hat{\mathbf{y}}_\mathbf{k}) := \frac{\exp(\mathbf{z_k})}{\sum_\mathbf{j} \exp(\mathbf{z_j})},$$

  where $\mathbf{z}$ is the output of the perceptron with input $\mathbf{x}$ and coefficients $\mathbf{w}$.
- Thus, the top layer has a softmax activation function, whereas the lower layers may have other activation functions.
- A suitable loss function is the categorical cross-entropy

$$\ell(\hat{y}, y) = -\sum_{j=1}^{k} y_j \log \hat{y}_j,$$

  where $\hat{y}_j$ are the predicted probabilities and y_j are the true labels in one-hot encoding.

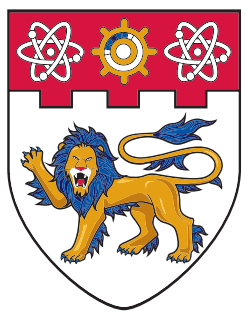
Question to Answer for Yourself / Discuss with Friends

- Repetition: What is logistic regression, and what does it have to do with single- or multi-layer perceptrons?
- Application of what you just learned: What is the computational complexity of computing the cross-entropy loss for d-dimensional input, K-dimensional output, and N data points?

MH3520:Mathematics of Deep Learning

Chapter 2

Basics of numerical optimization



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

Last chapter:

- Deep learning is an optimization problem
- E.g., given input-output pairs $(x_1, y_1), \ldots, (x_n, y_n)$, one has to minimize $\sum_i (y_i - f(x_i))^2$ over all multi-layer perceptrons f.

This chapter:

- Introduction to optimization
- Focus on unconstrained continuous optimization
- Before we get there: recapitulation of differential calculus

Next chapter:

- Application to deep learning problems.

Overview of Chapter 2

1. Differential calculus
2. Mathematical optimization
3. Numerical optimization
4. Gradient flow and gradient descent (Optional)
5. Minimizers of convex objectives (optional)
6. Gradient flow and gradient descent of convex objectives (Optional)

Sources for this chapter:

- Lang: Foundations of Differential Geometry. Springer, 1999.
- Nesterov: Introductory Lectures on Convex Optimization. Kluwwer, 2004.
- Nocedal and Wright: Numerical optimization. 2nd edition. Springer, 2006.

MH3520 Chapter 2

Part 1
Differential calculus

Differential calculus

What is differential calculus?

- Differential calculus refers to a notion of differentiability (e.g. k-fold continuous differentiability) together with some rules on how to obtain new differentiable functions from old ones (e.g. by composition).
- The definitions and results, when written down correctly, are the same for Euclidean spaces as for normed vector spaces.
- Here, 'correctly' means coordinate-free.

Main message:

- You can differentiate with respect to anything that belongs to a normed space.

For the fainthearted:

- If this is too abstract for you or you dislike generality, just think of Euclidean spaces throughout.

Normed vector space

Definition

A norm on a vector space E is a function $\|\cdot\| : E \to [0, \infty)$ such that for all $x, y, z \in E$ and $\lambda \in \mathbb{R}$,

- Non-degeneracy: $\|x\| = 0$ if and only if $x = 0$,
- Absolute homogeneity: $\|\lambda x\| = |\lambda| \|x\|$, and
- Triangle inequality: $\|x + y\| \leq \|x\| + \|y\|$.

The tuple $(E, \|\cdot\|)$ is called a normed vector space.

Example

For any $p \in [1, \infty)$, the p-norm and supremum norm on $\mathbb{R}^d$ are given by

$$\|x\|_p := \left(\sum_{i=1}^{d} |x_i|^p \right)^{1/p}, \qquad \|x\|_\infty := \max_{i \in \{1,\ldots,d\}} |x_i|$$

Convergence in normed spaces

Standing assumption. E, F, and G are normed vector spaces.

Definition

A sequence (x_n) in E converges to a point x in E if

$$\forall \epsilon > 0 \quad \exists N \in \mathbb{N} \quad \forall n > N : \qquad \|x_n - x\| < \epsilon.$$

Symbolically, this is expressed as

$$\lim_{n\to\infty} x_n = x, \qquad \text{or} \qquad x_n \xrightarrow[n\to\infty]{} x.$$

Remark. Convergence can be defined more generally on any topological space. However, the structure of a normed vector space is needed (or at least highly useful) for differential calculus.

Completeness of normed spaces

Definition

A sequence (x_n) in E is called Cauchy sequence if

$$\forall \epsilon > 0 \quad \exists N \in \mathbb{N} \quad \forall m, n > N : \qquad \|x_m - x_n\| < \epsilon.$$

Symbolically, this is expressed as

$$\lim_{m,n\to\infty} x_m - x_n = 0, \qquad \text{or} \qquad x_m - x_n \xrightarrow[m,n\to\infty]{} 0.$$

Definition

A normed vector space is called complete if every Cauchy sequence therein has a limit.

Continuity in normed spaces

Definition

A function $f : E \to F$ is continuous if for any sequence (x_n) in E,

$$x_n \xrightarrow[n\to\infty]{} x \qquad \text{implies} \qquad f(x_n) \xrightarrow[n\to\infty]{} f(x).$$

Remark. In a normed space, addition and scalar multiplication are continuous.

Remark. Again, continuity can be defined more generally for functions between topological spaces.

Linear functions

Definition

$L(E, F)$ denotes the set of continuous linear functions $A : E \to F$ with the operator norm

$$\|A\| = \sup\left\{\|Ax\| : x \in E, \|x\| \leq 1\right\}.$$

Remark.

- Linear functions with finite operator norm are called bounded, and boundedness is equivalent to continuity.
- As an aside, there are other important norms on linear operators; the operator norm is not the only possible choice.

Composition of continuous linear functions

Lemma

Composition is a continuous bilinear operation

$$L(E,F) \times L(F,G) \to L(E,G).$$

Proof. Bilinearity is clear. Continuity follows from

$$\begin{aligned} \|AB - A'B'\| &\leq \|(A-A')B\| + \|A'(B-B')\| \\ &\leq \|A-A'\|\|B\| + \|A'\|\|B-B'\|. \end{aligned}$$

□

Continuous bilinear functions

Definition

$L^2(E, F; G)$ denotes the set of continuous bilinear functions $A : E \times F \to G$ with the operator norm

$$\|A\| = \sup\big\{\|A(x, y)\| : x \in E, y \in F, \|x\| \le 1, \|y\| \le 1\big\}.$$

Lemma

$L^2(E, F; G)$ is isometrically isomorphic to $L\big(E, L(F, G)\big)$.

Proof. Consider $(x, y) \in E \times F$ as free variables. Then $A(x, y)$ belongs to $L^2(E, F; G)$ if and only if $A(x)(y)$ belongs to $L(E, L(F, G))$. Moreover, the operator norms of these functions coincide. □

Remark. Similar statements hold for multilinear maps.

Differentiability

Definition

A function $f : E \to F$ is differentiable at a point $x \in E$ if there exists a linear function $df(x) \in L(E, F)$ such that

$$\lim_{h \to 0} \frac{f(x+h) - f(x) - df(x)h}{\|h\|} = 0.$$

The derivative is denoted by $df(x) = f'(x) = \partial_x f(x) = \frac{d}{dx} f(x)$.

Remark. The remainder

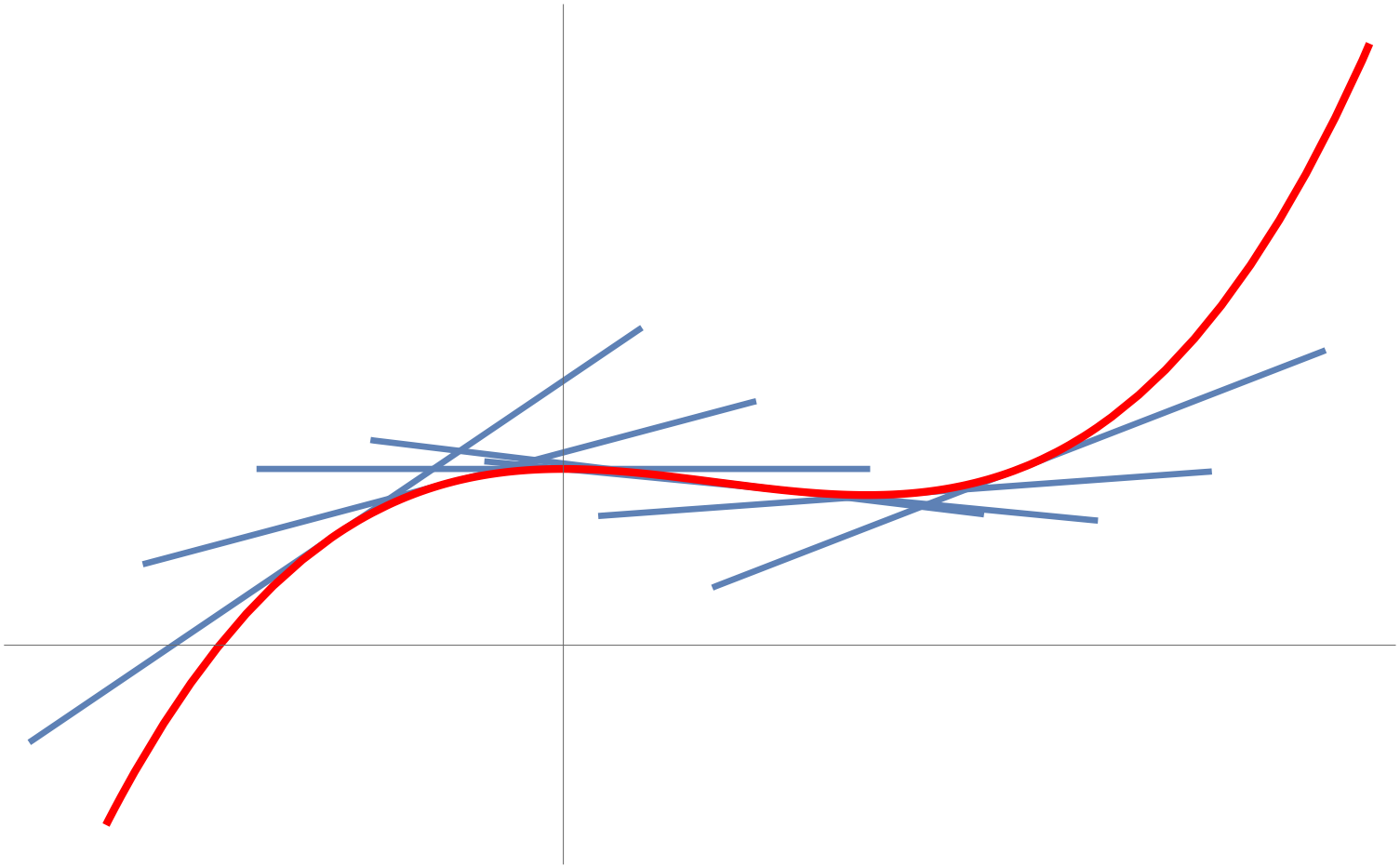
$$R_f(x, h) := f(x+h) - f(x) - df(x)h$$

tends superlinearly to zero, i.e., $\lim_{h \to 0} \|h\|^{-1} R_f(x, h) = 0$.

Remark. Differentiability implies continuity because

$$f(x+h) - f(x) = df(x)h + R_f(x, h) \xrightarrow[h \to 0]{} 0.$$

Derivatives as linearizations



The derivative is a linearization. The remainder (i.e., the difference between the function and its linearization) tends superlinearly to zero

Examples of differentiable functions

Example (Differentiability of linear functions)

If $f \in L(E,F)$ is continuous linear, then $df(x) = f$ for all x because

$$\frac{f(x+h) - f(x) - f(h)}{\|h\|} = 0 \xrightarrow[h\to 0]{} 0.$$

Example (Differentiability of quadratic functions)

If $f(x) = A(x,x)$ for some $A \in L^2(E,E;F)$, then $df(x) = A(x,\cdot) + A(\cdot,x)$ because

$$\frac{A(x+h,x+h) - A(x,x) - A(x,h) - A(h,x)}{\|h\|} = \frac{A(h,h)}{\|h\|} \xrightarrow[h\to 0]{} 0.$$

Chain rule

Lemma (Chain rule)

If $f : E \to F$ is differentiable at $x \in E$ and $g : F \to G$ is differentiable at $y := f(x) \in F$, then $g \circ f : E \to G$ is differentiable at x with derivative

$$d(g \circ f)(x) = dg(y) \circ df(x).$$

Proof. Compute $R_{g\circ f}$ in terms of R_f and R_g: with $k = f(x+h) - f(x)$,

$$\begin{aligned} R_{f\circ g}(x,h) &:= g(f(x+h)) - g(f(x)) - dg(f(x))\, df(x)h \\ &= g(y+k) - g(y) - dg(y)\, df(x)h \\ &= dg(y)k + R_g(y,k) - dg(y)\, df(x)h \\ &= dg(y)R_f(x,h) + R_g(y,k). \end{aligned}$$

As f is differentiable, $\|k\| \lesssim \|h\|$ for small h, and

$$\frac{R_{f\circ g}(h)}{\|h\|} = \frac{dg(y)R_f(x,h)}{\|h\|} + \frac{\|k\|}{\|h\|}\frac{R_g(y,k)}{\|k\|} \xrightarrow[h\to 0]{} 0.$$

□

Continuous and higher-order differentiability

Definition

A function $f : E \to F$ is called continuously differentiable or C^1 if it is differentiable at every point and has a continuous derivative $df : E \to L(E, F)$.

Definition

The k-th derivative of $f : E \to F$ is defined inductively as

$$d^k f := d(d^{k-1} f) : E \to L(E, L(E, \dots, L(E, F), \dots)) = L^k(E, \dots, E; F).$$

If this derivative exists and is continuous, then f is called C^k.

Remark. There is also a higher-order chain rule, which states that the composition of C^k functions is C^k.

Dual space and Riesz isomorphism

Definition

The dual space of a normed space E is the space E^* of continuous linear maps $A : E \to \mathbb{R}$ with the operator norm

$$\|A\| = \sup\big\{|A(x)| : x \in E, \|x\| \leq 1\big\}.$$

Theorem (Riesz)

Any Hilbert space E is isomorphic to its dual E^* via the Riesz isomorphism

$$E \ni x \mapsto \langle x, \cdot \rangle \in E^*.$$

Remark:

The inverse of the Riesz isomorphism is called Riesz representation.

Riesz isomorphism in coordinates

On Euclidean spaces E, the Riesz isomorphism identifies column vectors with row vectors:

$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in E \qquad \leftrightsquigarrow \qquad \big(x_1, \ldots, x_n\big) \in E^*$$

Gradients

Definition

If $f : E \to \mathbb{R}$ is differentiable at a point x in a Hilbert space E, then the gradient $\nabla f(x) \in E$ is the Riesz representation of the derivative $df(x) \in E^*$, i.e.,

$$\langle \nabla f(x), \cdot \rangle = df(x).$$

Remark. Thus, in coordinates, derivatives are row vectors, and gradients are column vectors.

Fundamental theorem of calculus

Theorem

- *If* $f : [a, b] \to E$ *is* C^1*, then* df *is integrable and*

$$f(x) = f(a) + \int_a^x df(t)dt, \qquad x \in [a, b].$$

- *If* $f : [a, b] \to E$ *is continuous and* E *is complete, then* f *is integrable, its indefinite integral is differentiable, and*

$$f(x) = \frac{d}{dx} \int_a^x f(t)dt, \qquad x \in [a, b].$$

Remark. The integral can be understood in the sense of Riemann or Lebesgue–Bochner; more on this later on.

Integral bounds on differentiable curves

Corollary

If $f : [a, b] \to E$ is C^1, then

$$\|f(b) - f(a)\| \leq \int_a^b \|\, df(t)\|\, dt \leq (b-a) \int_a^b \|\, df(t)\|^2\, dt.$$

Proof.

- The first inequality is Minkowski's inequality

$$\left\| \int_a^b g(t)\, dt \right\| \leq \int_a^b \|g(t)\|\, dt, \qquad \text{for integrable } g,$$

  which can be seen either as a theorem about Riemann integrals or a defining property of Lebesgue–Bochner integrals.
- The second inequality is the Cauchy-Schwarz inequality or Jentzen's inequality. □

Questions to answer for yourself / discuss with friends

- Repetition: What are the main definitions and results in differential calculus?
- Check your understanding: In High School we learned that the derivative of a function $f : \mathbb{R} \to \mathbb{R}$ is a real number, but here it is a linear functional!?
- Check your understanding: We have just learned that the derivative of a function $f : \mathbb{R} \to E$ is a linear functional, but in the fundamental theorem of calculus it is treated as an element of E!?
- Discussion: Why is there a completeness assumption in the fundamental theorem of calculus?

MH3520 Chapter 2

Part 2
Mathematical optimization

Motivation

People optimize:

- Computer scientists adjust parameters of neural networks to optimize the performance in learning tasks.
- Investors seek to create portfolios that avoid excessive risk while achieving a high rate of return.
- Manufacturers aim for maximum efficiency in the design and operation of their production processes.

Nature optimizes:

- Physical and chemical systems tend to states of minimum energy.
- Rays of light follow paths that minimize their travel time.

Mathematical formulation

Standing assumption:

- E is a normed vector space,
- X is a subset of E, which is called admissible region or search domain
- $f : X \to \mathbb{R}$ is a function, which is called objective function

Optimization problem:

$$\text{minimize} \qquad f(x) \qquad \text{over all} \qquad x \in X.$$

Remark:

- Minimization refers to finding the infimum $\inf_x f(x)$ and, if it exists, also the $\arg\min_x f(x)$.
- Minimization of f is equivalent to maximization of $-f$.

Classification of optimization problems

Variables:

- Continuous versus discrete
- Constrained versus unconstrained

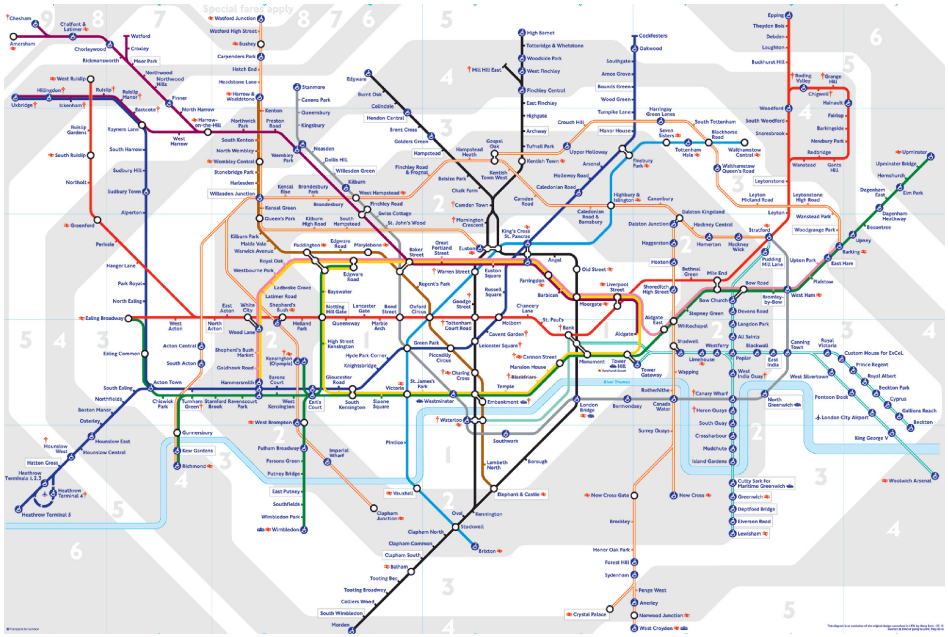
Objective:

- Convex versus non-convex
- Smooth versus non-smooth
- Deterministic versus stochastic

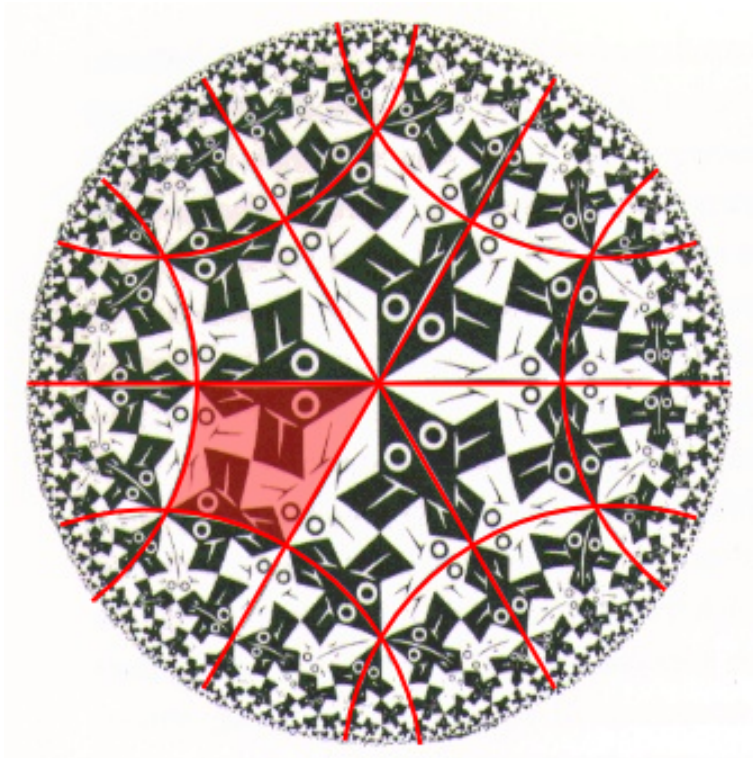
Optimization goal:

- Local versus global optimization

Continuous versus discrete variables

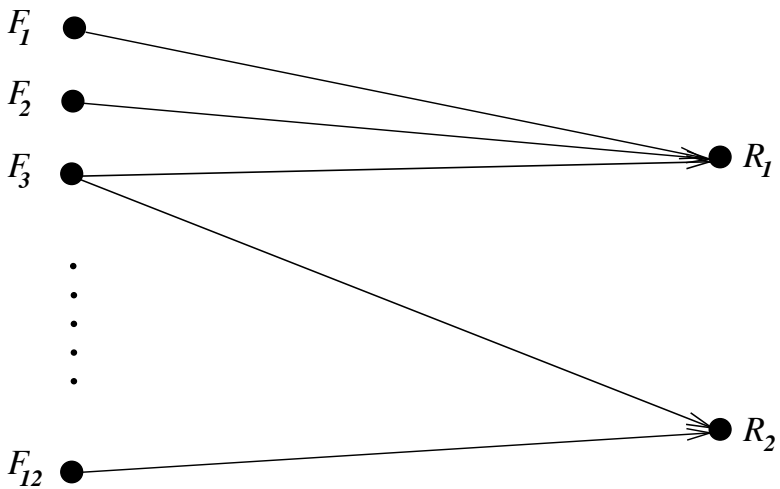


Shortest path in a graph
Discrete variables



Shortest path on a curved space
Continuous variables

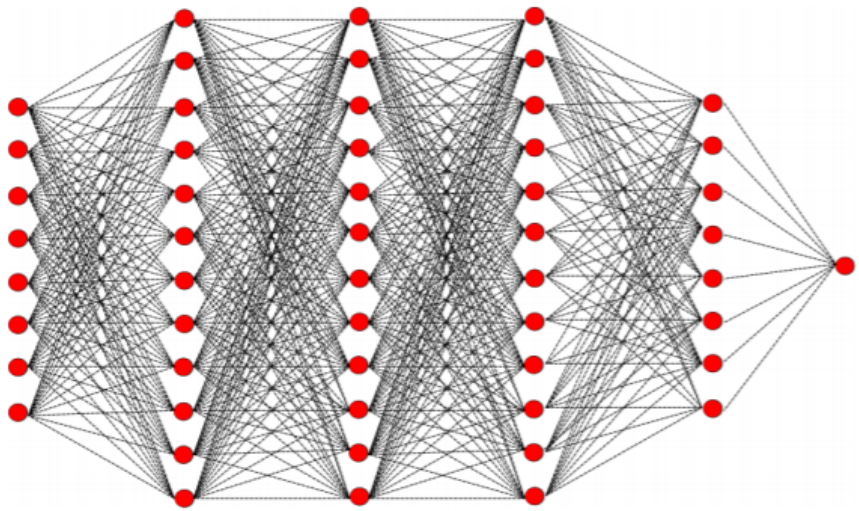
Constrained versus unconstrained variables



Optimal transport

Mass is non-negative and sums to 1

Constrained optimization

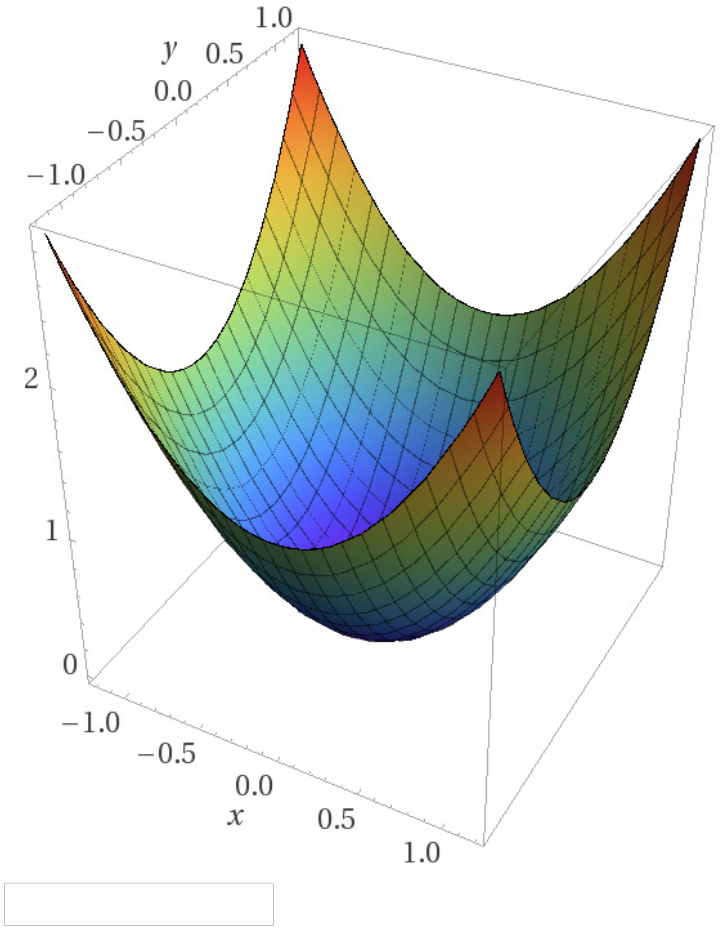


Deep learning

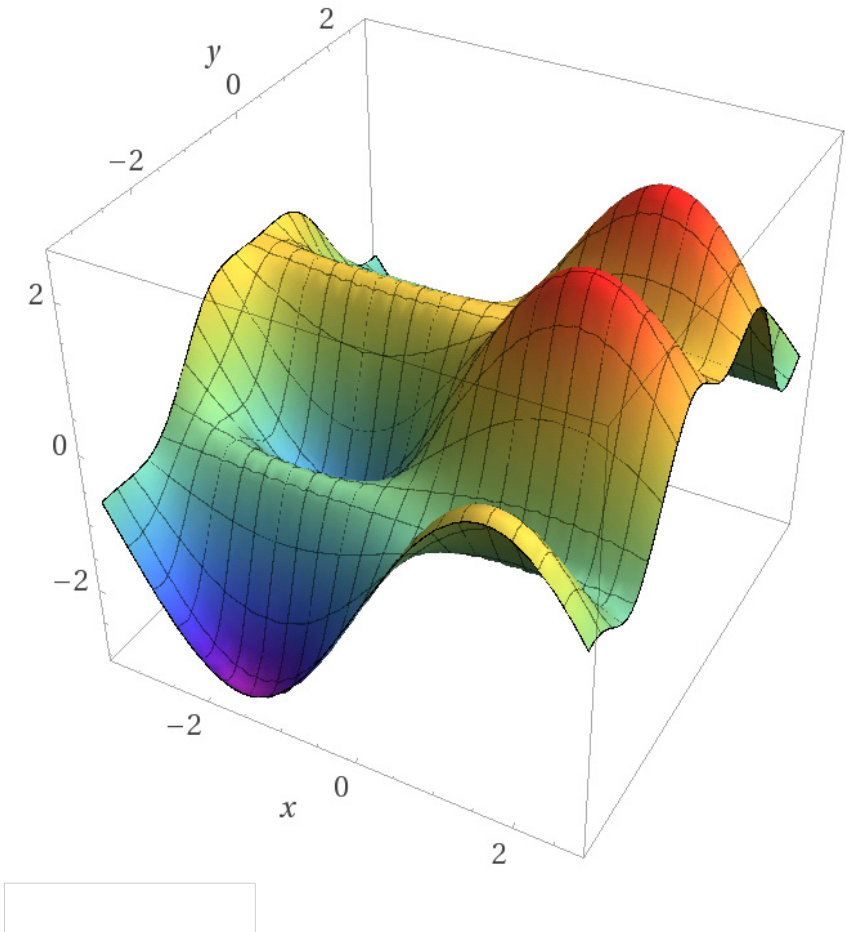
No restriction on network coefficients

Unconstrained optimization

Convex versus non-convex objective

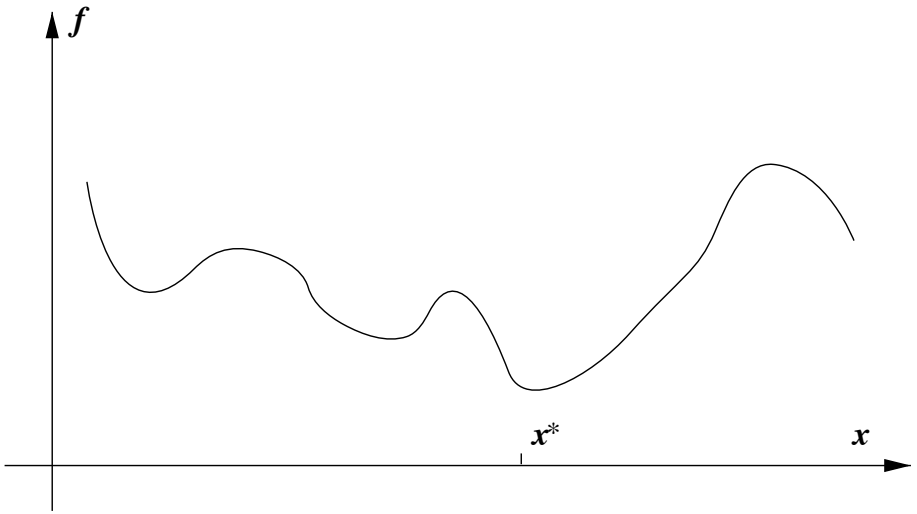


Convex objective

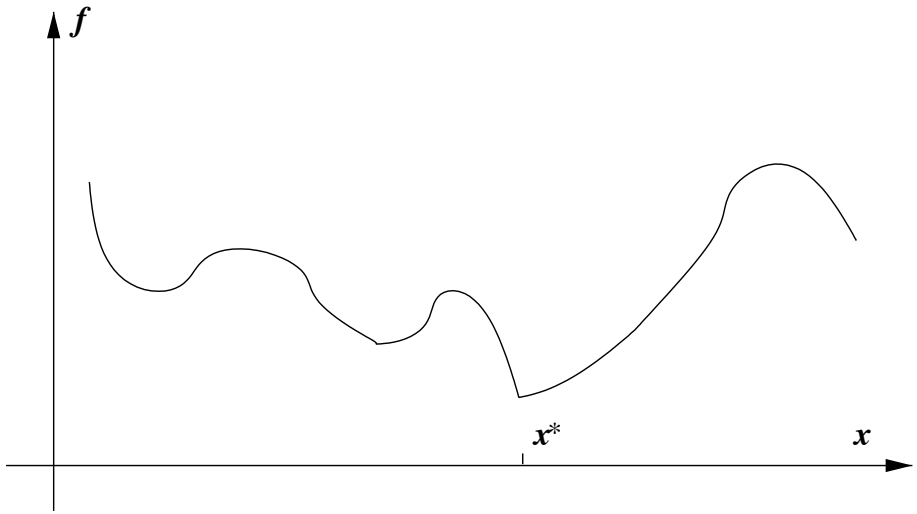


Non-convex objective

Smooth versus non-smooth objective

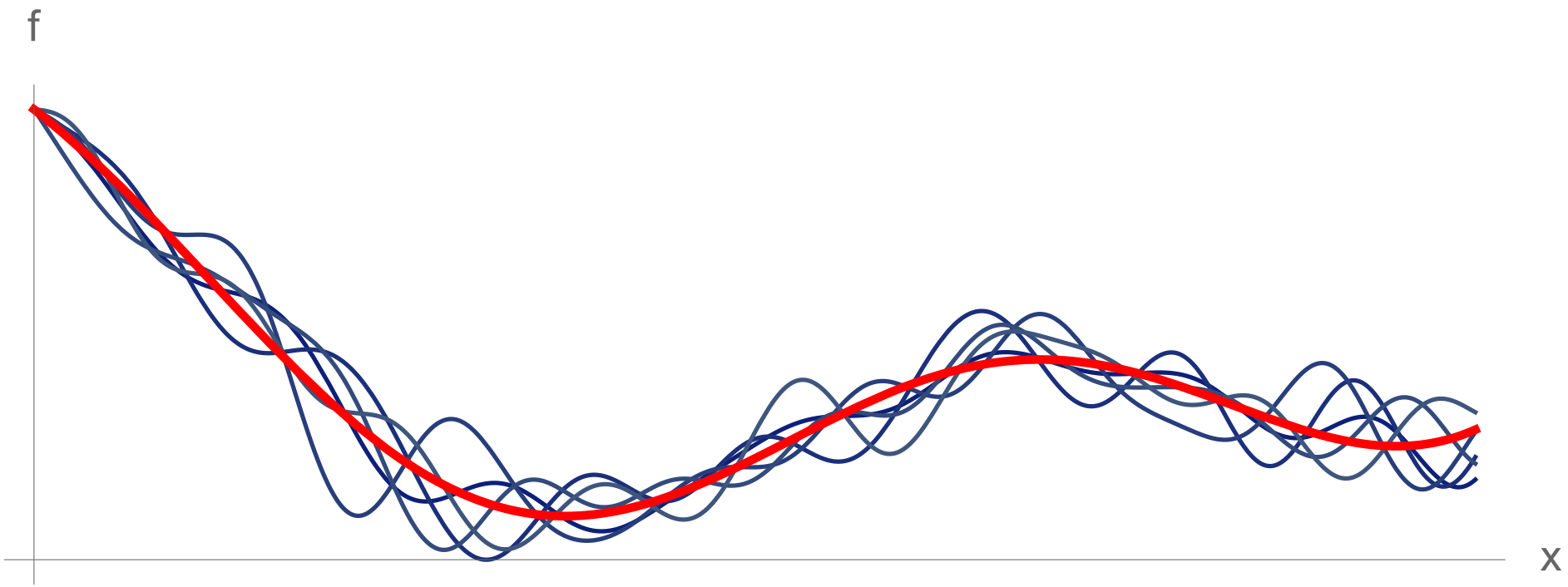


Smooth objective



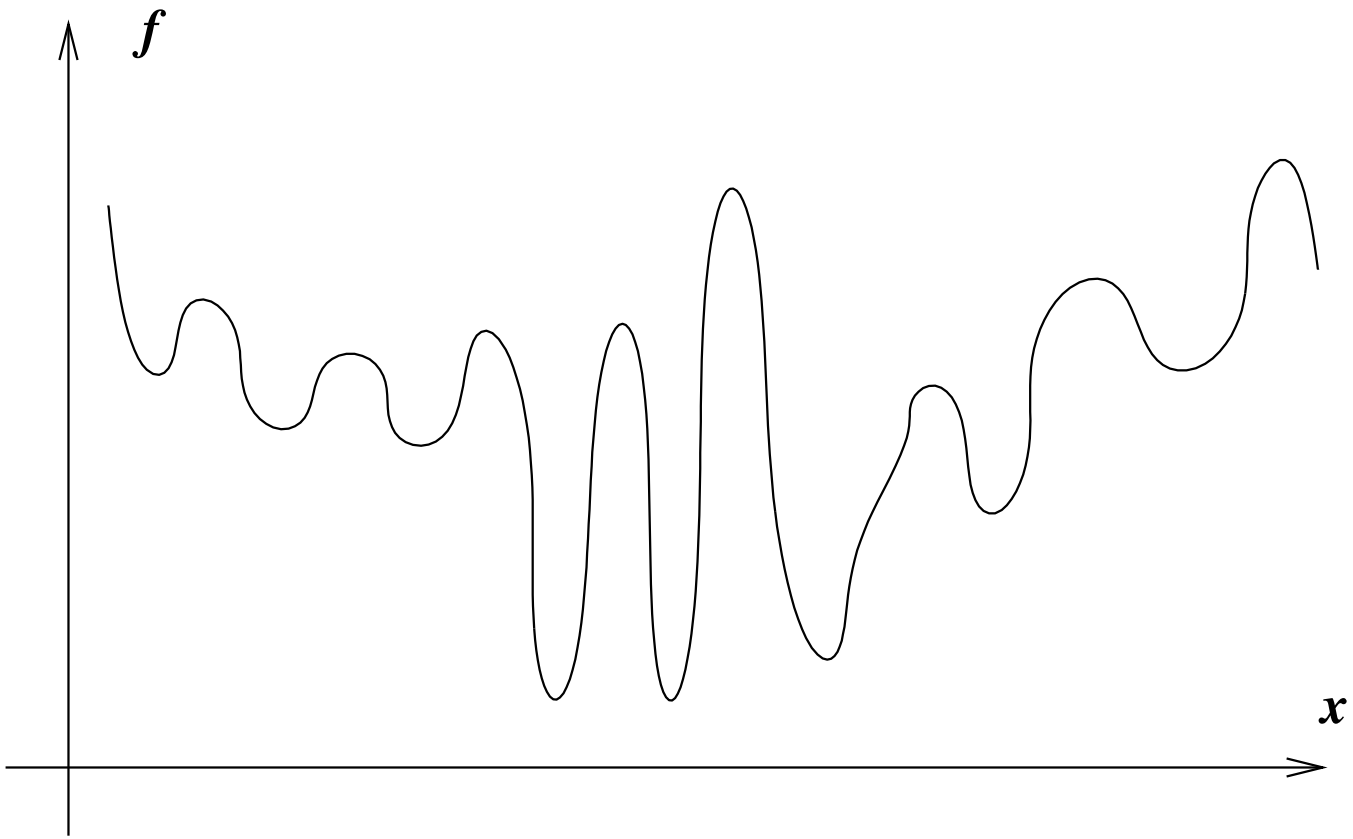
Non-smooth objective

Deterministic versus stochastic objective



Red: deterministic objective, dark: stochastic objectives

Local versus global objective



A difficult case for global minimization
Many local minima separated by steep walls

Types of minima

Definition

A point $x^* \in X$ is called a...

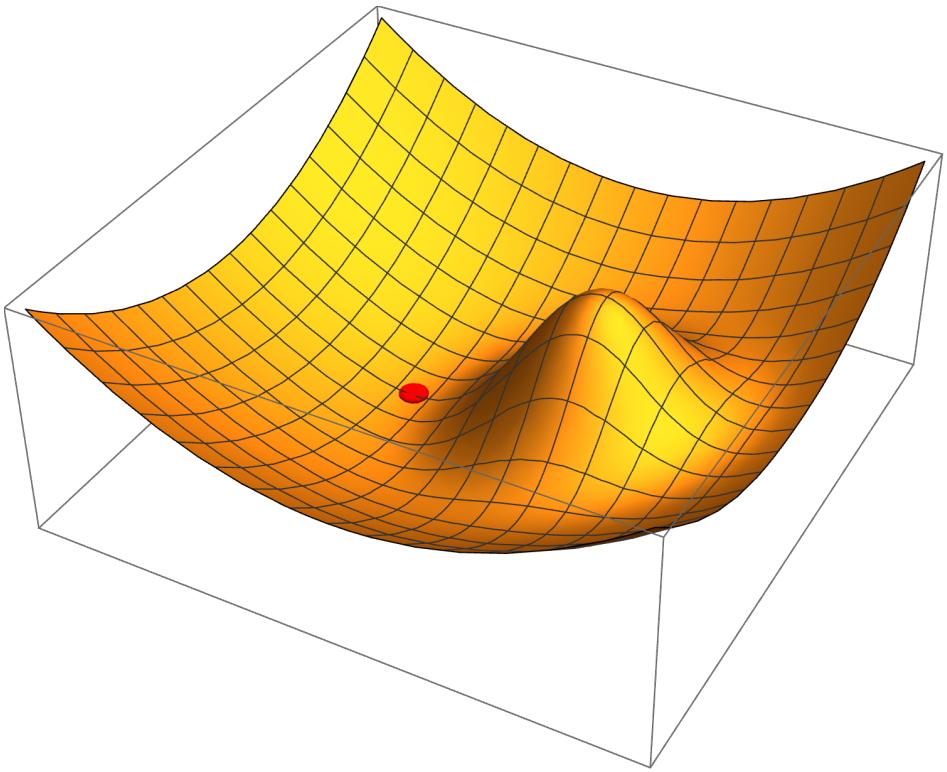
- Global minimum if $f(x^*) \leq f(x)$ for all $x \in X$
- Local minimum if $f(x^*) \leq f(x)$ for all x sufficiently close to x^*
- Strict local/global minimum if the above holds with $\leq$ replaced by $<$

Remark. Similarly for maxima.

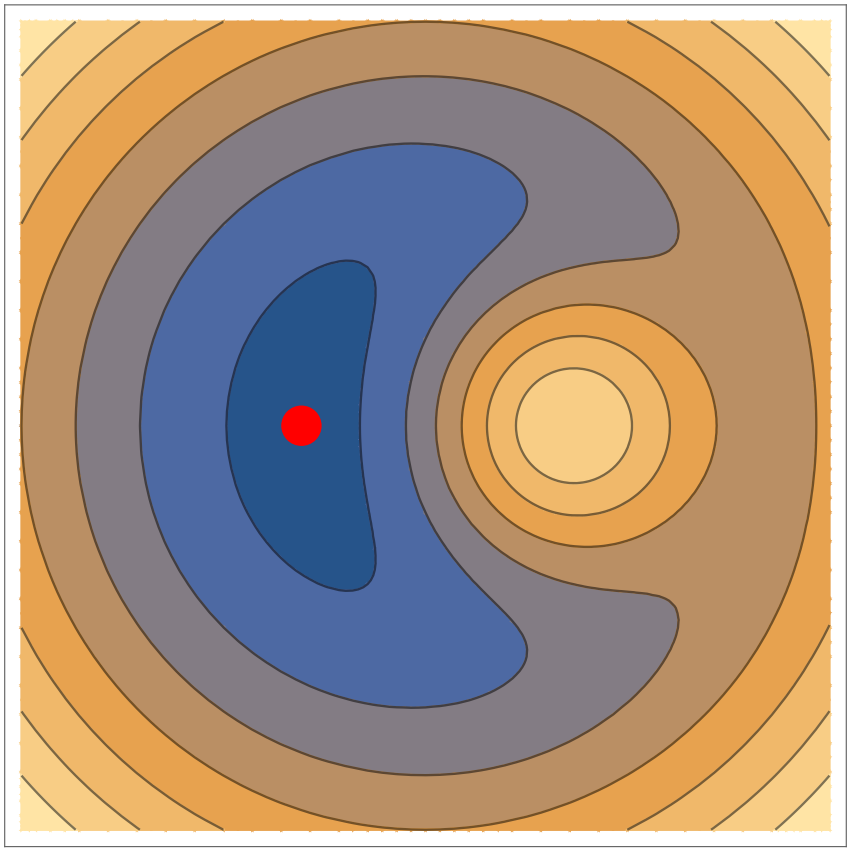
Remark. To visualize minima, one may plot:

- The graph $\{(x, f(x) : x \in X\}$
- The level sets $\{x \in X : f(x) = \ell\}$ at level $\ell \in \mathbb{R}$

Graphs and level sets



Graph of a function
Global minimum in red



Levelsets of the same function
Global minimum in red

Compactness

Do minimizers even exist? This is typically guaranteed by continuity and compactness assumptions...

Definition

A subset X of a normed vector space is called compact if every sequence in X has a converging subsequence.

Example

In finite dimensions, compactness is equivalent to boundedness and closedness. Thus, all intervals $[a, b]$ and closed balls $\{x \in \mathbb{R}^d : \|x\| \leq r\}$ are compact, but open or half-open intervals and open balls are not

Non-example

In infinite-dimensional spaces E, the closed balls $\{x \in E : \|x\| \leq r\}$ are not compact. Existence of minimizers is a delicate issue there.

Remark. Again, there is a more general definition of compactness for topological spaces.

Compactness and existence of minimizers

Theorem (Weierstrass)

If f is continuous and has a compact sublevel set

$$\{x \in X : f(x) \leq c\}, \qquad \textit{for some} \qquad c \in \mathbb{R},$$

then f has a global minimum.

Proof.

- Choose a minimizing sequence (x_n), i.e., a sequence such that

$$\lim_{n\to\infty} f(x_n) = \inf\{f(y) : y \in X\}.$$

- By compactness, one may replace (x_n) by a converging subsequence.
- The limit $x := \lim_{n\to\infty} x_n$ is a global minimum because

$$f(x) = f\Big(\lim_{n\to\infty} x_n\Big) = \lim_{n\to\infty} f(x_n) = \inf\{f(y) : y \in X\}. \qquad \square$$

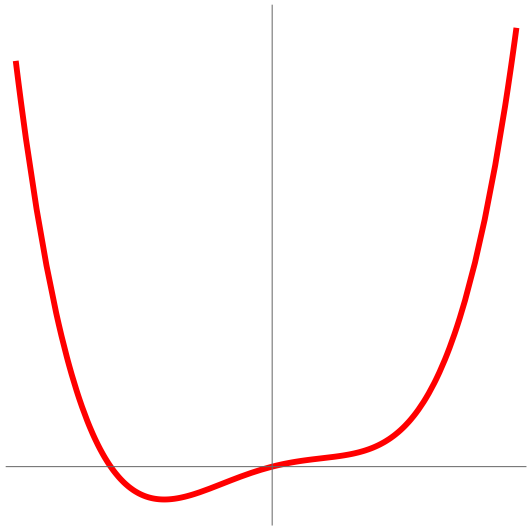
Coercivity and existence of minimizers

Corollary

If $f : \mathbb{R}^d \to \mathbb{R}$ is continuous and coercive, i.e.,

$$\lim_{\|x\| \to \infty} f(x) = \infty,$$

then f has compact sublevel sets, and the Weierstrass theorem applies.



Proof.

- Coercivity 'forces' the sublevel sets to be bounded (hence the name).
- As f is continuous, the sublevel sets are closed.
- In finite dimensions, this implies that they are compact. □

Remark. The argument breaks down in infinite dimensions, and existence of minimizers is a delicate issue there.

Characterization of local minima

Lemma (First derivative test)

If X is an open subset of a normed space and f has a local minimum at $x \in X$, then $df(x) = 0$.

Proof.

- By the chain rule, the function $t \mapsto f(x + th)$ is differentiable, for any $h \in E$.
- As x is a local minimum, the difference quotient $\frac{1}{t}(f(x + th) - f(x))$ is non-negative, for small t.
- In the limit $t \to 0$, one obtains that the derivative $df(x)h$ is non-negative.
- As this holds for h and $-h$, the derivative vanishes. □

Characterization of local minima

Lemma (Second derivative test)

If a twice differentiable function $f : X \to \mathbb{R}$ defined on an open subset of a Banach space E has a local minimum at $x \in X$, then $d^2 f(x) \geq 0$.

Proof.

- The Taylor expansion of $t \mapsto f(x + th)$ for $h \in E$ is

$$0 \leq f(x + th) - f(x) = \frac{t^2}{2} d^2 f(x)(h, h) + R(t), \quad \lim_{t \to 0} |t|^{-2} R(t) = 0.$$

- Divide by t^2 and send $t \to 0$ to obtain $d^2 f(x)(h, h) \geq 0$. □

Remark. Conversely, any point x with $df(x) = 0$ and $d^2 f(x) > 0$ is a local minimum.

From calculus to computation

Optimization problems in high-school Mathematics:

- Solve for $df(x) = 0$ by algebraic manipulations.
- Get closed-form solutions.

Obstacles in real-life problems (including deep learning):

- Algebraically solving for $df(x) = 0$ is typically infeasible.
- There are no closed-form solutions.

Numerical computation as a way out:

- There are only approximate solutions.
- These are typically obtained via iterative algorithms.
- No single algorithm works well for all problems.
- Calculus can guide algorithmic design

Questions to answer for yourself / discuss with friends

- Repetition: What properties ensure existence of minimizers?
- Check your understanding: In finite dimensions, compactness is equivalent to boundedness and closedness—but what does this mean again?
- Check your understanding: Write down the definition of coercivity using existential and universal quanitifiers.
- Transfer: In the classification of optimization problems, where do you see deep learning?

MH3520 Chapter 2

Part 3
Numerical optimization

Numerical optimization

Context:

- Mathematical optimization asks: do minimizers exist, are they unique, are they local or global, etc.
- Numerical optimization asks: how can I find these minimizers on my computer?

Main messages:

- Iteration is the single most important technique in continuous optimization.
- The simplest iteration is gradient descent, a discrete-time version of the gradient flow.
- It pays off to carefully formulate or reformulate the problem before feeding it into an optimization algorithm.

Numerical optimization pipeline

1. Creating a model:
   - Identifying objective, variables, and constraints
2. Optimizing the model:
   - Selecting an algorithm
   - Running the algorithm
3. Analyzing the model:
   - Robustness, efficiency, accuracy
   - Predictive power

Continuous optimization

From now on, let's focus on continuous optimization.

Standing assumption:

- $x \in E$ is the vector of variables in some Euclidean space E
- $f : E \to \mathbb{R}$ is the objective function
- $c_i : E \to \mathbb{R}$ are constraint functions

Optimization problem:

$$\min_{x \in E} f(x) \qquad \text{subject to} \qquad \begin{aligned} c_i(x) &= 0, \quad i \in \mathcal{E}, \\ c_i(x) &\geq 0, \quad i \in \mathcal{I}, \end{aligned}$$

where $\mathcal{E}$ and $\mathcal{I}$ are sets of indices for equality and inequality constraints. Luckily, most of the optimization problem in the machine learning does not have constraint

Global optimization in high dimensions is impossible

- In the present generality, optimization covers almost all needs of operations research and numerical analysis.
- As real life is too complicated to believe in universal tools, you might conjecture that optimization problems are unsolvable. Take the following theorem as an example.

Theorem

Consider an optimization algorithm $\mathcal{A}$ which depends only on n point evaluations of the objective, i.e., $\mathcal{A}(f)$ is a function of $f(x_1), \ldots, f(x_n)$ for some points $x_1, \ldots, x_n$ in the search domain. If the search domain is $[0,1]^d$ and the algorithm has accuracy ϵ, i.e.,

$$\left| \mathcal{A}(f) - \min_{x \in [0,1]^d} f(x) \right| \leq \epsilon$$

for every 1-Lipschitz objective function $f : [0,1]^d \to \mathbb{R}$, then $n \geq \lfloor \frac{1}{4\epsilon} \rfloor^d$.

Global optimization in high dimensions is impossible (cont.)

Example

If $\epsilon = 0.0025$ and $d = 10$, one needs $n \geq 10^{20}$ evaluations.

Proof.

- Assume for contradiction that $n < \lfloor \frac{1}{4\epsilon} \rfloor^d$ test points are sufficient for accuracy ϵ.
- Partition $[0,1]^d$ into disjoint small boxes of side length $\lfloor \frac{1}{4\epsilon} \rfloor^{-1}$.
- There are more boxes than test points. Thus, at least one of the boxes is empty, i.e., contains no test point.
- Construct a 'bad' objective function f, which is as negative as possible inside the empty box and zero outside.
- Then, the minimum of f is -2ϵ or lower.
- As $\mathcal{A}(f) = \mathcal{A}(0)$, the accuracy of $\mathcal{A}$ cannot be better than ϵ, a contradiction. □

Iterative algorithms

Beyond global optimization:

- A more modest goal is to find local minima, again up to some accuracy.
- This can be done iteratively: at each step an approximate minimizer x_n is updated by a hopefully better approximate minimizer x_{n+1}.
- Nearly all continuous optimization algorithms work like this!
- The big question is what kind of updating rule to use.

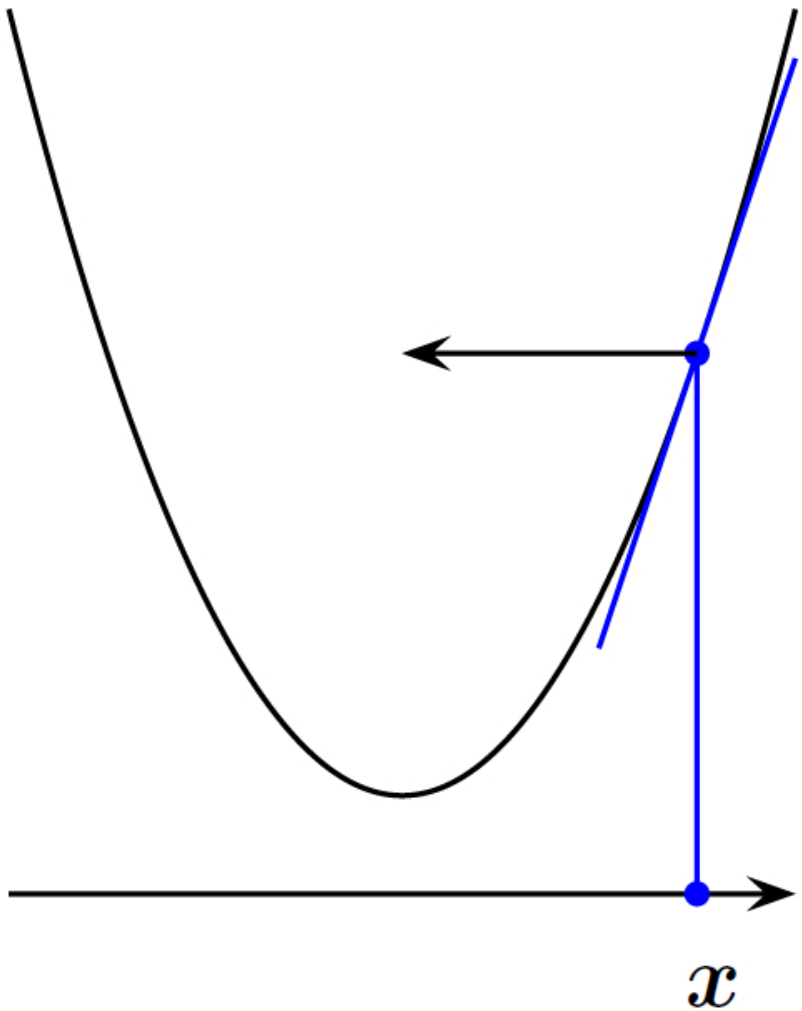
Intuition for gradient descent in dimension 1

- Given x_k, what is the best direction to search for a point x_{k+1} with $f(x_{k+1}) < f(x_k)$?

- What is a good step size?
- Idea: use increments proportional to the gradient,

$$x_{k+1} - x_k = -\alpha \nabla f(x_k),$$

for some learning rate $\alpha > 0$.



[Hutter and Boedecker]

Intuition for gradient descent in higher dimensions

Lemma

The negative gradient $-\nabla f(x)$ is the direction of strongest descent of f at x, i.e., it minimizes $df(x)h$ over all h in the unit sphere.

Proof.

- By Cauchy–Schwarz,

$$df(x)h = \langle \nabla f(x), h \rangle \leq \|\nabla f(x)\| \|h\|,$$

  with equality iff h is a multiple of $\nabla f(x)$.
- Thus, the normalized gradient $h^* = \nabla f(x) \|\nabla f(x)\|^{-1}$ maximizes the function $df(x)h$ over the sphere $\|h\| = 1$.
- Equivalently, $-h^*$ minimizes $df(x)h$ over the same sphere. □

Gradient descent algorithm

A greedy algorithm for continuous unconstrained optimization:

---

input : function $f : E \to \mathbb{R}$, learning rate $\alpha > 0$, initial guess $x \in E$
output: point x such that $f(x)$ is small
repeat
  $x \leftarrow x - \alpha \nabla f(x)$
until convergence
return x

Questions:

- Convergence? Convergence rate?
- Choice of learning rate? Time-dependent learning rate?
- Refinements

Convergence Results of Gradient Descent

Below we list the theorem and proof can be found in the optional parts later this chapter.

Theorem

If f is convex with minimum at x^*, and ∇f has Lipschitz constant $1/\alpha$, then f decreases linearly along the gradient descent with learning rate α:

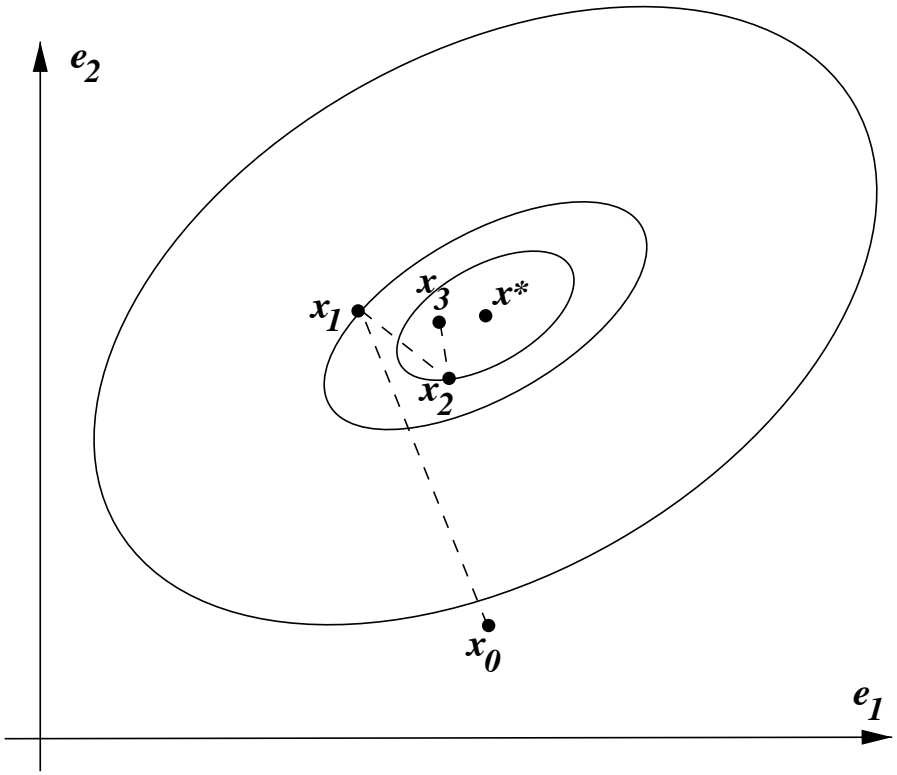
$$f(x_k) - f(x^*) \leq \frac{2\|x_0 - x^*\|^2}{\alpha k} = O((\alpha k)^{-1}).$$

Variations of gradient descent

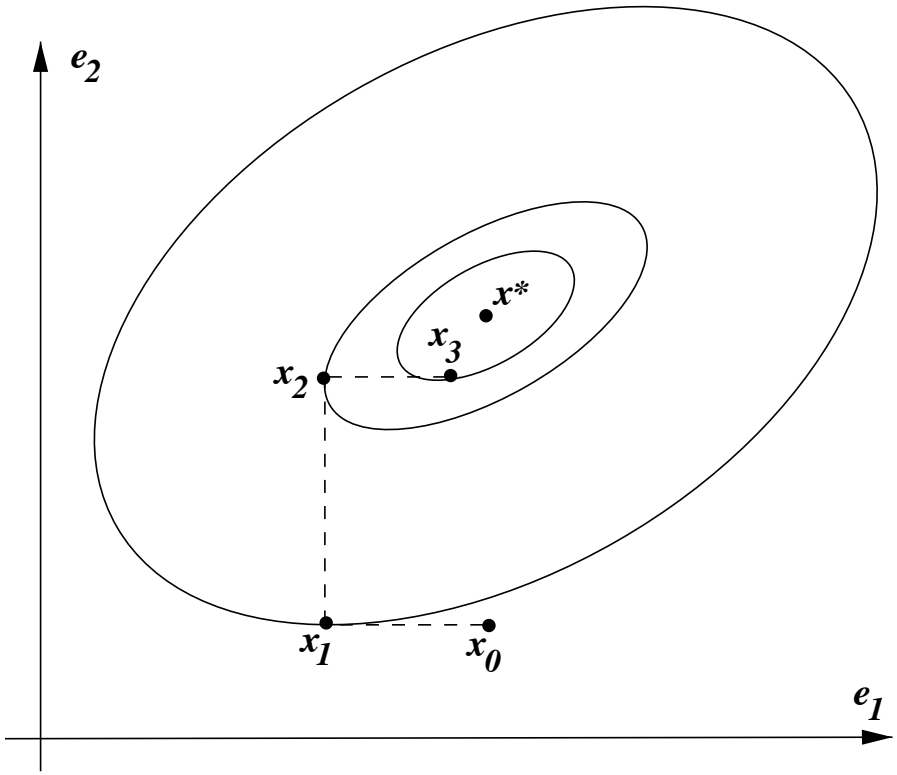
There are many variations and refinements...

- Gradient descent: the increment is the negative gradient of f at x_n, multiplied by some 'learning rate' $\alpha > 0$
- Coordinate descent: as before, but change only one variable at a time
- Trust region: x_{n+1} is the minimizer of a linear or quadratic approximation of f, restricted to a small 'trusted' region around x_n
- Constraints can be handled by penalization (see later) or by projection to the admissible set

Gradient descent versus coordinate descent



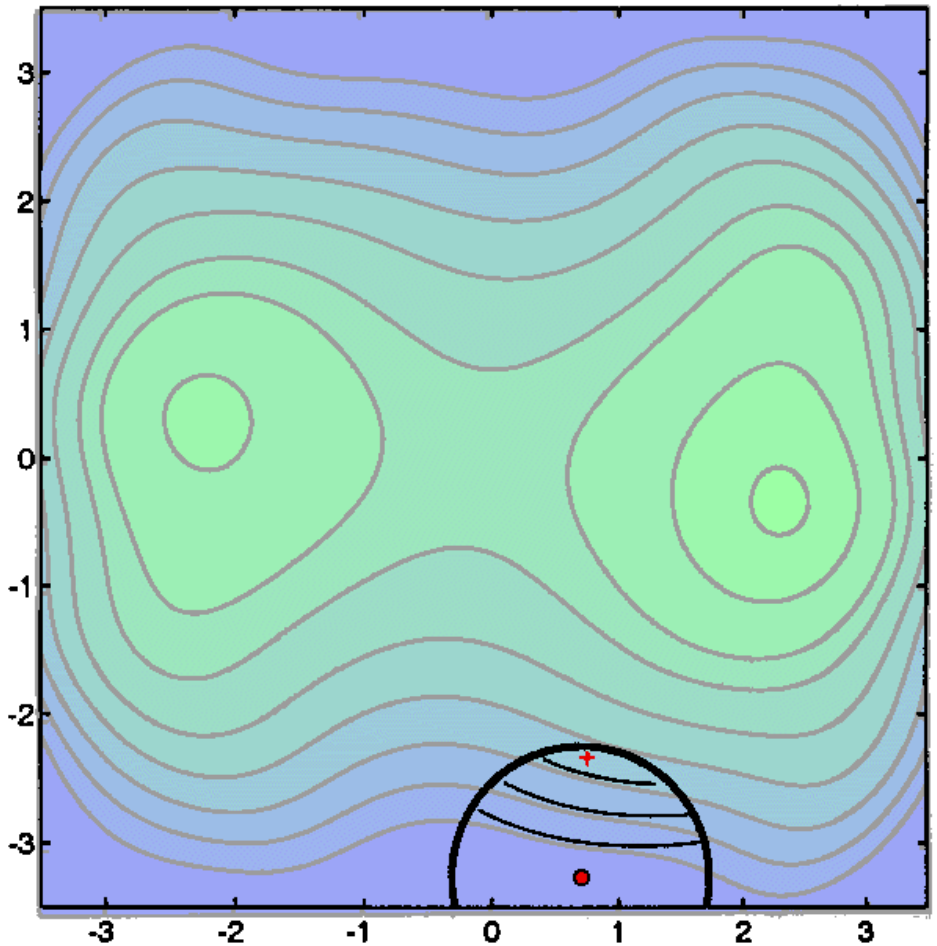
Gradient descent



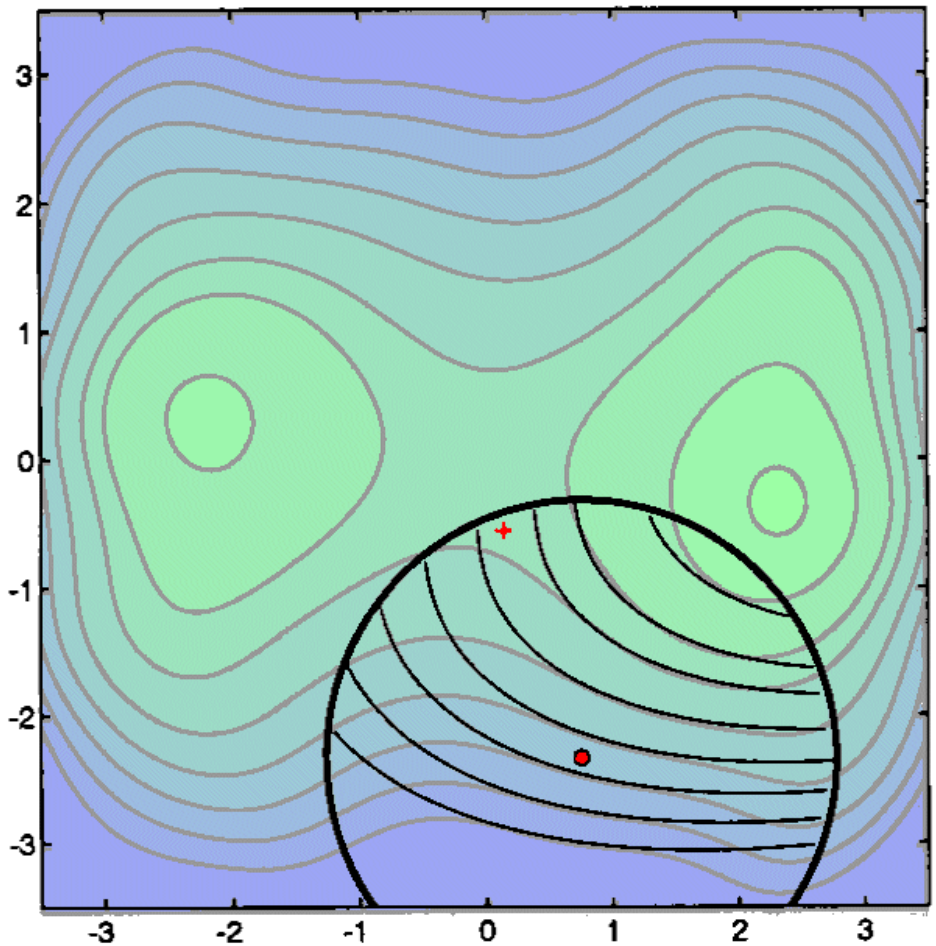
Coordinate descent

Trust region

Contours lines of the objective function (gray) and of a quadratic approximation inside of circular trust region (black):



First step
Small confidence



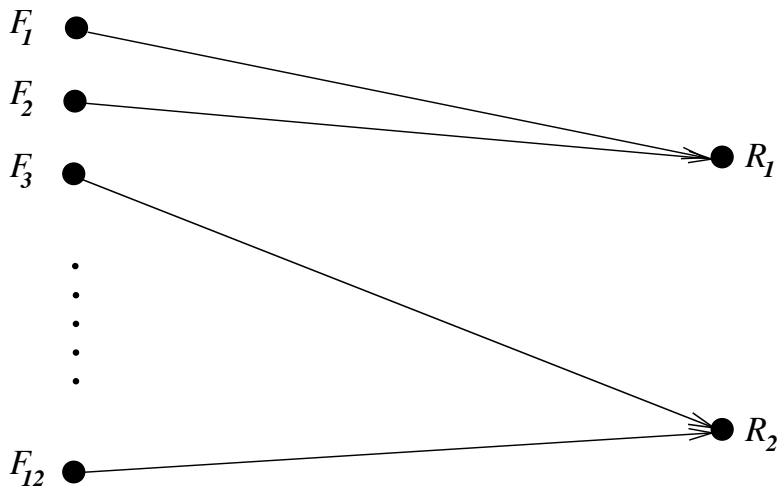
Second step
More confidence

Transformations to non-equivalent models

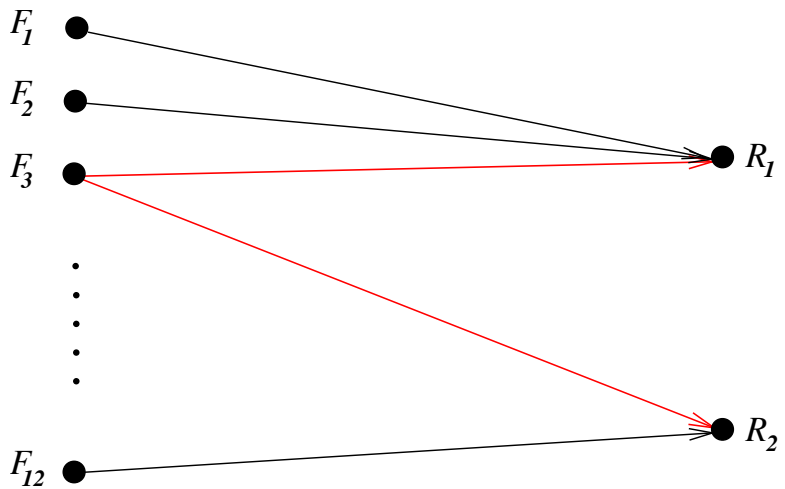
It can pay off to change the model before feeding it into an algorithm:

- Relaxation: solve an easier problem with less constraints or more free variables
- Regularization: force x to be well-behaved by adding a penalty function $p(x)$ to the objective $f(x)$
- Discretization: consider x as an n-dimensional approximation of some infinite-dimensional $\hat{x}$ and send $n \to \infty$

Relaxation



Monge optimal transport
mass cannot split
non-convex problem



Kantorovich optimal transport
mass can split
convex problem

Regularization

Linear equations $Ax = b$ can be solved by optimization methods:

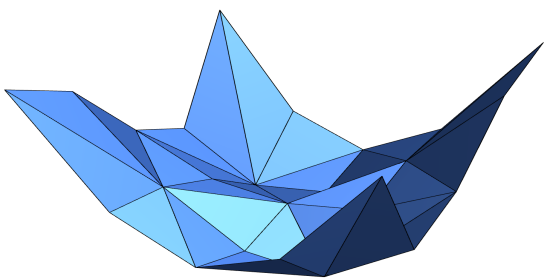
$$\min_x \|Ax - b\|^2 \qquad \min_x \|Ax - b\|^2 + \epsilon\|x\|^2$$

potentially ill posed
no regularization
numerically difficult

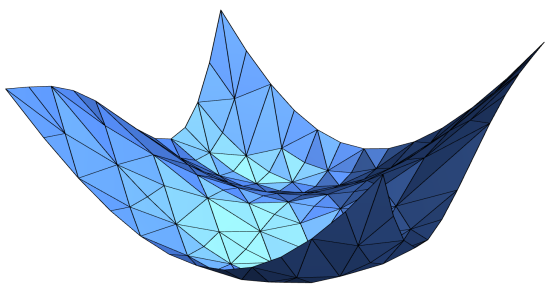
well posed
regularized
better conditioned

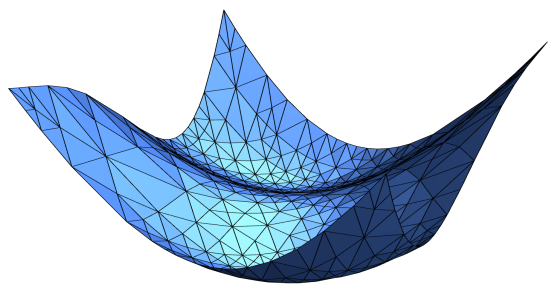
Discretization

Infinite-dimensional function spaces must be discretized to make them amenable to numerical optimization:



coarse discretization
few variables





fine discretization
many variables

Transformations to equivalent models

It can pay off to reformulate the model before feeding it into an algorithm:

- Transformation of objective/equality constraints/inequality constraints by monotonic/root-preserving/positivity-preserving transformations
- Moving constraints to objective: replace constraints by penalty or barrier functions, which are added to the objective
- Variable elimination: remove an equality constraint $c_i(x) = 0$ by eliminating some degrees of freedom in x
- Duality is about writing the objective as a supremum $f(x) = \max_y a_y + b_y x$ over affine functions and interchanging $\min_x \max_y$ and $\max_y \min_y$.

Transformation of objective and constraints

Geometric programming is a systematic way of transforming certain polynomial problems into equivalent convex problems by logarithmic transformations of the variables, constraints, and objectives:

$$\text{minimize } x_1^{-1} + 2x_2^{1/3}$$
$$\text{subject to } x_1 \leq 5x_2^{-1/7}$$

positive variables x_i
non-convex problem

$$\text{minimize } \log\left(e^{-y_1} + 2e^{x_2/3}\right)$$
$$\text{subject to } y_1 \leq \log(5) - y_2/7$$

variables $y_i := \log x_i$
equivalent convex problem

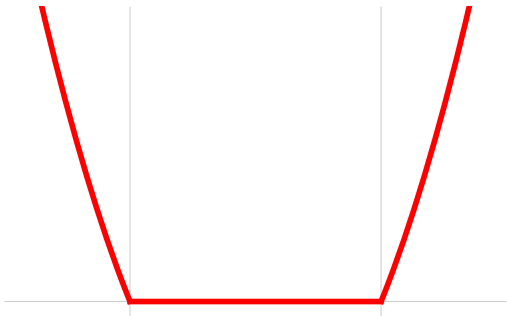
Moving constraints to the objective

Constrained problem:

$$\text{minimize } f(x) \text{ subject to } c_i(x) \leq 0$$

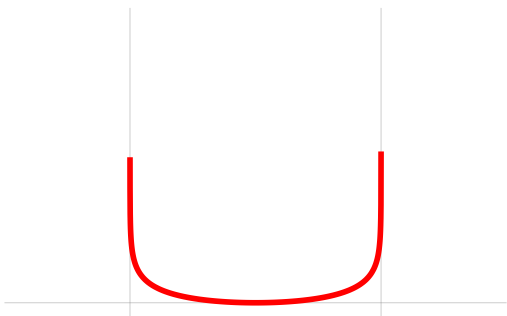
Constraints replaced by penalties:

$$\text{minimize } f(x) + \sum_i \max\{0, c_i(x)\}$$



Constraints replaced by barriers:

$$\text{minimize } f(x) - \sum_i \log(-c_i(x))$$



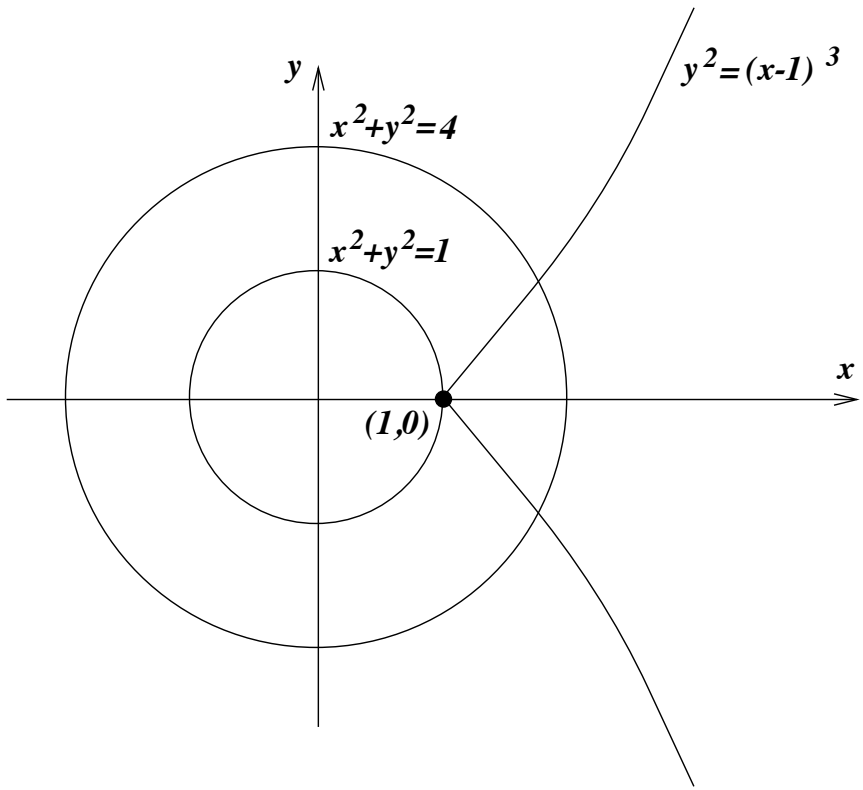
(Alternative penalty and barrier functions are possible.)

Variable elimination

Elimination techniques must be used with care as they may alter the problem or introduce ill conditioning. Here is an example.

Constrained problem:

$$\begin{aligned} &\text{minimize } x^2 + y^2 \\ &\text{subject to } (x-1)^3 = y^2 \end{aligned}$$



Elimination of y:

$$\text{minimize } x^2 + (x-1)^3$$

Forgotten implicit constraint:

$$\text{subject to } x \geq 1$$

Duality

Lagrangian for the constrained problem:

$$L(x, \lambda, \mu) = f(x) + \sum_{i \in \mathcal{E}} \lambda_i c_i(x) + \sum_{i \in \mathcal{I}} \mu_i c_i(x).$$

Primal problem: always equivalent to the original problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \ \underset{\lambda_i \in \mathbb{R}_+, \mu_i \in \mathbb{R}}{\text{maximize}} \ L(x, \lambda, \mu)$$

Dual problem: always convex; sometimes equivalent to the original problem

$$\underset{\lambda_i \in \mathbb{R}_+, \mu_i \in \mathbb{R}}{\text{maximize}} \ \underset{x \in \mathbb{R}^n}{\text{minimize}} \ L(x, \lambda, \mu)$$

Questions to answer for yourself / discuss with friends

- Repetition: In what sense is global optimization in high dimensions impossible?
- Repetition: What is the intuition behind gradient descent, and how does it work?
- Repetition: How can optimization problems be transformed into equivalent or related problems? Give some examples.

MH3520 Chapter 2

Part 4

Gradient flow and gradient descent (Optional)

Gradient flow and gradient descent

Context:

- Gradient descent is a greedy optimization algorithm.
- So far, we have no convergence guarantees, and we don't even know if it actually decreases the objective.

Continuous versus discrete time:

- Gradient flow is a continuous-time version of gradient descent.
- Analysis is easier in continuous time.
- Numerical implementation requires discrete time.

Main messages:

- Gradient flow always decreases the objective.
- Gradient descent decreases the objective if the step size is small.

Definitions

Standing assumption. $f : E \to \mathbb{R}$ is a C^1 function on a Hilbert space E.

Definition

The gradient flow of f (if it exists) is a C^1 function $x : [0, \infty) \to E$ which satisfies the differential equation

$$\partial_t x(t) = -\nabla f(x(t)), \qquad x(0) = x_0.$$

Definition

The gradient descent of f is a sequence (x_k) in E such that

$$x_{k+1} - x_k = -\alpha \nabla f(x_k),$$

where the parameter $\alpha > 0$ is called learning rate.

Lipschitz property

Definition

A function $F : E \to E$ is Lipschitz with Lipschitz constant $L > 0$ if

$$\|F(x) - F(y)\| \leq L\|x - y\| \qquad \text{for all } x, y \in E.$$

F is called locally Lipschitz if it is Lipschitz near every point in E.

Example

Every every function with bounded derivative is Lipschitz, and every C^1 function is locally Lipschitz.

Application

∇f is locally Lipschitz if f is C^2.

Theorem (Picard–Lindelöf)

For any locally Lipschitz function $F : E \to E$ on a Banach space E and any initial value $x_0 \in E$, the differential equation

$$x'(t) = F(x(t)), \qquad x(0) = x_0$$

has a unique solution $x \in C^1((-T, T), E)$ for sufficiently small T.

Sketch of proof. Rewrite the differential equation as an integral equation

$$x(t) = x_0 + \int_0^t F(x(s))\, ds,$$

show that the right-hand side is a contraction on $C([-T, T], E)$ for small T, and invoke the Banach fixed point theorem. □

Existence of the gradient flow

Theorem

The gradient flow of f is well defined if f is C^2 and bounded from below.

Remark. The main point here is existence for all time.

Proof.

- As ∇f is locally Lipschitz, Picard–Lindelöf ensures that the gradient flow is well defined on $[0, t)$ for some t.
- Without loss of generality, t is maximal.
- Assume for contradiction that t is finite, let $t_n \nearrow t$, and define $x_n := x(t_n)$.
- Note that $f(x_n)$ is bounded because f is bounded from below and decreases along the gradient flow, as we will see in a moment.

Existence of the gradient flow (cont.)

- The sequence (x_n) is Cauchy because

$$\begin{aligned}
\|x_n - x_m\|^2 &= \Big\| \int_{t_m}^{t_n} x'(t)\,dt \Big\|^2 = \Big\| \int_{t_m}^{t_n} \nabla f(x(t))\,dt \Big\|^2 \\
&\le \Big(\int_{t_m}^{t_n} \|\nabla f(x(t))\|\,dt \Big)^2 \le (t_n - t_m) \int_{t_m}^{t_n} \|\nabla f(x(t))\|^2\,dt \\
&= (t_n - t_m) \int_{t_m}^{t_n} \langle \nabla f(x(t)), x'(t) \rangle\,dt \\
&= (t_n - t_m) \int_{t_m}^{t_n} \partial_t f(x(t))\,dt \\
&= (t_n - t_m) \underbrace{(f(x_n) - f(x_m))}_{\text{bounded}} \xrightarrow[n,m\to\infty]{} 0.
\end{aligned}$$

- By Picard–Lindelöf, the gradient flow can be restarted at $\lim_{n\to\infty} x_n$, in contradiction to the maximality of t. Thus, t cannot be finite. □

Decay along the gradient flow

Lemma

Any differentiable function decreases along its gradient flow.

Proof.

$$\begin{aligned} d(f \circ x)(t) &= df(x(t)) \circ dx(t) \\ &= \langle \nabla f(x(t)), dx(t) \rangle = -\|\nabla f(x(t))\|^2 < 0. \end{aligned}$$

□

Remark. Without any further assumptions, the decay may be very slow.

Decay along the gradient descent

If the learning rate is too large, the objective may increase along the gradient descent. Let's try to rule this out:

Lemma

If ∇f has Lipschitz constant $L > 0$ and the learning rate is at most $2/L$, then f decreases along its gradient descent.

Proof: auxiliary bound on the Taylor remainder of f, proof thereafter.

Remark.

- The preferred choice is $\alpha = 1/L$.
- In applications, one rarely knows L. That's a problem.

Decay along the gradient descent (cont.)

Lemma (Bound on the Taylor remainder)

If ∇f has Lipschitz constant L, then

$$\left|f(y) - f(x) - \langle \nabla f(x), y - x\rangle\right| \leq \frac{L}{2}\|y - x\|^2.$$

Proof.

$$\begin{aligned}
&\left|f(y) - f(x) - \langle \nabla f(x), y - x\rangle\right| \\
&\quad = \Big| \int_0^1 \langle \nabla f(x + t(y - x)) - \nabla f(x), y - x\rangle \, dt \Big| \\
&\quad \leq \int_0^1 \|\nabla f(x + t(y - x)) - \nabla f(x)\| \|y - x\| \, dt \\
&\quad \leq \int_0^1 Lt\|y - x\|^2 \, dt = \frac{L}{2}\|y - x\|^2.
\end{aligned}$$

□

Decay along the gradient descent (cont.)

Proof.

- The bound on the Taylor remainder implies the one-sided estimate

$$f(y) \le f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2.$$

- Setting $y := x - \alpha \nabla f(x)$ yields

$$\begin{aligned} f(y) &\le f(x) - \alpha\|\nabla f(x)\|^2 + \alpha^2 \frac{L}{2}\|\nabla f(x)\|^2 \\ &= f(x) - \alpha(1 - \alpha L/2)\|\nabla f(x)\|^2 \le f(x) \quad \text{if } \alpha \le 2/L. \end{aligned}$$

□

Questions to answer for yourself / discuss with friends

- Repetition: What properties ensure that the objective decays along the gradient flow or gradient descent?
- Check your understanding: Why are we working in a Hilbert space?
- Check your understanding: What does it mean to be the solution of a differential equation?
- Discussion: Can these results be used for solving optimization problems, and if so, how?

MH3520 Chapter 2

Part 5

Minimizers of convex objectives (optional)

Convexity

Motivation:

- Whenever an optimization problem can be solved efficiently, there tends to be some convexity involved.

Key points:

- For convex objectives, local minima are global minima
- For strictly convex objectives, global minima are unique.

Convexity

Definition

- A convex set is a subset X of a vector space such that

$$[x, y]_\alpha := \alpha x + (1 - \alpha)y \in X, \qquad x, y \in X, \alpha \in (0, 1).$$

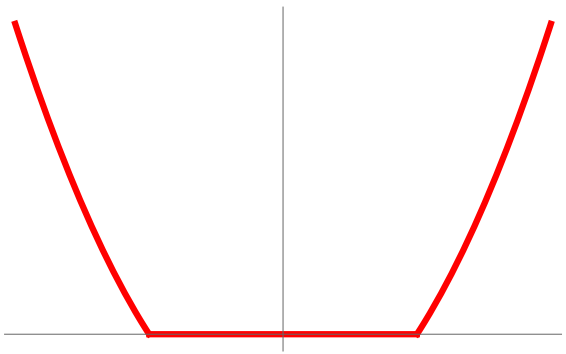
- A convex function is a function f with convex domain X such that

$$f([x, y]_\alpha) \leq [f(x), f(y)]_\alpha, \qquad x, y \in X, \alpha \in (0, 1).$$

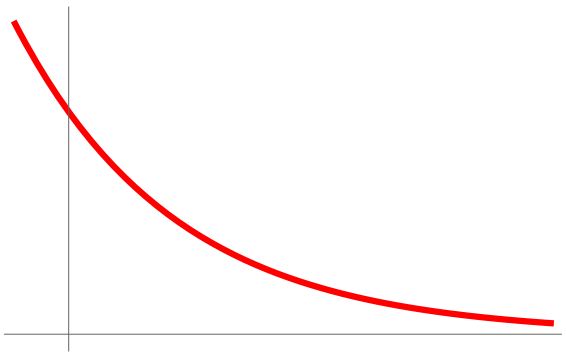
- f is strictly convex if the inequality is strict for $x \neq y$.
- f is uniformly convex if there exists $\mu > 0$ such that

$$f([x, y]_\alpha) \leq [f(x), f(y)]_\alpha - \tfrac{1}{2}\mu\alpha(1 - \alpha)\|x - y\|^2, \quad x, y \in X, \alpha \in (0, 1).$$

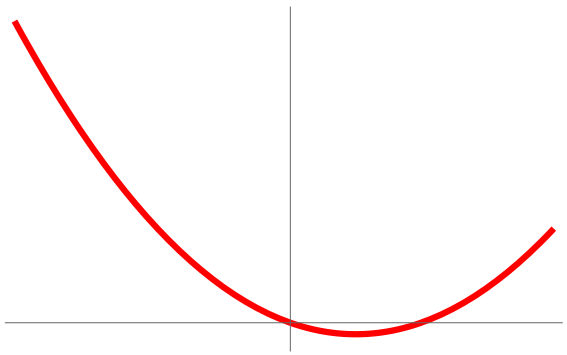
Convexity



convex



strictly convex



uniformly convex

First-order characterization of convexity

Proposition

If f is continuously differentiable on an open convex subset X of a normed vector space E, then

- *f is convex iff*

$$df(x)(y-x) \leq f(y) - f(x), \qquad x, y \in X.$$

- *f is strictly convex iff the inequality is strict for $x \neq y$.*
- *f is uniformly convex iff there exists $\mu > 0$ such that*

$$df(x)(y-x) + \tfrac{\mu}{2}\|y-x\|^2 \leq f(y) - f(x).$$

Second-order characterization of convexity

Proposition

If f is twice continuously differentiable on an open convex subset X of a normed vector space E, then

- *f is convex iff*

$$d^2 f(x)(h,h) \geq 0, \qquad x \in X, h \in E.$$

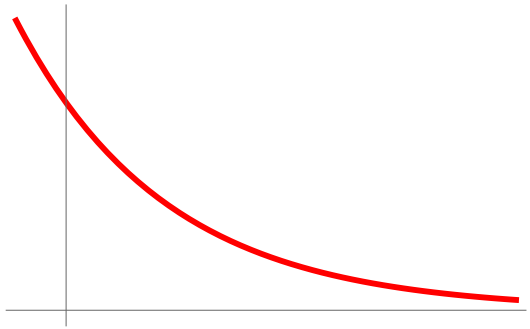
- *f is strictly convex iff the inequality is strict for $x \neq y$.*
- *f is uniformly convex iff there exists $\mu > 0$ such that*

$$d^2 f(x)(h,h) \geq \mu \|h\|^2.$$

Minima of convex functions

Non-example

The strictly convex function $x \mapsto e^{-x}$ has no minimum.



Lemma

Any critical point of a C^1 convex function is a global minimum.

Proof.

- Let $df(x) = 0$, and let y be any other point.
- As f is convex, $f(y)$ lies above the tangent line at x, i.e.,

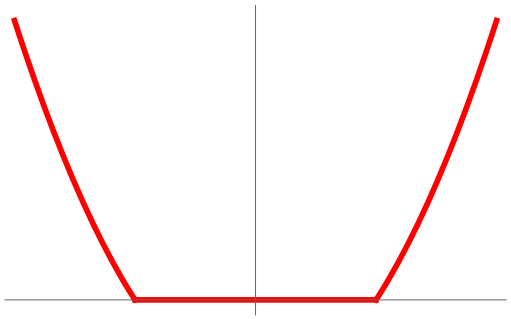
$$f(y) - f(x) \geq df(x)(y - x) = 0.$$

- Thus, $f(x)$ is a global minimum. □

Convexity and uniqueness of minima

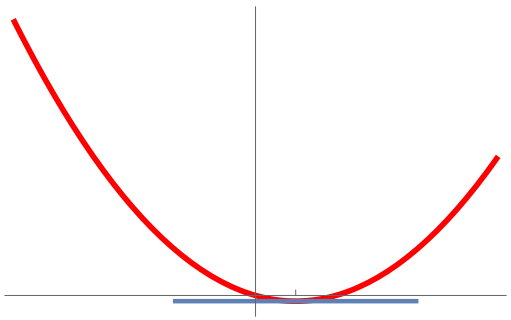
Counter-example

The convex function $x \mapsto \max\{0, x^2 - 1\}$ has multiple global minima.



Lemma

Any strictly convex function has at most one local or global minimum.



Proof.

- Assume for contradiction that there are are two distinct local (and thus global) minima x, y.
- Let z be a strict convex combination of x and y.
- By strict convexity, $f(z)$ is strictly smaller than $f(x)$ and $f(y)$, in contradiction to the minimality of x, y. □

Questions to answer for yourself / discuss with friends

- Repetition: What are the definitions and some characterizations of convexity, strict convexity, and uniform convexity?
- Check your understanding: Why did we require the domain of f to be open?
- Check your understanding: Are there any non-differentiable convex functions?
- Transfer: Is deep learning a convex optimization problem? If not, is it at least convex in some of its variables?

MH3520 Chapter 2

Part 6

Gradient flow and gradient descent of convex objectives (Optional)

Gradient flow and gradient descent of convex objectives

Context:

- The objective may decay arbitrarily slow along the gradient flow or gradient descent.

Main results:

- For convex objectives one gets linear decay.
- For uniformly convex objectives one gets exponential decay.

Warning:

- The terms linear and exponential decay are used with different meanings in the literature.

Linear decay along the gradient flow

A convex objective decreases linearly along the gradient flow:

Theorem

If f is convex and has a minimum at x^*, then f decreases linearly along the gradient flow:

$$f(x(t)) - f(x^*) \leq \frac{\|x_0 - x^*\|^2}{t} = O(t^{-1}).$$

Proof: skipped.

Remark. The idea is to show that for some $C > 0$ that

$$\frac{d}{dt} \frac{1}{f(x(t)) - f(x^*)} \geq C.$$

Linear decay along the gradient descent

A convex objective decreases linearly along the gradient descent, provided that the learning rate is sufficiently small:

Theorem

If f is convex with minimum at x^*, and ∇f has Lipschitz constant $1/\alpha$, then f decreases linearly along the gradient descent with learning rate α:

$$f(x_k) - f(x^*) \leq \frac{2\|x_0 - x^*\|^2}{\alpha k} = O((\alpha k)^{-1}).$$

Proof: skipped.

Remark. The idea is to show for some $C > 0$ that

$$\frac{1}{f(x_{k+1}) - f(x^*)} - \frac{1}{f(x_k) - f(x^*)} \geq C.$$

The Łojasiewicz inequality

For uniformly convex functions, the optimality gap is bounded in terms of the squared norm of the gradient:

Lemma

If f is uniformly convex with convexity constant μ, i.e.,

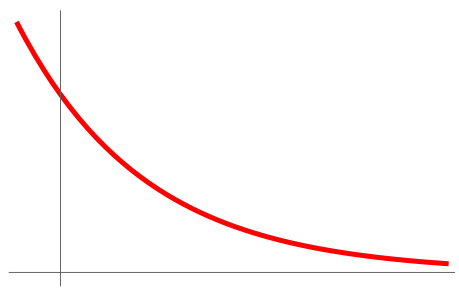
$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \tfrac{\mu}{2}\|y - x\|^2,$$

and x^* is a minimizer of f, then the Łojasiewicz inequality holds:

$$\|\nabla f(x)\|^2 \geq 2\mu\big(f(x) - f(x^*)\big).$$

Counter-example

The function $f(x) = e^{-x}$ is not uniformly convex, and the Łojasiewicz inequality is not satisfied.



The Łojasiewicz inequality

Proof.

- We minimize both sides of the uniform-convexity inequality over y:

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \tfrac{\mu}{2}\|y - x\|^2.$$

- On the LHS we get $f(x^*)$. For the RHS, we set the derivative to zero,

$$\nabla f(x) + \mu(y - x) = 0, \qquad y = x - \frac{1}{\mu}\nabla f(x),$$

and compute

$$\min_y \text{RHS} = f(x) - \frac{1}{2\mu}\|\nabla f(x)\|^2.$$

- As minimization maintains the inequality relation, we get

$$f(x^*) \geq f(x) - \frac{1}{2\mu}\|\nabla f(x)\|^2.$$

□

Exponential decay along the gradient flow

Lemma

If f satisfies the Łojasiewicz inequality for some constant μ and minimizer x^*, then f decays exponentially along its gradient flow,

$$f(x(t)) - f(x^*) \leq (f(x_0) - f(x^*))e^{-2\mu t} = O(e^{-2\mu t}).$$

Proof: See next slide.

Remark. The idea is to show for some $C > 0$ that

$$\frac{d}{dt} \log f(x(t)) \leq -2\mu.$$

Exponential decay along the gradient flow (cont.)

Proof. Without loss of generality, $f(x^*) = 0$.

- By the Łojasiewicz inequality,

$$\frac{d}{dt} \log f(x(t)) = \frac{\partial_t f(x(t))}{f(x(t))} = -\frac{\|\nabla f(x(t))\|^2}{f(x(t))} \leq -2\mu.$$

- Integration in time yields

$$\log f(x(t)) - \log f(x_0) \leq -2\mu t.$$

- Exponentiation yields $f(x(t)) \leq f(x_0)e^{-2\mu t}$. □

Exponential decay along gradient descent

Lemma

If f satisfies the Łojasiewicz inequality with some constant μ and minimizer x^*, and ∇f has Lipschitz constant $1/\alpha$, then f decays exponentially along its gradient descent with learning rate α,

$$f(x_k) - f(x^*) \leq (f(x_0) - f(x^*))(1 - \alpha\mu)^k = O(e^{-\alpha\mu k}),$$

provided that $\alpha\mu < 1$.

Remark. The idea is to show that

$$\log f(x_{k+1}) - \log f(x_k) \leq -\alpha\mu.$$

Exponential decay along gradient descent

Proof. Without loss of generality, $f(x^*) = 0$.

- The Lipschitz continuity of ∇f and the Łojasiewicz inequality imply

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) + \langle \nabla f(x_k), x_{k+1} - x_k \rangle + \frac{1}{2\alpha} \|x_{k+1} - x_k\|^2 \\ &= f(x_k) - \frac{\alpha}{2} \|\nabla f(x_k)\|^2 \\ &\leq f(x_k) - \alpha\mu f(x_k) = f(x_k)(1 - \alpha\mu). \end{aligned}$$

- Taking the logarithm, one obtains

$$\log f(x_{k+1}) - \log f(x_k) \leq \log(1 - \alpha\mu) < 0.$$

- By recursion over k, one obtains

$$\log f(x_k) \leq \log f(x_0) + k \log(1 - \alpha\mu),$$

and the result follows by exponentiation. □

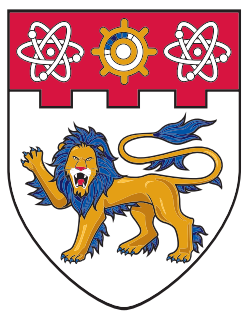
Questions to answer for yourself / discuss with friends

- Repetition: For convex or uniformly convex objectives, how fast is the decay of the objective along the gradient flow or gradient descent?
- Pronunciation: The Polish letter Ł is pronounced like the letter w in water.
- Discussion: How applicable are these results?

MH3520:Mathematics of Deep Learning

Chapter 3

Training neural networks



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

Last chapter:

- Global optimization is impossible, but local optimization can be achieved using gradient descent.

This chapter:

- How to compute gradients for training neural networks.
- Some adaptations and refinements of gradient descent.

Next chapter:

- Recap on probability theory, as a preparation for statistical learning theory.

Overview of Chapter 3

1. Training neural networks
2. Backpropagation
3. Loss landscape
4. Stochastic gradient descent

Sources for this chapter:

- Bottou, Curtis, Nocedal. Optimization methods for large scale machine learning. SIAM Review 60:2 (2018), pp. 223–311.

MH3520 Chapter 3

Part 1

Training neural networks

Learning versus learning by heart

So far, we have looked at learning as the problem of minimizing the empirical risk over all multi-layer perceptrons. The following example suggests that this may not be what we ultimately want:

Example (Learning by heart)

Consider a function which simply memorizes the given examples,

$$h(x) = \begin{cases} y_i & \text{if } x = x_i \text{ for some } i \in \{1, \ldots, n\}, \\ 0 & \text{(arbitrarily) otherwise.} \end{cases}$$

This function is a global minimizer of the empirical risk and is easy to compute (no optimization involved).

The reason for not using this function is that it doesn't generalize well to unseen data.

Generalization to unseen data

- To make sense of the term 'unseen data', let's assume that the training data consists of random samples from some probability distribution P.
- So far, we tried to minimize the empirical risk (aka. empirical loss)

$$R_n(w) := \frac{1}{n} \sum_{i=1}^{n} \ell(h_w(x_i), y_i).$$

- A better goal would be to minimize the expected risk (aka. expected loss)

$$R(w) := E[\ell(h_w(x), y)] := \int \ell(h_w(x), y)) P(dx, dy),$$

  whose minimizer by definition generalizes well to all data.
- Unfortunately, we don't know P and therefore don't know R.

Cross-validation

We will have a detailed look at this issue in the upcoming chapter on statistical learning theory. Here is how to deal with it practically:

Cross-validation:

- A simple method for tracking out-of sample performance
- Split your dataset in two parts: training data and validation data
- Train your network only on training data (never on validation data!) This amounts to empirical risk minimization.
- When you are done, check the loss on the unseen validation data.

Improving generalization capability

This is still more an art than a science.

Some recipes:

- Use stochastic training algorithms (the main topic in this chapter).
- Don't train your network all the way, but stop early.
- Regularize the problem by penalizing the size of weights, number of non-zero weights, roughness of the function represented by the network, etc.
- Choose a sufficiently 'small' class of perceptrons. (With over-parameterization, your network can still be large.)
- Choose a network architecture which fits well to the data.

Questions to answer for yourself / discuss with friends

- Repetition: Learning is more than empirical risk minimization—can you explain why?
- Repetition: What is training and validation data, and what is it used for?
- Discussion: Is a shallow ReLu network with 1-dimensional input able to learn by heart, i.e., to perfectly memorize n data points?

MH3520 Chapter 3

Part 2

Backpropagation

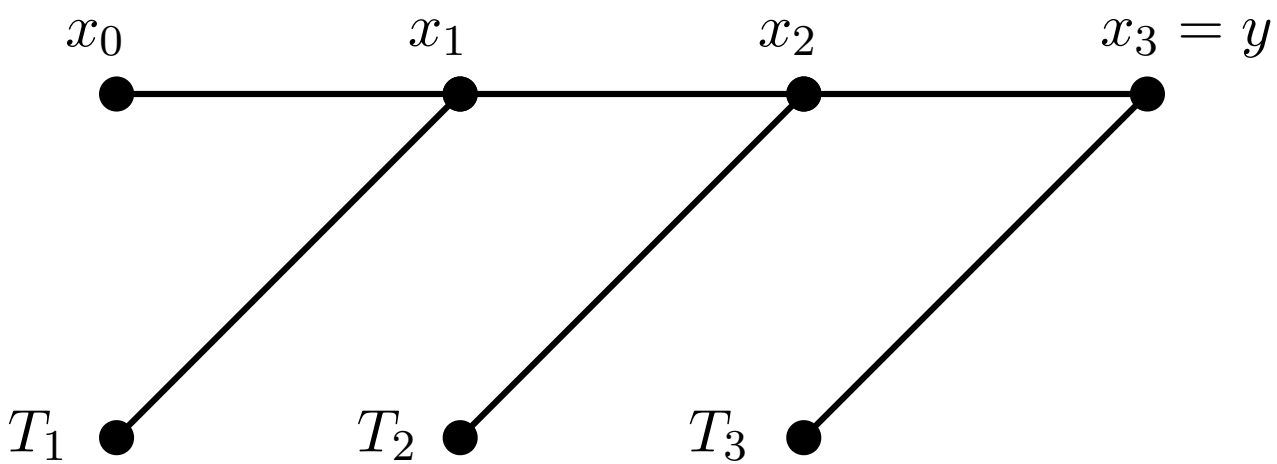
Backpropagation

Backpropagation

- Automatically computes derivatives of multi-layer perceptrons with respect to network parameters
- Mathematically speaking, it's just the chain rule
- Algorithmically speaking, it's a backwards pass through the network (hence the name)

Computational graph

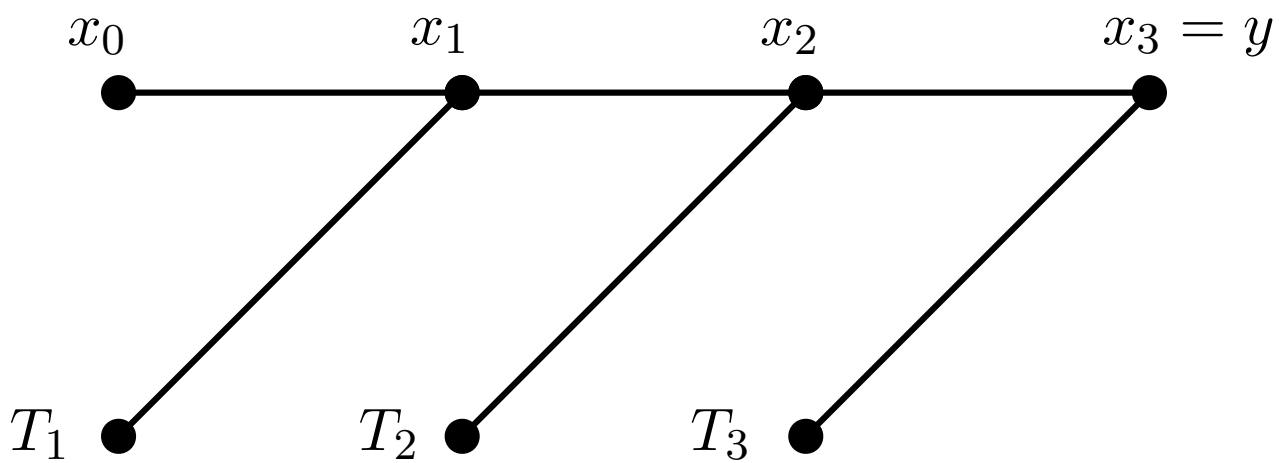
- Below is the computational graph of a multi-layer perceptron, seen as a function of the network parameters
- By the chain rule, the derivative of a composition is the composition of the derivatives
- Thus, the computational graph for the derivatives is the same as the original computational graph[1] but all maps are linear.



[1] Provided that the function values along the graph are known.

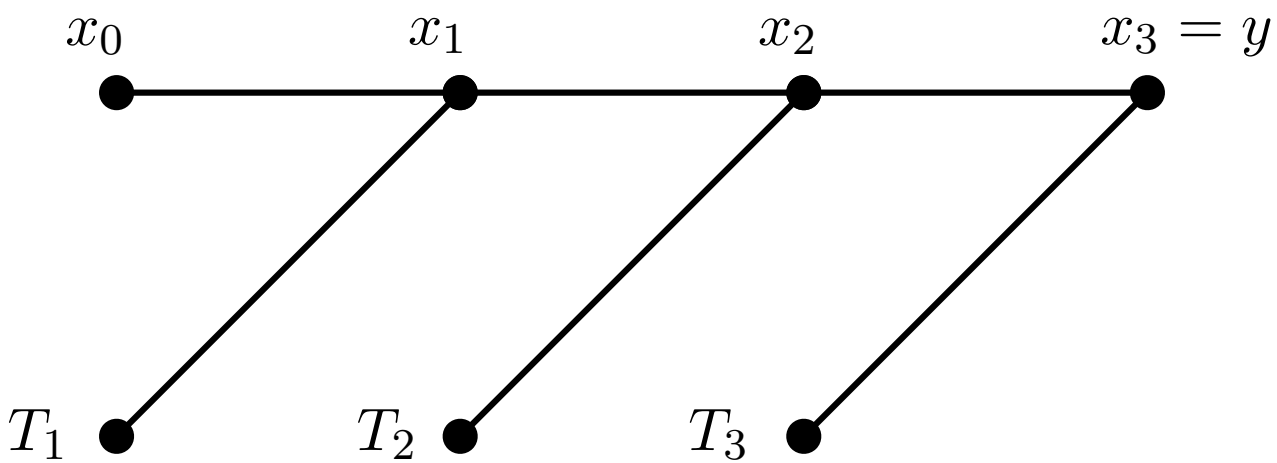
Why not forward propagation?

- In forward propagation, derivatives are computed recursively starting from the input
- This works badly if there are wide hidden layers



Backward propagation

- In backward propagation, derivatives are computed recursively starting from the output
- This works well if the output is low-dimensional



Forward and backward pass

Overall, the computation proceeds in two passes:

- Forward pass for computing function values

$$x_0, \quad x_1, \quad \ldots, \quad x_L = y.$$

- Backward pass for computing derivatives

$$\frac{\partial y}{\partial x_3}, \quad \frac{\partial y}{\partial (x_2, T_3)}, \quad \ldots, \quad \frac{\partial y}{\partial (x_1, T_2, \ldots, T_L)}, \quad \frac{\partial y}{\partial (T_1, \ldots, T_L)}.$$

Example with scalar parameters

Recall that composition of linear maps in $L(\mathbb{R},\mathbb{R}) = \mathbb{R}$ corresponds to multiplication of real numbers.

$$
\begin{aligned}
y &= x_3, & \frac{\partial y}{\partial x_3} &= 1,\\
x_3 &= w x_2, & \frac{\partial y}{\partial (x_2,w)} &= 1\cdot(w,x_2) = (w,x_2),\\
x_2 &= \rho(x_1), & \frac{\partial y}{\partial (x_1,w)} &= (w\cdot d\rho(x_1), x_2),\\
x_1 &= a x_0 + b, & \frac{\partial y}{\partial (a,b,w)} &= (w\cdot d\rho(x_1)\cdot(x_0,b), x_2)\\
& & &= (w\cdot d\rho(x_1)\cdot x_0, w\cdot d\rho(x_1)\cdot b, x_2).
\end{aligned}
$$

Gradient of the empirical loss

- We write $h_w(x)$ as a short-hand for the multi-layer perceptron with parameter $w = (T_L, \ldots, T_1)$ and input x.
- The sample loss of a data point (x_i, y_i) is the function

$$f_i(w) = \ell(h_w(x_i), y_i).$$

- The empirical risk of a dataset $(x_1, y_1), \ldots, (x_n, y_n)$ is

$$R_n(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w).$$

- As differentiation is linear, the batch gradient is the average of the sample gradients,

$$\nabla R_n(w) = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(w),$$

and each sample gradient is computed by a forward-backward pass.

Questions to answer for yourself / discuss with friends

- Repetition: What is backpropagation, and how does it work mathematically and algorithmically?
- Repetition: Describe the forward and backward passes in gradient computations.
- Check your understanding: What is the difference between a derivative, directional derivative, and partial derivative?

MH3520 Chapter 3

Part 3
Loss landscape

Loss landscape

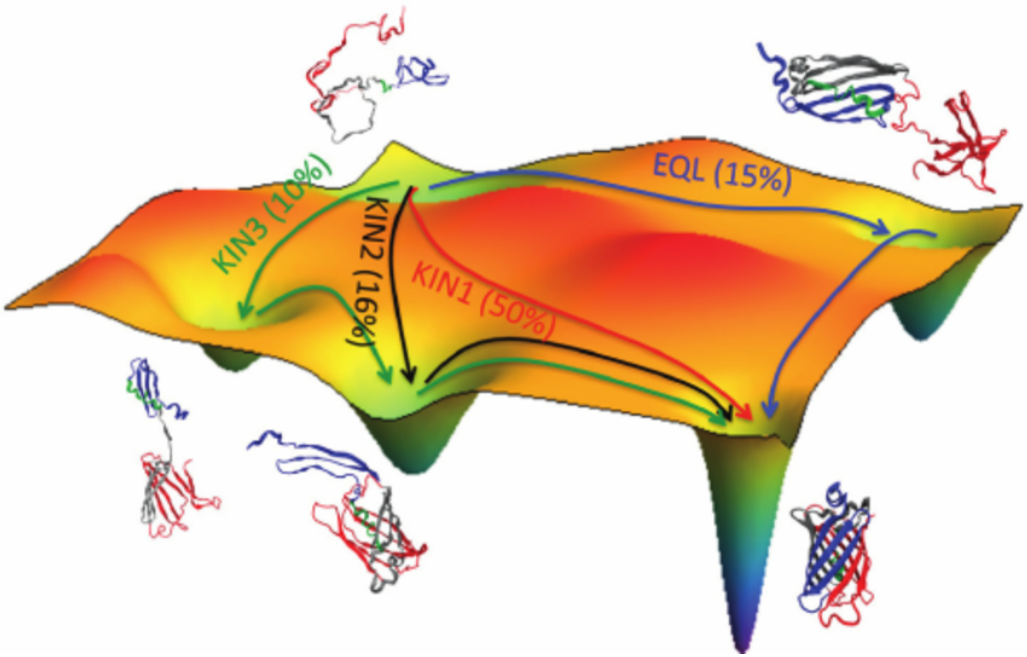
Context:

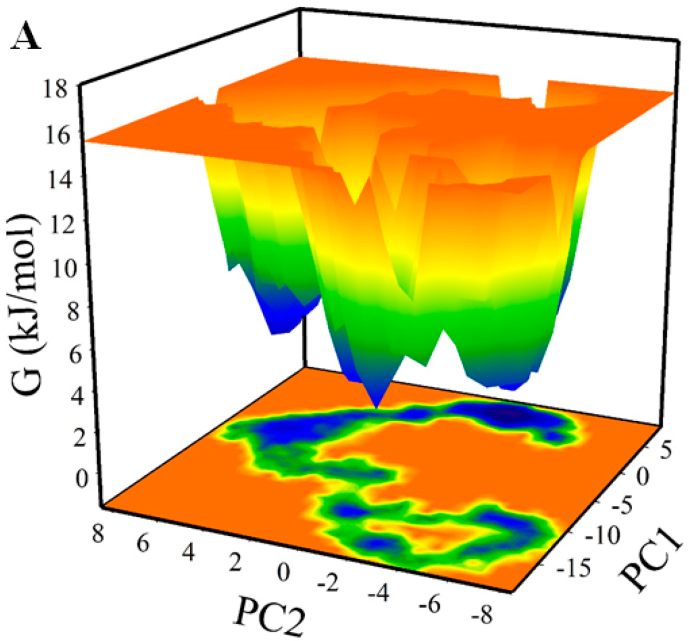
- Gradient descent algorithms work surprisingly well for training neural networks.
- To put things into perspective, no one would dream of using the same simple algorithms for protein folding.

Loss landscape:

- This is a poetic name for the graph of the loss function.
- One would expect the landscapes for deep learning and protein folding to be qualitatively different.
- Theoretical results on this are still quite scattered, and no general picture has emerged yet.

Loss landscape for protein folding

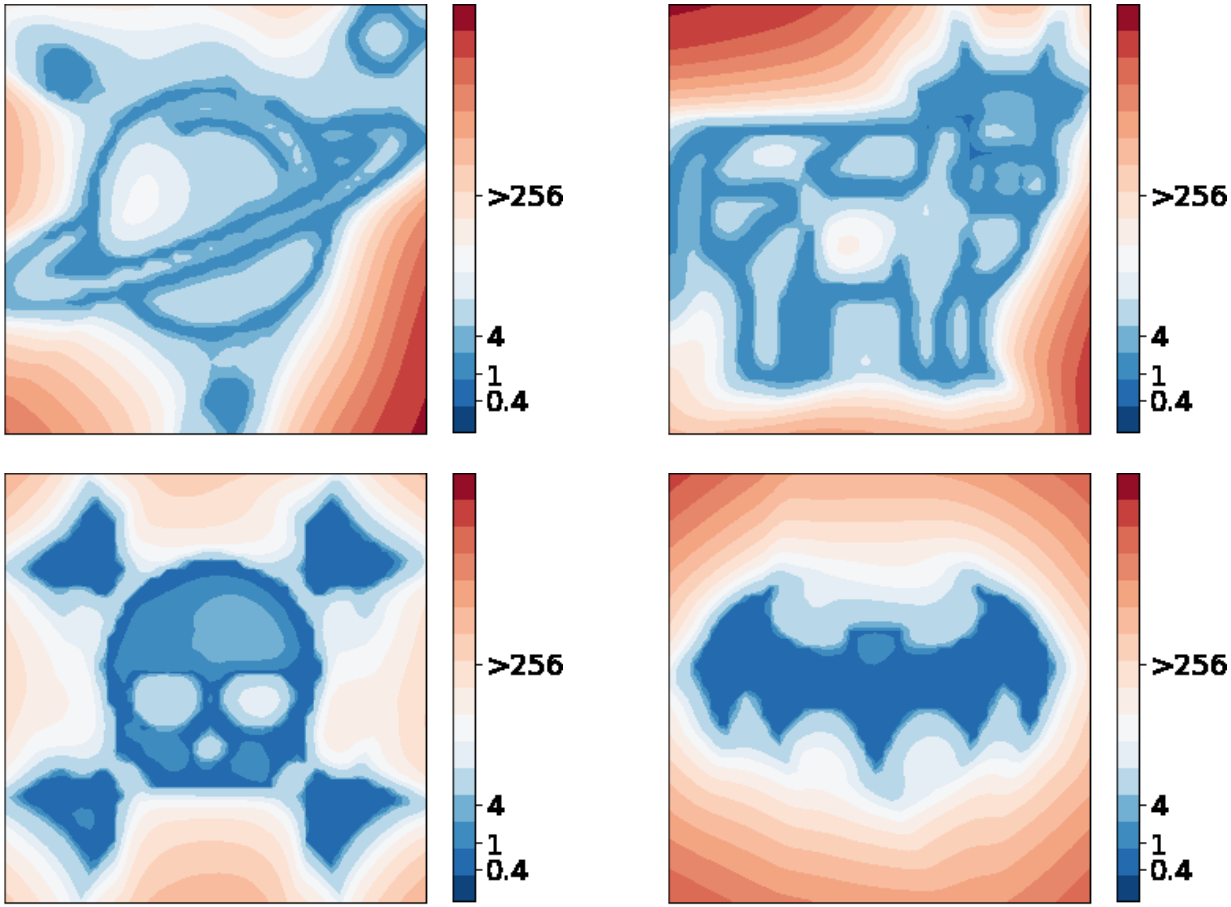




Local minima correspond to different 3-dimensional protein folding structures. There are many local minima.

[E. K. Peter] [T. Mohammad, S. Siddiqui, e.a.]

Loss landscape for deep learning

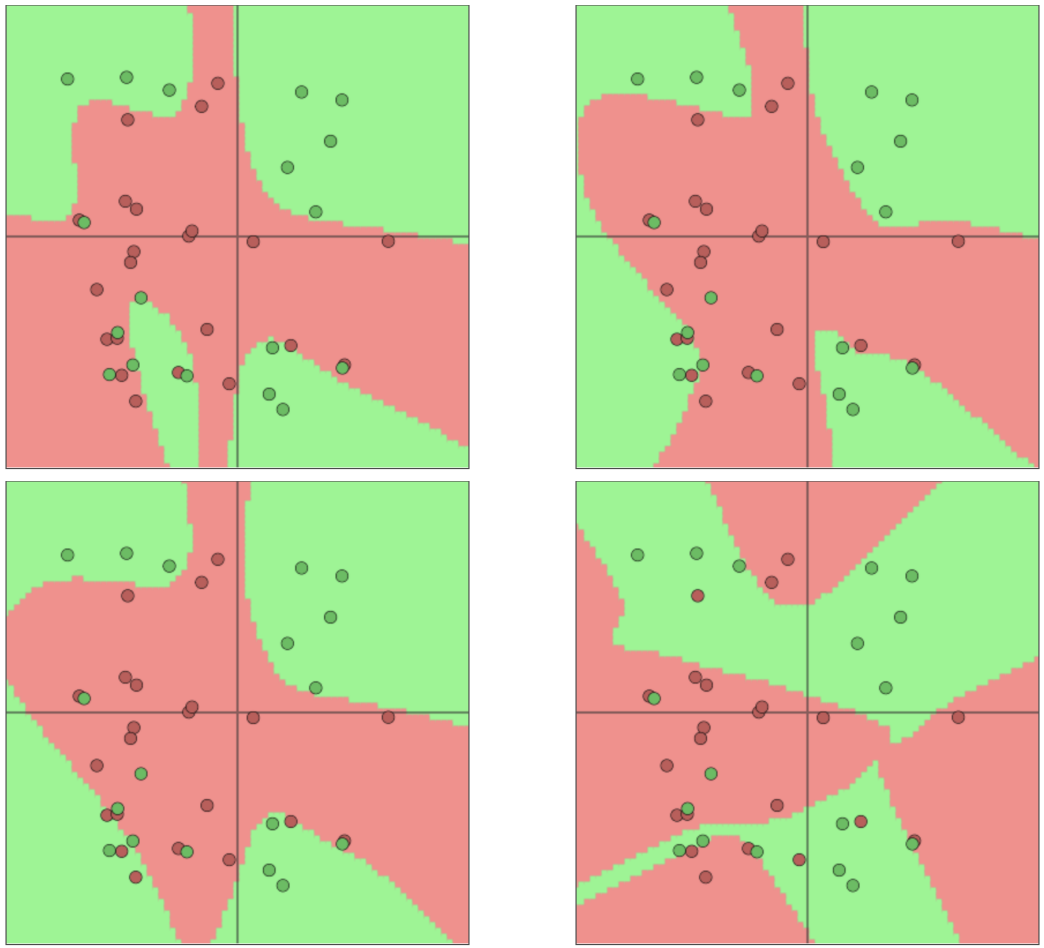


Virtually any pattern can be found somewhere in the loss surface of a deep network. Here, the loss is computed on the MNIST (top) or CIFAR10 (bottom) datasets.

[Skorokhodov Burtsev]

Getting stuck in local minima

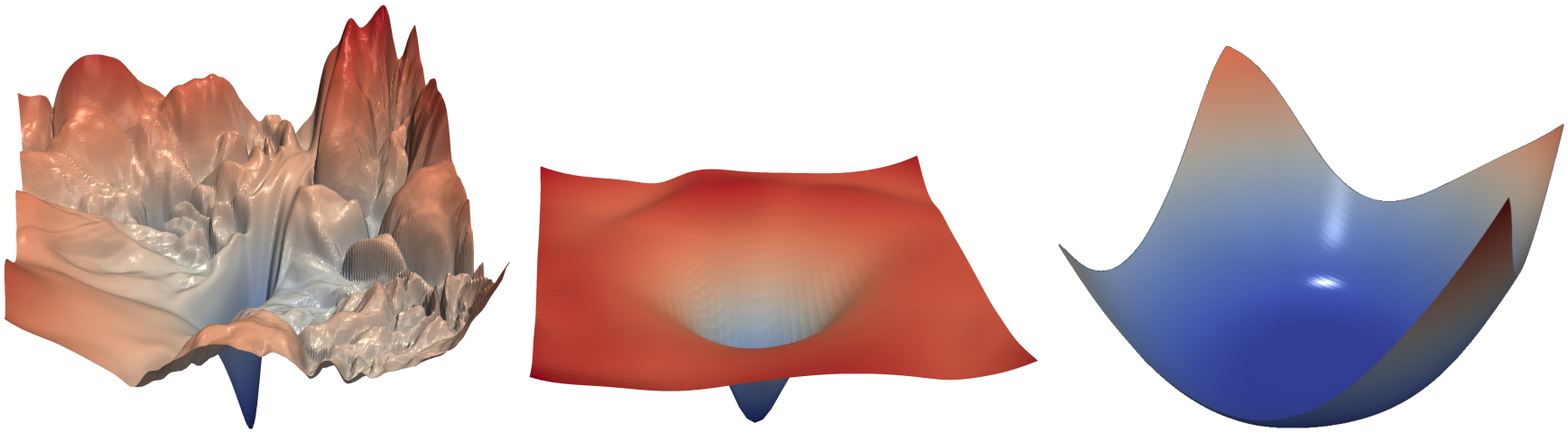
Unsurprisingly, with complicated objective functions, the training outcome depends on the network initialization:



`https://cs.stanford.edu/people/karpathy/convnetjs/demo/classify2d.html`

The network architecture matters

The loss landscape heavily depends on the choice of network architecture:



Loss functions of a residual network (left), a residual network with skip connections (center), and a dense net (right) trained on the CIFAR10 dataset.

[Li Xu e.a.]

Loss function of over-parameterized networks

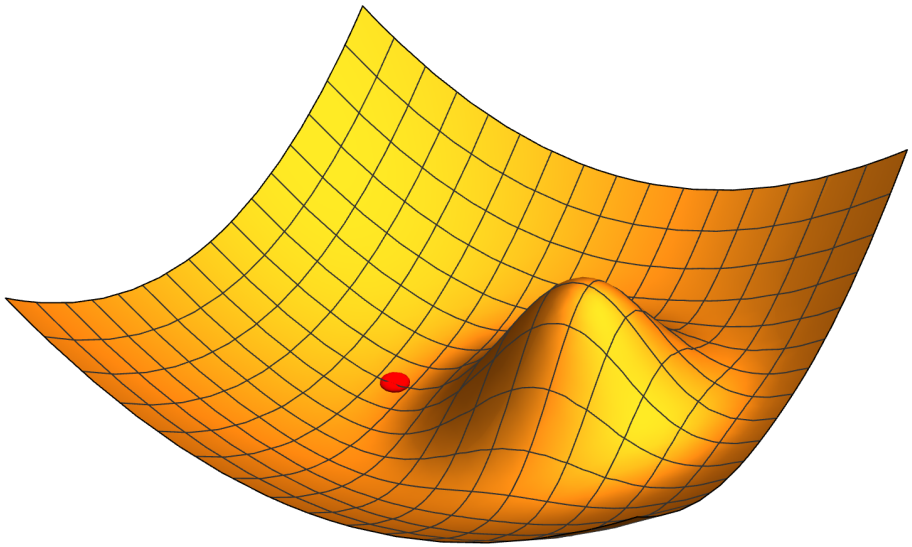
Theorem (Cooper)

Consider the empirical loss function of a neural network with smooth activation function and more parameters than training data. If the loss vanishes for some parameter configuration, then the set of global minimizers is generically a smooth manifold.

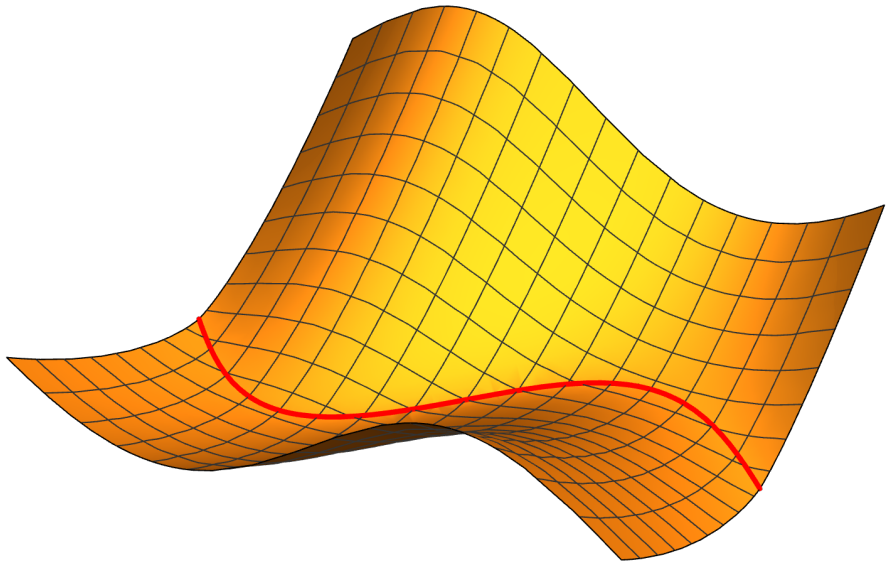
Remark.

- These smooth manifolds are higher-dimensional versions of smooth curves or surfaces.
- Their dimension is the number of network parameters minus the number of training points.
- Generically means up to an arbitrarily small change to the dataset.

Manifold of global minima

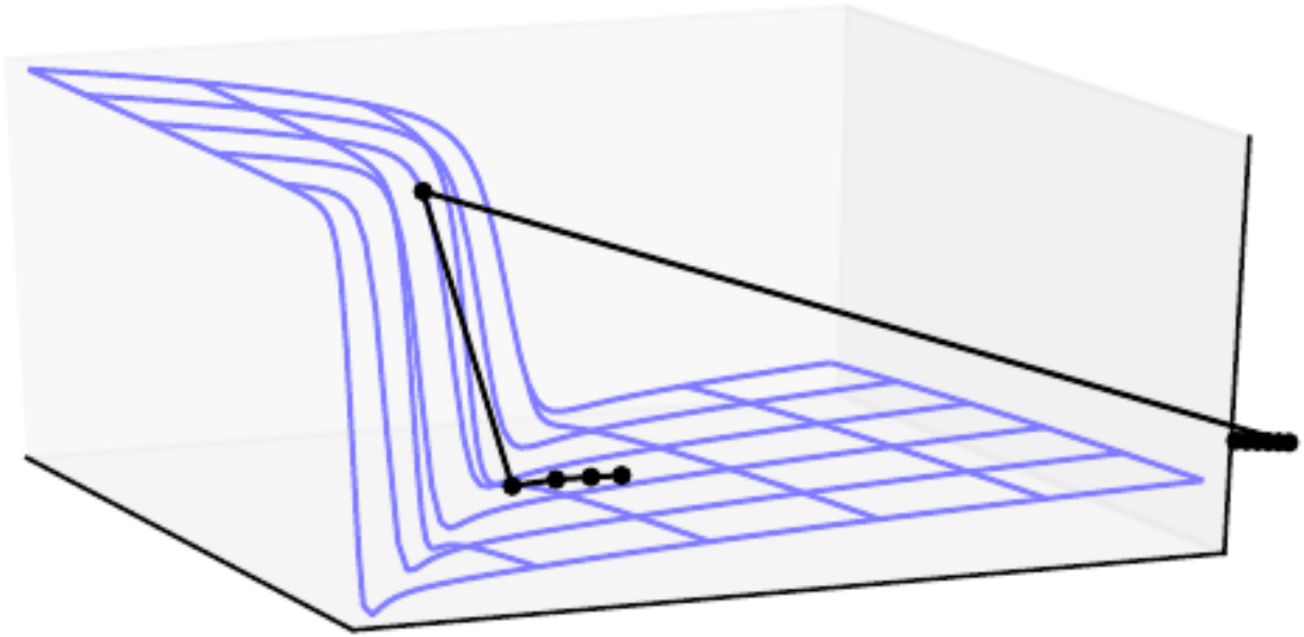


Isolated global minimum



Manifold of global minima

Cliffs in the loss landscape of deep networks



Cliffs can occur in loss landscapes of deep and recurrent networks and result in large gradients. (Remedy: gradient clipping)

Questions to answer for yourself / discuss with friends

- Repetition: What is the loss landscape of an optimization problem?
- Discussion: Can you guess how the pictures of the loss landscape might have been created? Do you think they are representative?

MH3520 Chapter 3

Part 4

Stochastic gradient descent

Mini-batching

Context:

- Recall that the batch gradient is the average of the sample gradients.

Mini-batching:

- Partition the dataset into random subsets of roughly equal size, called mini-batches
- At each step of the gradient descent, compute the gradient with respect to a mini-batch instead of the full batch
- The resulting algorithm is called stochastic gradient descent.

Stochastic gradient descent

Notation:

- Perceptron $h_w(x)$ with parameter w and input x
- Training data (x_i, y_i)
- Sample loss $f_i(w) = \ell(h_w(x_i), y_i)$

---

input : learning rate α, initial guess w, batch size m

repeat

  randomly choose m data points (x_i, y_i)

  $w \leftarrow w - \alpha \frac{1}{m} \sum_{i=1}^{m} \nabla f_i(w)$

until convergence

return w

Motivation for stochastic gradient descent

Redundancy in the data:

- If there are duplicates, then a mini-batch without these duplicates contains the same information as the full batch
- Typically, there are no strict duplicates but a good deal of approximate redundancy.

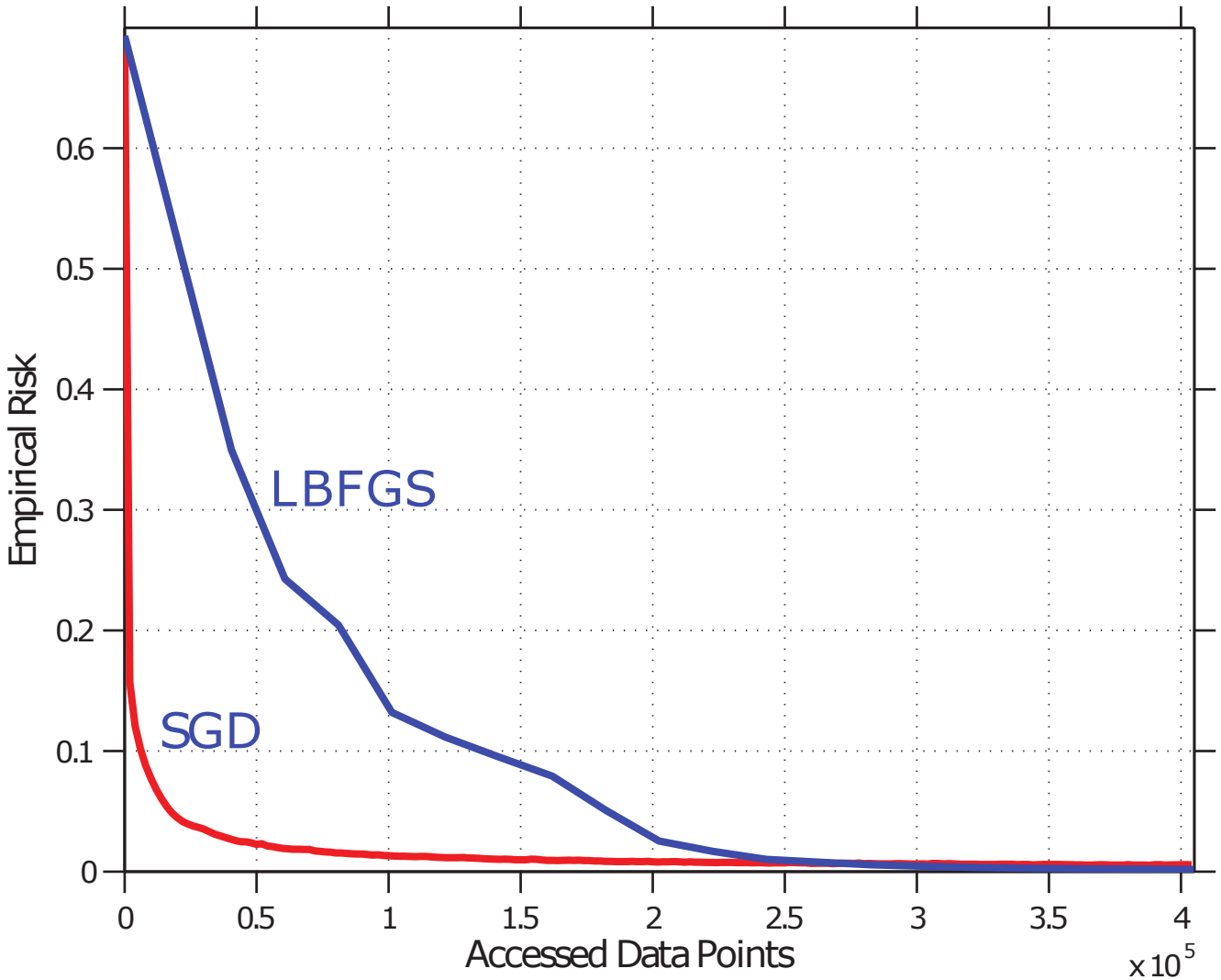
Numerical efficiency:

- Mini-batch gradients are faster to compute than batch gradients
- Stochastic gradient descent converges rapidly at the beginning
- Randomness helps to escape local minima

Good generalization:

- Suppose there were infinitely many independent mini-batches
- Then stochastic gradient descent for the empirical risk is the same as stochastic gradient descent for the expected risk

Initially rapid convergence of stochastic gradient descent



LBFGS is a variation of GD.

[Bottou, Curtis, Nocedal]

Random gradients

Standing assumption

- $f : E \to \mathbb{R}$ is a loss function on a Euclidean space E
- The Łojasiewicz inequality holds for some minimizer w^* and $\mu > 0$:

$$\|\nabla f(w)\|^2 \geq 2\mu\big(f(w) - f(w^*)\big).$$

- ∇f has Lipschitz constant $L \geq \mu$
- $\xi, \xi_1, \xi_2, \ldots$ are iid. random variables with values in some space Ξ
- $g : E \times \Xi \to E$ is a function which satisfies for some $M \geq 1$ that

$$\mathbb{E}[g(w, \xi)] = \nabla f(w), \qquad \mathbb{E}[\|g(w, \xi)\|^2] \leq M(1 + \|\nabla f(w)\|^2).$$

Remark. Think of g as a noisy version of the true gradient.

Convergence of stochastic gradient descent

Theorem

The stochastic gradient descent defined by

$$w_{k+1} = w_k - \alpha g(w_k, \xi_k), \qquad w_0 \in E,$$

with $\alpha \leq 1/(LM)$ *satisfies*

$$\mathbb{E}[f(w_k) - f(w^*)] \leq \frac{\alpha LM}{2\mu} + (1 - \alpha\mu)^k \big(F(w_0) - F(w^*) - \tfrac{\alpha LM}{2\mu}\big)$$

Remark.

- The assumptions $\alpha < 1/(LM)$, $\mu \leq L$, and $M \geq 1$ imply that $0 < \alpha\mu \leq 1$, so there is exponential convergence.
- In particular, the RHS tends to $\frac{\alpha LM}{2\mu} > 0$ as $k \to \infty$.
- For small α, $\frac{\alpha LM}{2\mu}$ is small, but $(1 - \alpha\mu)^k$ converges slower.

Convergence of stochastic gradient descent

Proof.

- The Taylor remainder bound gives

$$f(w_{k+1}) \leq f(w_k) - \alpha\langle \nabla f(w_k), g(w_k, \xi_k)\rangle + \tfrac{1}{2}\alpha^2 L\|g(w_k, \xi_k)\|^2.$$

- Take the expectation over ξ_k and use the moment bound for g:

$$\mathbb{E}[f(w_{k+1})] \leq f(w_k) - \alpha\|\nabla f(w_k)\|^2 + \tfrac{1}{2}\alpha^2 LM\big(1 + \|\nabla f(w_k)\|^2\big).$$

- The assumption $\alpha < 1/(LM)$ implies that

$$\mathbb{E}[f(w_{k+1})] \leq f(w_k) - \tfrac{1}{2}\alpha\|\nabla f(w_k)\|^2 + \tfrac{1}{2}\alpha^2 LM.$$

- Assume wlog. that $f(w^*) = 0$. By the Łojasiewicz inequality,

$$\mathbb{E}[f(w_{k+1})] \leq f(w_k) - \alpha\mu f(w_k) + \tfrac{1}{2}\alpha^2 LM.$$

- Take total expectations and subtract $\frac{\alpha LM}{2\mu}$ from both sides:

$$\begin{aligned}\mathbb{E}[f(w_{k+1}) - \tfrac{\alpha LM}{2\mu}] &\leq (1 - \alpha\mu)\mathbb{E}[f(w_k)] + \tfrac{1}{2}\alpha^2 LM - \tfrac{\alpha LM}{2\mu} \\ &= (1 - \alpha\mu)\mathbb{E}[f(w_k) - \tfrac{\alpha LM}{2\mu}].\end{aligned}$$

□

Variable learning rate

The previous theorem suggests to decrease the learning rate over time:

Theorem

Choose $\beta > 1/\mu$ and $\gamma \geq \beta LM$. Then, the stochastic gradient descent with variable learning rate $\alpha_k = \beta/(\gamma + k)$ satisfies

$$\mathbb{E}[f(w_k) - f(w^*)] \leq \frac{1}{\gamma + k} \max\left\{ \frac{\beta^2 LM}{2(\beta\mu - 1)}, \gamma\big(f(w_0) - f(w^*)\big) \right\}.$$

Proof: skipped.

Remark. Only linear instead of exponential decay, but convergence to zero.

Momentum, acceleration, and rescaling

In practice, several refinements of stochastic gradient descent are used, based on the following ideas:

- Momentum stabilizes the gradient (unless β is too large):

$$w_{k+1} = w_k - \alpha g(w_k, \xi_k) + \beta(w_k - w_{k-1})$$

- Acceleration computes the gradient after adding the momentum: [Nesterov]

$$\tilde{w}_k = w_k + \beta(w_k - w_{k-1}), \qquad w_{k+1} = \tilde{w}_k - \alpha g\big(\tilde{w}_k, \xi_k\big).$$

- Rescaling aims at equal progress along each coordinate: [RMSProp]

$$[R_k]_i = (1-\lambda)[R_{k-1}]_i + \lambda[g(w_k, \xi_k)]_i^2,$$
$$[w_{k+1}]_i = [w_k]_i - \frac{\alpha}{\sqrt{1 + [R_k]_i}}[g(w_k, \xi_k)]_i.$$

- Adaptive moment combines momentum and rescaling. [Kingma, Ba]

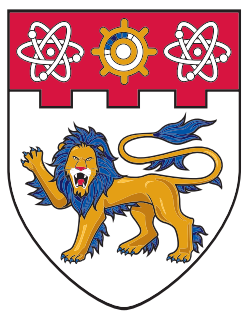
Questions to answer for yourself / discuss with friends

- Repetition: What is stochastic gradient descent, and how does it work?
- Repetition: What convergence rate can one expect for uniformly convex loss functions?
- Discussion: Momentum strategies are often described as gradient averaging strategies. Can you see why?
- Discussion: In rescaling, why would you want to preserve the norm of activations and gradients when propagated through the network?

MH3520:Mathematics of Deep Learning

Chapter 4

Basics of probability theory



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

Last chapter:

- Training neural networks by stochastic gradient descent

This chapter:

- What is length, area, and volume? What is an integral?
- What is probability? What is a random variable?

Next chapter:

- Error Bounds for Deep Learning via Statistical Learning Theory

Measure and probability theory in 3 hours!?

- Obviously, I will have to be selective.
- The story that I will present is that any integral can be obtained as a continuous linear extension of an elementary integral.
- You only need basic topology topology for this.

Overview of Chapter 4

1. Measures
2. L_0 spaces and convergence in measure
3. L_1 spaces and integration
4. L_p spaces, inequalities, and limit theorems
5. Probability

Sources for this chapter:

- Taylor: An introduction to measure and probability. Springer, 1997.

MH3520 Chapter 4

Part 1
Measures

Measure in classical geometry

This is how the ancient Greeks went about this:

- Want to know the measure (i.e., length, area, or volume) of a geometric body?
- Partition it into small pieces, whose measures you know, and add up their measures.
- If this doesn't work, compute upper and lower bounds using inscribed or circumscribed bodies.

Measure in analytical geometry

Geometric bodies are now seen as subsets of $\mathbb{R}^d$. This spells trouble:

Theorem (Banach–Tarski paradox, 1924)

The unit ball in $\mathbb{R}^3$ can be disassembled into 5 pieces, which can then be reassembled (after translating and rotating each piece) to form two disjoint bodies of the unit ball.

Remark

- Sketch of proof: `en.wikipedia.org/wiki/Banach-Tarski_paradox`
- Each piece is an incredibly complicated uncountable union of sets.
- The take-away is that such complicated sets should be considered non-measurable, and we'll define a measure only for certain 'nice' measurable sets.

Countability

Definition

A set A is countable if there exists an injective function $A \to \mathbb{N}$, and uncountable otherwise.

Remark.

- A countable set is either finite or countably infinite (i.e., in bijection with the natural numbers).
- The natural and rational numbers are countable, but the real numbers are not.

Remark. Measurability will be defined axiomatically such that countable (but not uncountable) unions are allowed; see next.

Measurable spaces

Definition

A measurable space is a set Ω together with a collection $\mathcal{F}$ of subsets of Ω, which are called measurable, such that

- The empty set $\emptyset$ and the full set Ω are measurable, i.e.,

$$\emptyset \in \mathcal{F}, \qquad \Omega \in \mathcal{F}.$$

- The countable union of measurable sets is measurable, i.e.,

$$A_1, A_2, \cdots \in \mathcal{F} \qquad \Longrightarrow \qquad \bigcup_{n \in \mathbb{N}} A_n \in \mathcal{F}$$

- The complement of a measurable set is measurable, i.e.,

$$A \in \mathcal{F} \qquad \Longrightarrow \qquad A^c := \Omega \setminus A \in \mathcal{F}.$$

Any collection $\mathcal{F}$ with these three properties is called a sigma algebra.

For comparison: Topological spaces

Definition

A topological space is a set Ω together with a collection $\mathcal{O}$ of subsets of Ω, which are called open, such that

- The empty set $\emptyset$ and the full set Ω are open, i.e.,

$$\emptyset \in \mathcal{O}, \qquad \Omega \in \mathcal{O}.$$

- The union of open sets is open, i.e., for any index set I,

$$A_i \in I \text{ for all } i \in I \quad \Longrightarrow \quad \bigcup_{i \in I} A_i \in \mathcal{O}.$$

- The finite intersection of open sets is open, i.e., for any $n \in \mathbb{N}$,

$$A_1, \ldots, A_n \in \mathcal{O} \quad \Longrightarrow \quad \bigcap_{i=1}^{n} A_i \in \mathcal{O}.$$

Any collection $\mathcal{O}$ with these three properties is called a topology.

Smallest and largest sigma algebra

Here is the smallest possible sigma algebra:

Example

The trivial sigma algebra on Ω is $\mathcal{F} = \{\emptyset, \Omega\}$.

Here is the largest one, which is commonly used on countable sets:

Example

The discrete sigma algebra on Ω is the power set $\mathcal{F} = 2^{\Omega} = \{A : A \subseteq \Omega\}$.

Here is the one commonly used on topological spaces:

Example

The Borel sigma algebra on a topological space is the smallest sigma algebra which contains all open sets.

For comparison: Examples of topological spaces

Here is the smallest possible topology:

Example

The trivial topology on Ω is $\mathcal{O} = \{\emptyset, \Omega\}$.

Here is the largest one, which is commonly used on countable sets:

Example

The discrete topology on Ω is the power set $\mathcal{O} = 2^{\Omega} = \{A : A \subseteq \Omega\}$.

And here is the topology of a normed or metric space:

Example

An open set in a normed or metric space is defined as a union of open balls.

Measurable functions

Definition

A function $f : \Omega_1 \to \Omega_2$ between measurable spaces $(\Omega_1, \mathcal{F}_1)$ and $(\Omega_2, \mathcal{F}_2)$ is measurable if pre-images of measurable sets are measurable, i.e.,

$$\forall A \in \mathcal{F}_2 : \qquad f^{-1}(A) \in \mathcal{F}_1.$$

Remark. In practice, it is rare to find non-measurable functions. Nevertheless, the concept is important for ruling out Banach–Tarski style paradoxa.

For comparison: Continuous functions

Definition

A function $f : \Omega_1 \to \Omega_2$ between topological spaces $(\Omega_1, \mathcal{O}_1)$ and $(\Omega_2, \mathcal{O}_2)$ is continuous if pre-images of open sets are open, i.e.,

$$\forall A \in \mathcal{O}_2 : \qquad f^{-1}(A) \in \mathcal{O}_1.$$

Remark. This definition of continuity coincides with the usual one for normed and metric spaces.

Continuous functions are measurable

Lemma

Continuous functions are Borel measurable.

Proof.

- Let $f : \Omega_1 \to \Omega_2$ for sets Ω_i with topologies $\mathcal{O}_i$, where $i \in \{1, 2\}$.
- The Borel sigma algebras are given by $\mathcal{F}_i = \sigma(\mathcal{O}_i)$, where $\sigma(\mathcal{O}_i)$ denotes the smallest sigma algebra containing $\mathcal{O}_i$.
- It follows that

$$f^{-1}(\mathcal{F}_2) = f^{-1}(\sigma(\mathcal{O}_2)) = \sigma(f^{-1}(\mathcal{O}_2)) \subseteq \sigma(\mathcal{O}_1) = \mathcal{F}_1. \qquad \square$$

New measurable spaces from old ones

Definition

The product of measurable spaces $(\Omega_i, \mathcal{F}_i)$ indexed by $i \in I$ is the set $\Omega := \prod_{i \in I} \Omega_i$ endowed with the product sigma algebra $\prod_{i \in I} \mathcal{F}_i$, i.e., the smallest sigma algebra such that all projections $\mathrm{pr}_i : \Omega \to \Omega_i$ are measurable.

Remark. The product sigma algebra is the smallest sigma algebra which contains alls sets of the form $A_1 \times A_2 \times A_3 \times \ldots$, where $A_i \in \mathcal{F}_i$ with $A_i = \Omega_i$ for all but one i.

Definition

A subset Ω' of Ω is a measurable space when endowed with the trace sigma algebra, i.e., the smallest sigma algebra such that the inclusion $i : \Omega' \to \Omega$ is measurable.

Remark. The trace sigma algebra consists of the sets $A \cap \Omega'$ for $A \in \mathcal{F}$.

New measurable functions from old ones

Lemma

- *The composition* $f_1 \circ f_2$ *of measurable functions is measurable.*
- *Linear combinations* $\lambda_1 f_1 + \lambda_2 f_2$ *and products* $f_1 f_2$ *of real-valued measurable functions are measurable.*
- *The point-wise limit* $\lim_{n\to\infty} f_n$ *of a sequence of measurable functions* f_n *is measurable.*
- *The collection* $f = (f_i)_{i\in I}$ *of measurable functions* $f_i : \Omega \to \Omega_i$ *indexed by a set* I *is a measurable function with values in the product of measurable spaces* $(\prod_{i\in I} \Omega_i, \prod_{i\in I} \mathcal{F}_i)$.

Measures

Definition

A measure on a measurable space $(\Omega, \mathcal{F})$ is a function $\mu : \mathcal{F} \to [0, \infty]$ such that

- The empty set has zero measure, i.e.,

$$\mu(\emptyset) = 0.$$

- The measure is countably additive, i.e.,

$$\mu\Big(\bigcup_{n\in\mathbb{N}} A_n\Big) = \sum_{n\in\mathbb{N}} \mu(A_n),$$

  for all mutually disjoint sets $A_1, A_2, \cdots \in \mathcal{F}$.

The triple $(\Omega, \mathcal{F}, \mu)$ is called measure space.

Remark. Note that non-measurable sets don't get measured.

Discrete measures

Example

The Dirac measure at a point $x \in \Omega$ is the map

$$\delta_x : \mathcal{F} \to [0, \infty], \qquad \delta_x(A) = \mathbb{1}_A(x) = \begin{cases} 1, & x \in A, \\ 0, & x \notin A. \end{cases}$$

Example

The counting measure on Ω is

$$\mu := \sum_{x \in \Omega} \delta_x, \qquad \mu(A) = (\text{the number of elements of } A).$$

Remark. Counting measures are commonly used on finite and countable sets Ω. On uncountable sets, they serve as examples for all kind of pathological behavior.

Borel measure

Example

The Borel measure on $\mathbb{R}^d$ is the unique measure μ on the Borel sigma algebra of $\mathbb{R}^d$ such that

$$\mu\left(\prod_{i=1}^{d}(a_i, b_i)\right) = \prod_{i=1}^{d}(b_i - a_i), \qquad \text{for all real numbers } a_i \leq b_i.$$

Remark.

- Existence and uniqueness of the Borel measure is a deep result in measure theory, which resolves the Banach–Tarski paradox.
- By not giving a proof, I'm hiding several dozens of pages on measure extension theory.

Monotonicity and continuity properties

Lemma

Let $(\Omega, \mathcal{F}, \mu)$ be a measure space, and let $A_1, A_2, \cdots \in \mathcal{F}$.

- *Monotonicity:* $A_1 \subseteq A_2 \Longrightarrow \mu(A_1) \leq \mu(A_2)$.
- *Continuity from below:* *if* $A_1 \subseteq A_2 \subseteq \ldots$, *then*

$$\mu\Big(\bigcup_{n\in\mathbb{N}} A_n\Big) = \lim_{n\to\infty} \mu(A_n) = \sup_{n\in\mathbb{N}} \mu(A_n).$$

- *Continuity from above:* *if* $\mu(A_1) < \infty$ *and* $A_1 \supseteq A_2 \supseteq \ldots$, *then*

$$\mu\Big(\bigcap_{n\in\mathbb{N}} A_n\Big) = \lim_{n\to\infty} \mu(A_n) = \inf_{n\in\mathbb{N}} \mu(A_n)$$

Null sets

Definition

A null set of a measure $\mu : \mathcal{F} \to [0, \infty]$ is a set $A \in \mathcal{F}$ with $\mu(A) = 0$. A property which holds on the complement of a null set is said to hold almost everywhere.

Example

Functions (f_n) converge almost everywhere to a function f if $\lim_{n\to\infty} f_n(x) = f(x)$ holds for all x in the complement of a null set.

Example

Functions f and g coincide almost everywhere if $f(x) = g(x)$ holds for all x in the complement of a null set.

Completion

Definition

The completion of $\mathcal{F}$ with respect to μ is the sigma algebra

$$\overline{\mathcal{F}} := \Big\{ A \cup N : A \in \mathcal{F} \text{ and } N \text{ is a subset of a null set} \Big\}.$$

The completion of μ is the measure $\overline{\mu}$ on $\overline{\mathcal{F}}$ defined by $\overline{\mu}(A \cup N) := \mu(A)$.

Example

The completion of the Borel measure is called Lebesgue measure.

New measures from old ones

The space of measures on a given measure space is a convex cone:

Lemma

If $\mu_1, \mu_2 : \mathcal{F} \to [0, \infty]$ are measures, then for any $\lambda_1, \lambda_2 \in [0, \infty)$, the positive linear combination

$$\lambda_1 \mu_1 + \lambda_2 \mu_2 \quad \textit{is a measure.}$$

Remark.

- This is sometimes called mixture of measures.
- If one considers signed measures, then these form a linear space.

New measures from old ones

Finite products of measures are measures:

Lemma

The product of two measures $\mu_{1,2}$ on measure spaces $(\Omega_{1,2}, \mathcal{F}_{1,2})$ is the unique measure μ on $(\Omega_1 \times \Omega_2, \mathcal{F}_1 \times \mathcal{F}_2)$ such that

$$\forall A_1 \in \mathcal{F}_1, \forall A_2 \in \mathcal{F}_2 : \qquad \mu(A_1 \times A_2) = \mu_1(A_1) \times \mu_2(A_2).$$

Remark.

- Existence of product measures requires measure extension theory; this is beyond the present scope.
- Infinite products of measures are well-defined only for probability measures.

New measures from old ones

The image of a measure under a measurable map is a measure:

Lemma

If $(\Omega_1, \mathcal{F}_1)$ and $(\Omega_2, \mathcal{F}_2)$ are measurable spaces, $f : \Omega_1 \to \Omega_2$ is measurable, and $\mu_1 : \mathcal{F}_1 \to [0, \infty]$ is a measure, then

$$\mu_2 : \Omega_2 \to [0, \infty], \qquad \mu_2(A_2) := \mu_1(f^{-1}(A_2)),$$

is a measure, which is called the image measure of μ_1 under f.

Remark.

- If you think of f as a transport map, then μ_2 is the mass of μ_1 after transportation by f.
- A common notation for the image measure is $\mu_2 = f_*(\mu_1)$.

Questions to answer for yourself / discuss with friends

- Repetition: What is a measurable space, and what are measurable functions?
- Check your understanding: If the elements of Ω are points, what are the elements of $\mathcal{F}$: points, sets of points, sets of sets of points, etc?
- Discussion: Can you think of a topology that one could put on the space of measures?
- Discussion: In Riemann integrals $\int f(x)\,dx$, can we interpret dx as a measure?

MH3520 Chapter 4

Part 2

L_0 spaces and convergence in measure

Equality almost everywhere

Definition

Two measurable functions f, g defined on a measure space $(\Omega, \mathcal{F}, \mu)$ are said to be equal almost everywhere if $\{x : f(x) \neq g(x)\}$ is a null set.

Remark.

- We will identify functions which are equal almost everywhere.
- Formally, we will work modulo the equivalence relation of equality almost everywhere.
- For all of our purposes, we can treat equivalence classes of functions as functions.

Measurable vector-valued functions

Standing assumption. $(\Omega, \mathcal{F}, \mu)$ is a measure space, and B is a separable Banach space endowed with its Borel sigma algebra.

Definition

$L_0(\Omega, B)$ denotes the vector space of measurable functions $f : \Omega \to B$ modulo equivalence almost everywhere.

Remark.

- For simplicity, just think of $B = \mathbb{R}$.
- Our goal is to endow $L_0(\Omega, B)$ with a complete metric such that addition and scalar multiplication are continuous.
- Separability of B is a technical detail, which you may safely ignore, and completeness of B will be used for completeness of $L_0(\Omega, B)$.

Metric space

Definition

A metric on a set S is a function $d : S \times S \to [0, \infty)$ such that for all $x, y, z \in S$,

- Non-degeneracy: $d(x, y) = 0$ if and only if $x = y$,
- Symmetry: $d(x, y) = d(y, x)$, and
- Triangle inequality: $d(x, z) \leq d(x, y) + d(y, z)$.

The tuple (S, d) is called metric space.

Example

For any $p \in (0, 1)$, $\mathbb{R}^d$ is a metric space with $d(x, y) := \left(\sum_{i=1}^{d} |x_i|^p \right)$.

In the following we do have a metric defined for all measurable functions, while it is not widely used.

Normed (vector) space

Definition

A norm on a (vector) space E is a function $\|\cdot\| : E \to [0, \infty)$ such that for all $x, y, z \in E$ and $\lambda \in \mathbb{R}$,

- Non-degeneracy: $\|x\| = 0$ if and only if $x = 0$,
- Absolute homogeneity: $\|\lambda x\| = |\lambda| \|x\|$, and
- Triangle inequality: $\|x + y\| \leq \|x\| + \|y\|$.

The tuple $(E, \|\cdot\|)$ is called a normed vector space.

Example

For any $p \in [1, \infty)$, the p-norm and supremum norm on $\mathbb{R}^d$ are given by

$$\|x\|_p := \left(\sum_{i=1}^{d} |x_i|^p \right)^{1/p}, \qquad \|x\|_\infty := \max_{i \in \{1,\ldots,d\}} |x_i|$$

Covering numbers and compactness

Theorem

A metric space is totally bounded if all of its covering numbers are finite, i.e., if for every $\epsilon > 0$, it can be covered by finitely many ϵ-balls.

Theorem

A metric space is compact if and only if it is complete and totally bounded.

Remark. In this sense, covering numbers can be seen as a quantification of compactness.

Size of a measurable function

How 'often' does a function take large values, say, of norm greater than r:

Definition

For any $f \in L_0(\Omega, B)$ and $r \geq 0$, we define the gauge

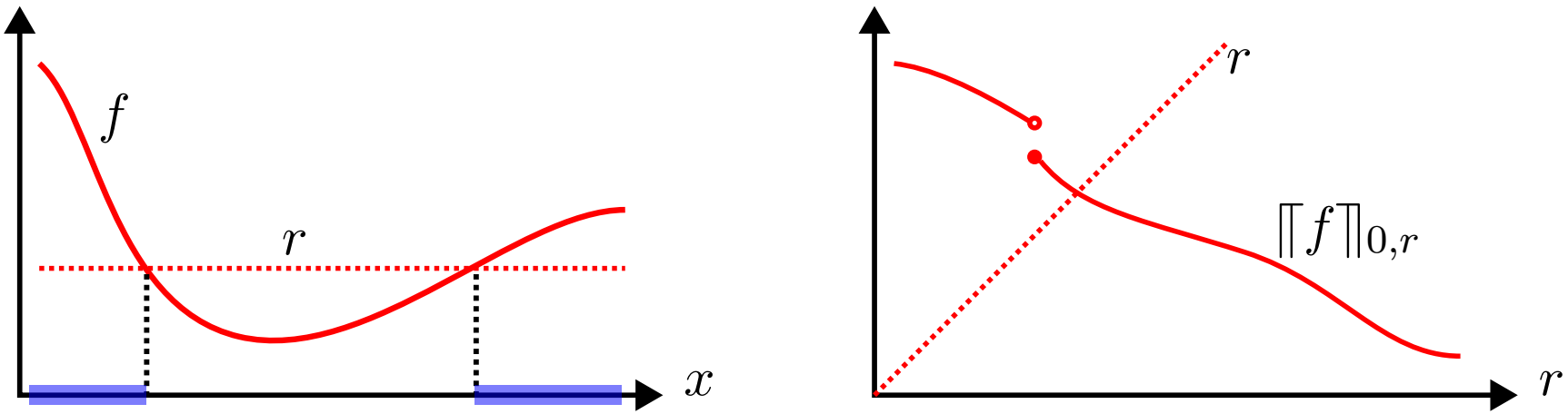
$$\llbracket f \rrbracket_{0,r} := \mu\Big(\big\{x \in \Omega : \|f(x)\| > r\big\}\Big).$$

Remark.

- The term gauge has no precise meaning.
- $\llbracket \cdot \rrbracket_{0,r}$ is not a norm (hence the different notation) and not a metric, but a variation of the triangle inequality holds:

$$\llbracket f + g \rrbracket_{0,r+s} \leq \llbracket f \rrbracket_{0,r} + \llbracket g \rrbracket_{0,s}.$$

Size of a measurable function (cont., optional)



Left: $[\![f]\!]_{0,r}$ measures the area (blue) where $\|f\| > r$.
Right: $[\![f]\!]_{0,r}$ is right-continuous and decreasing in r.

L_0 metric on measurable functions

Theorem

$L_0(\Omega, B)$ is a complete metric space with the L_0 metric:

$$d(f,g) := \llbracket f-g \rrbracket_0, \qquad \llbracket f \rrbracket_0 := \min\{1\} \cup \left\{ r \geq 0 : \llbracket f \rrbracket_{0,r} \leq r \right\}.$$

Addition and scalar multiplication are continuous w/r to this metric.

Sketch of proof.

- The $\min$ exists because $\llbracket f \rrbracket_{0,r}$ is right-continuous decreasing in r.
- $\llbracket f-g \rrbracket_0 = 0$ implies $f = g$ almost everywhere.
- The triangle inequality $\llbracket f+g \rrbracket_0 \leq \llbracket f \rrbracket_0 + \llbracket g \rrbracket_0$ holds. This implies the triangle inequality for the metric and the continuity of addition.
- Continuity of scalar multiplication $\lim_{\lambda \to 0} \llbracket \lambda f \rrbracket_0 = 0$ follows from the continuity from above of μ.
- The completeness proof takes around 2 pages and is skipped. □

Convergence in measure

Definition

Convergence in measure is convergence with respect to the L_0 metric.

Lemma

(f_n) converges to f in measure if and only if for all $r > 0$:

$$\lim_{n\to\infty} \llbracket f_n - f \rrbracket_{0,r} = 0.$$

Proof. Unroll the definition of $\llbracket \cdot \rrbracket_0$ as follows:

- For any given ϵ and sufficiently large n, $\llbracket f_n - f \rrbracket_0 \leq \epsilon$.
- This implies $\llbracket f_n - f \rrbracket_{0,\epsilon} \leq \epsilon$.
- By monotonicity, this implies $\llbracket f_n - f \rrbracket_{0,r} \leq \epsilon$, for any $r \geq \epsilon$. □

Questions to answer for yourself / discuss with friends

- Repetition: Define the space $L_0(\Omega, B)$ and the L_0 metric thereon.
- Check your understanding: An element of $L_0(\Omega, B)$ is a function, a set of functions, a set of sets of functions, etc?
- Check your understanding: What is a metric? When is a metric called complete?
- Complete the sentence: A function f is ϵ-close to a function g in $L_0(\Omega, B)$ if...
- Discussion: If B was merely a metric space, what would $L_0(\Omega, B)$ be?

MH3520 Chapter 4

Part 3
L_1 spaces and integration

Elementary integrands

Standing assumption: $(\Omega, \mathcal{F}, \mu)$ is a measure space, and B is a separable Banach space.

Definition

- An elementary function is a measurable function $f : \Omega \to B$ with finite range.
- An elementary integrand is an elementary function f such that $\mu(\{x : f(x) \neq 0\}) < \infty$.

Remark. Every elementary function is of the canonical form

$$f = \sum_{i=1}^{n} y_i \mathbb{1}_{A_i}, \qquad \text{where } n \in \mathbb{N}, y_i \in B, A_i \in \mathcal{F}, A_i \cap A_j = \emptyset,$$

if $i \neq j$. For elementary integrands, additionally $\mu(A_i) < \infty$.

Size of elementary integrands

Definition

We write $L_{1,\mathrm{elem}}(\Omega, B)$ for the space of elementary integrands modulo equality almost everywhere, and we endow this space with the L_1-norm

$$\|f\|_1 = \sum_{i=1}^{n} \|y_i\| \mu(A_i), \qquad \text{for any } f = \sum_{i=1}^{n} y_i \mathbb{1}_{A_i} \text{ in canonical form.}$$

Remark.

- Here, we need that elementary functions are measurable.
- This is a norm because $\|f\|_1 = 0$ if and only if f vanishes almost everywhere.

Elementary integral

Definition

The elementary integral of an elementary integrand

$$f = \sum_{i=1}^{n} y_i \mathbb{1}_{A_i} \qquad \text{(in canonical form)}$$

is defined as

$$\int_\Omega f\mu := \sum_{i=1}^{n} y_i \mu(A_i) \in B.$$

Remark.

- Again, we need that elementary functions are measurable.
- Another notation is $\int_\Omega f(x)\mu(dx)$.

Continuity of the elementary integral

Lemma

The elementary integral is a continuous linear function

$$L_{1,\mathrm{elem}}(\Omega, B) \ni f \mapsto \int_\Omega f\mu \in B$$

with operator norm bounded by 1.

Proof.

$$\left\| \int f\mu \right\| = \left\| \sum_i y_i \mu(A_i) \right\| \leq \sum_i \|y_i\| \mu(A_i) = \|f\|_1$$

□

Continuous linear extension

We want to extend the elementary integral to larger classes of integrands.

Theorem

Let V be a dense subspace of a normed vector space E, and let F be a complete normed vector space. Then, any continuous linear function $T : V \to F$ extends uniquely to a continuous linear function $T : E \to F$ with the same operator norm.

Sketch of proof. Set $Tx := \lim_n Tx_n$ if $x_n \in V$ converges to $x \in E$. □

Remark.

- The statement is true also for metric vector spaces or topological vector spaces, and the metric or norm may even be degenerate.
- Any continuous linear function is uniformly continuous, and the statement holds also (potentially nonlinear) uniformly continuous functions.

Completion

Theorem

Any normed vector space E has a completion $\overline{E}$, i.e.,

- *$\overline{E}$ is a complete normed vector space, and*
- *E is densely and isometrically embedded in $\overline{E}$.*

The completion is unique up to isometry.

Sketch of proof. Define $\overline{E}$ as the set of Cauchy sequences modulo null sequences in E. □

Example

$\mathbb{R}$ is the completion of $\mathbb{Q}$.

Bochner integral

Definition

$L_1(\Omega, B)$ is the completion of $L_{1,\mathrm{elem}}(\Omega, B)$ with respect to $\|\cdot\|_1$.

By the theorem on continuous linear extensions, we get immediately:

Theorem

The elementary integral extends uniquely to a continuous linear function

$$L_1(\Omega, B) \ni f \mapsto \int_\Omega f\mu \in B,$$

which is called Bochner integral.

Remark.

- This is the most general integral you'll encounter for a while.
- The Lebesgue integral is the special case $B = \mathbb{R}$.

Characterization of integrands

The abstractly defined L_1 space is actually a function space:

Theorem

$L_1(\Omega, B)$ is continuously embedded in $L_0(\Omega, B)$.

Remark.

- The embedding is the continuous linear extension of the embedding

$$L_{1,\mathrm{elem}}(\Omega, B) \to L_{0,\mathrm{elem}}(\Omega, B).$$

- The key point is that the extension is injective. (In general, it is not.)
- By not giving a proof, I'm hiding a couple of pages of genuine measure theory (mostly Fatou's lemma).

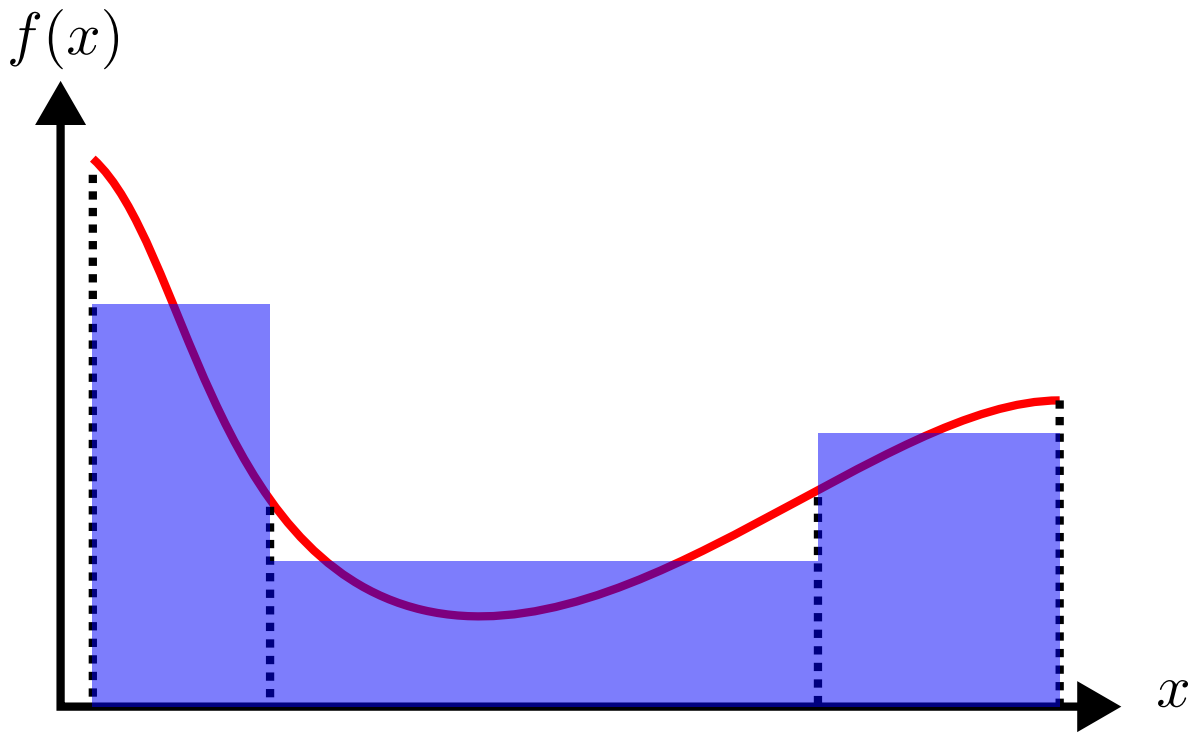
Characterization of integrands (cont.)

Corollary

$f \in L_1(\Omega, B)$ if and only if $\|f\| \in L_1(\Omega, \mathbb{R})$.

Proof. The map $f \mapsto \|f\|$ is an isometry $L_{1,\mathrm{elem}}(\Omega, B) \to L_{1,\mathrm{elem}}(\Omega, \mathbb{R})$. Thus, its continuous (nonlinear!) extension is also isometric. □

Lebesgue versus Riemann integration



- Riemann integration requires control over all partitions (even bad ones), whereas Lebesgue integration requires control over some partition (which may be chosen optimally).
- Hence, Lebesgue integration is more general, but Riemann integration provides stronger error bounds for numerically computing the integral.

Questions to answer for yourself / discuss with friends

- Repetition: Define the space $L_1(\Omega, B)$ and the L_1 norm thereon.
- Check your understanding: What is the relation between the Lebesgue integral and the Lebesgue measure?
- Check your understanding: Is $\int_\Omega \mathbb{1}_A \mu$ equal to $\mu(A)$?
- Discussion: Can you design a numerical integrator, which takes as input arbitrary functions $f \in L_1(\Omega, B)$ and returns the (approximate) integral $\int_\Omega f \mu$?

MH3520 Chapter 4

Part 4
L_p spaces, inequalities, and limit theorems

L_p spaces

Standing assumption. $(\Omega, \mathcal{F}, \mu)$ is a measure space, and B is a separable Banach space.

Definition

- For any $p \in (0, \infty)$, $L_p(\Omega, B)$ is the set of all $f \in L_0(\Omega, B)$ such that $\|f\|^p \in L_1(\Omega, \mathbb{R})$.
- $L_\infty(\Omega, B)$ is the set of all $f \in L_0(\Omega, B)$ such that $\|f\|$ is, up to a null set, bounded by a constant.

Remark. If $B = \mathbb{R}$, one often abbreviates $L_p(\Omega, \mathbb{R})$ to $L_p(\Omega)$.

Size of L_p functions

Definition

- For any $p \in (0,1)$, $L_p(\Omega, B)$ carries the complete metric

$$d(f,g) = \llbracket f - g \rrbracket_p, \qquad \llbracket f \rrbracket_p := \int_\Omega \|f\|^p \mu.$$

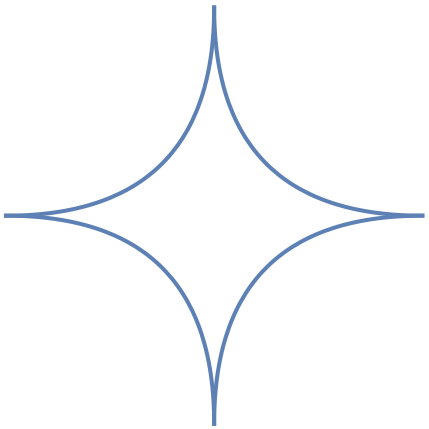
- For any $p \in [1,\infty)$, $L_p(\Omega, B)$ carries the complete norm

$$\|f\|_p := \Big(\int_\Omega \|f\|^p \mu \Big)^{1/p}.$$

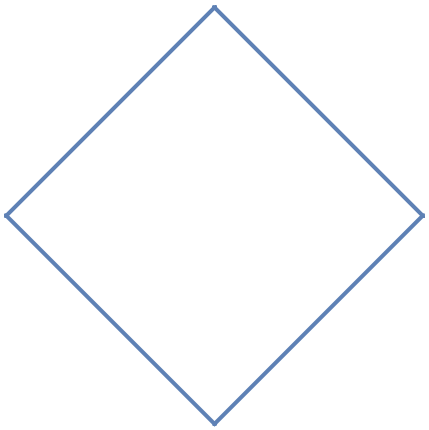
- $L_\infty(\Omega, B)$ carries the complete norm

$$\|f\|_\infty := \min \big\{ r \geq 0 : \|f\| > r \text{ almost everywhere} \big\}.$$

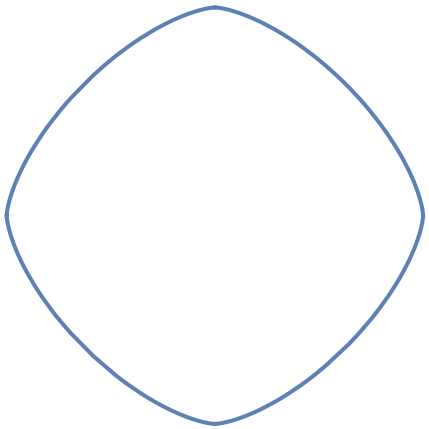
Unit spheres in $L_p(\{0,1\}) \cong \mathbb{R}^2$



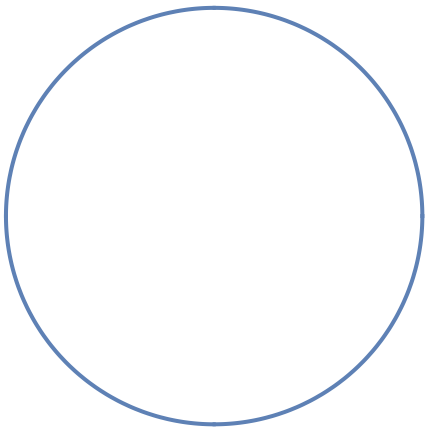
$p = 1/2$



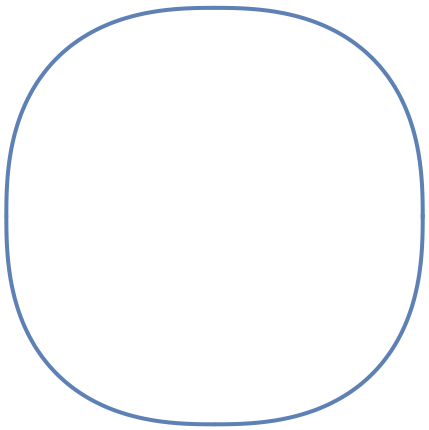
$p = 1$



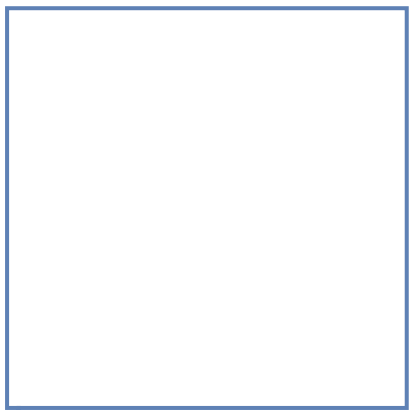
$p = 3/2$



$p = 2$



$p = 3/2$



$p = \infty$

Monotonicity

Lemma

For any $f, g \in L_1(\Omega, \mathbb{R})$,

$$f \leq g \qquad \Longrightarrow \qquad \int_\Omega f\mu \leq \int_\Omega g\mu.$$

Minkowski's integral inequality

In its simplest form, Minkowski's integral inequality is:

Lemma

For any $f \in L_1(\Omega, B)$,

$$\left\| \int_\Omega f\mu \right\| \leq \int_\Omega \|f\|\mu.$$

This is just the defining inequality in the construction of the integral.

Chebychev's inequality

Lemma

For any $p \in (0, \infty)$ *and* $f \in L_p(\Omega, B)$,

$$[\![f]\!]_{0,r} \leq \mu\big(\{x \in \Omega : \|f(x)\| \geq r\}\big) \leq \frac{1}{r^p} \int_\Omega \|f\|^p \mu.$$

Remark. The special case $p = 1$ is called Markov's inequality.

Jensen's inequality

Lemma

Assume that $\mu(\Omega) = 1$, *and let* $\varphi : \mathbb{R} \to \mathbb{R}$ *be a convex function. Then, one has for any* $f \in L_0(\Omega, \mathbb{R})$ *that*

$$\varphi \circ f \in L_1(\Omega, \mathbb{R}) \quad \Longrightarrow \quad f \in L_1(\Omega, \mathbb{R}) \text{ with } \varphi\Big(\int_\Omega f\mu \Big) \leq \int_\Omega (\varphi \circ f)\mu.$$

Remark.

- Convex functions are continuous and therefore measurable.
- Jensen's inequality can be applied to finite measures by first normalizing their total mass to $\mu(\Omega) = 1$.

Embeddings of L_p spaces

Corollary

Assume that $\mu(\Omega) < \infty$. Then, for any $0 \leq p \leq q \leq \infty$, $L_q(\Omega, B)$ is continuously and densely embedded in $L_p(\Omega, B)$.

Proof. Normalize the measure to $\mu(\Omega) = 1$. Then, the continuity of the embedding follows from Jensen's inequality applied to $\int_\Omega \|f\|^p \mu$, and the density of the embedding follows from the density of elementary integrands. □

Careful: The embeddings are wrong for infinite measures.

Cauchy-Schwarz inequality

Lemma

If B is a Hilbert space, then $L_2(\Omega, B)$ is a Hilbert space, and the Cauchy-Schwarz inequality holds:

$$\langle f, g\rangle_2 := \int_\Omega \langle f, g\rangle \mu \leq \sqrt{\int_\Omega \|f\|^2 \mu} \sqrt{\int_\Omega \|g\|^2 \mu} =: \|f\|_2 \|g\|_2.$$

Hölder's inequality

Lemma

Let $p, q, r \in [1, \infty]$ be Hölder conjugates, i.e., $\frac{1}{p} + \frac{1}{q} = \frac{1}{r}$. Then, the product of any $f \in L_p(\Omega, \mathbb{R})$ and $g \in L_q(\Omega, \mathbb{R})$ belongs to $L_r(\Omega, \mathbb{R})$, and

$$\|fg\|_r \leq \|f\|_p \|g\|_q.$$

Remark.

- This extends to $p, q, r \in (0, \infty]$ if μ is finite (or sigma-finite).
- For $p = q = 2$ one gets the the Cauchy–Schwarz inequality in $L_2(\Omega, \mathbb{R})$.

Monotone convergence

Theorem

Let (f_n) be a monotonically increasing sequence of real-valued measurable functions, which converges almost everywhere to a function $f \in L_1(\Omega, \mathbb{R})$. Then, (f_n) converges to f in $L_1(\Omega, \mathbb{R})$.

Remark. By the continuity of the integral, this implies

$$\lim_{n\to\infty} \int_\Omega f_n \mu = \int_\Omega f \mu.$$

Remark. There is no such theorem for Riemann integrals.

Dominated convergence

Theorem

Let (f_n) be a sequence in $L_1(\Omega, E)$ which converges almost everywhere to $f \in L_1(\Omega, E)$. Assume that some $g \in L_1(\Omega, \mathbb{R})$ dominates all f_n, i.e.,

$$\forall n \in \mathbb{N}: \qquad \|f_n(x)\| \leq g(x) \qquad \text{for almost every } x \in \Omega.$$

Then (f_n) converges to f in $L_1(\Omega, E)$.

Remark. By the continuity of the integral, this implies

$$\lim_{n\to\infty} \int_\Omega f_n \mu = \int_\Omega f \mu.$$

Remark. There is no such theorem for Riemann integrals.

Questions to answer for yourself / discuss with friends

- Repetition: Define the space $L_p(\Omega, B)$ and the L_p norm (or metric) thereon.
- Repetition: The dominated convergence theorem is among the most important properties of the Lebesgue integral. What does it say?
- Discussion: Some of the previous results use the normed structure and some others the ordered structure of $\mathbb{R}$—which ones?
- Discussion: Can you now make sense of the integral in the fundamental theorem of calculus:

$$\forall f \in C^1([a,b], B): \qquad f(b) - f(a) = \int_a^b df(x)\, dx.$$

MH3520 Chapter 4

Part 5
Probability

Axioms of probability

Definition

- A probability space is a measure space $(\Omega, \mathcal{F}, \mathbb{P})$ with a probability measure $\mathbb{P}$, i.e., a measure of total mass $\mathbb{P}(\Omega) = 1$.
- An event is an element of the sigma algebra $\mathcal{F}$.
- A random variable is a measurable function on Ω.

Remark.

- This axiomatization of probability is due to A. Kolmogorov (1933).
- As an aside, the axiomatization of causality is due to J. Pearl (2000).
- All of this is quite recent!

Standing assumption. $(\Omega, \mathcal{F}, \mathbb{P})$ is a probability space, on which all random variables are defined unless specified otherwise. Moreover, $(B, \mathcal{B})$, $(B_1, \mathcal{B}_1)$, etc. are a measurable spaces.

Terminology

Measure theory	Probability theory
measure	probability
measurable set	event
measurable function	random variable
image measure	distribution of the r.v.
almost everywhere	almost surely
convergence a.e.	convergence a.s.
convergence in measure	convergence in probability
integral	mean, expectation
L_1-norm	first moment
L_2-norm	second moment

Distributions

Typically, one does not specify the probability space, but merely the (joint) distribution of all random variables of interest:

Definition

- The distribution of a random variable $X : \Omega \to B$ is the image measure $X_*\mathbb{P}$, i.e., the unique probability measure $P : \mathcal{B} \to [0, \infty]$ such that $P(A) = \mathbb{P}(X^{-1}(A))$.
- The joint distribution of a family of random variables $X_i : \Omega \to B_i$ is the distribution of $X : \Omega \to \prod_i B_i$.
- The marginal distributions of a family of random variables X_i is the the family of distributions of X_i.

Remark. For any specified distribution P on $(B, \mathcal{B})$, one may construct a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and a random variable $X : \Omega \to B$ with the distribution P. (Hint: let X be the identity map on $\Omega := B$.)

Densities

To specify a distribution P, one often starts from a measure μ (e.g. the counting measure on a countable set or the Borel measure on Euclidean space) and multiplies it with a density f:

Theorem

Let $(B, \mathcal{B}, \mu)$ be a measure space. Then, any non-negative $f \in L_1(B, \mathbb{R})$ with $\|f\|_1 = 1$ defines a probability measure $P := f\mu$ as follows:

$$P : \mathcal{B} \to [0, \infty), \qquad P(A) := \int_B \mathbb{1}_A f \mu \in [0, \infty).$$

f is called the Radon–Nikodym density of P with respect to μ.

Proof: next slide.

Densities (cont.)

Proof. For any mutually disjoint sets $A_1, A_2, \cdots \in \mathcal{B}$, by the dominated convergence theorem,

$$\sum_{i=1}^{\infty} P(A_i) = \lim_{n\to\infty} \sum_{i=1}^{n} \int_B \mathbb{1}_{A_i} f\mu = \lim_{n\to\infty} \int_B \sum_{i=1}^{n} \mathbb{1}_{A_i} f\mu$$

$$= \int_B \lim_{n\to\infty} \sum_{i=1}^{n} \mathbb{1}_{A_i} f\mu = \int_B \mathbb{1}_{\bigcup_i A_i} f\mu = P\Big(\bigcup_i A_i\Big). \qquad \square$$

Examples of discrete probability measures

Bernoulli distribution

$B = \{0, 1\}$, $\mathcal{B}$ is the discrete sigma algebra, μ is the counting measure, and $P = f\mu$ with $f(1) = p$ and $f(0) = 1 - p$ for some $p \in [0, 1]$.

Binomial distribution

$B = \{0, \ldots, n\}$, $\mathcal{B}$ is the discrete sigma algebra, μ is the counting measure, and $P = f\mu$ with $f(k) = \binom{n}{k} p^k (1-p)^{n-k}$ for some $p \in [0, 1]$.

Uniform distribution

$B = \{1, \ldots, n\}$, $\mathcal{B}$ is the discrete sigma algebra, μ is the counting measure, and $P = \frac{1}{n}\mu$.

Examples of continuous probability measures

Normal distribution

$B = \mathbb{R}$, $\mathcal{B}$ is the Borel sigma algebra, μ is the Borel measure, and $P = f\mu$ with $f(x) = (2\pi\sigma^2)^{-1/2} e^{-\frac{1}{2}(x-\mu)^2/\sigma^2}$ for some $\mu \in \mathbb{R}$ and $\sigma \in (0, \infty)$.

Multi-variate normal distribution

$B = \mathbb{R}^d$, $\mathcal{B}$ is the Borel sigma algebra, μ is the Borel measure, and $P = f\mu$ with $f(x) = (\det 2\pi\Sigma)^{-1/2} e^{-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)}$ for some $\mu \in \mathbb{R}^d$ and positive definite symmetric $\Sigma \in \mathbb{R}^{d\times d}$.

Uniform distribution

$B = [0, 1]$, $\mathcal{B}$ is the Borel sigma algebra, and P is the Borel measure restricted[2] to $[0, 1]$.

[2] If A is Borel-measurable in $[0, 1]$, then A is Borel-measurable in $\mathbb{R}$.

Sigma algebras generated by random variables

Sigma algebras encode information contained in random variables:

Definition

The sigma algebra generated by a random variable $X : \Omega \to B$ is

$$\sigma(X) := X^{-1}(\mathcal{B}) := \left\{X^{-1}(A) : A \in \mathcal{B}\right\}.$$

Remark. As X was assumed to be measurable, $\sigma(X)$ is contained in $\mathcal{F}$.

Lemma

A random variable Y is measurable with respect to the sigma algebra generated by X if and only if $Y = f \circ X$ for some measurable function f.

Independence

Definition

- Two events $A_1, A_2 \in \mathcal{F}$ are independent if $P[A_1 \cap A_2] = P[A_1]P[A_2]$.
- Two sigma algebras $\mathcal{F}_1, \mathcal{F}_2$ are independent if any events $A_1 \in \mathcal{F}_1$ and $A_2 \in \mathcal{F}_2$ are independent.
- Two random variables $X_{1,2} : \Omega \to B_{1,2}$ are independent if their generated sigma algebras $\sigma(X_{1,2}) = X_{1,2}^{-1}(\mathcal{B}_{1,2})$ are independent.

Lemma

Random variables X_1 and X_2 as above are independent if and only if their joint distribution is a product measure on $B_1 \times B_2$.

Product measures

To specify the distribution of independent random variables, one has to construct products of probability measures. This is always possible:

Lemma

The product of probability measures P_i on $(B_i, \mathcal{B}_i)$, indexed by a set I, is the unique probability measure P on $(\prod_{i\in I} B_i, \prod_{i\in I} \mathcal{B}_i)$ such that

$$P\Big(\prod_{i\in I} A_i\Big) = \prod_{i\in I} P(A_i),$$

where $A_i \in \mathcal{B}_i$ is equal to all of B_i for all but finitely many $i \in I$.

Remark: For general measures (with total mass different from 1), infinite products may fail to exist.

Moments

Standing assumption. From now on, let B be a separable Banach space.

Definition

- The first moment of $X \in L_1(\Omega, B)$ is $\|X\|_1$.
- The second moment of $X \in L_2(\Omega, B)$ is $\|X\|_2$.
- More generally, for any $p \in [1, \infty)$, the p-th moment of $X \in L_p(\Omega, B)$ is $\|X\|_p$.

Mean, variance, and covariance

Definition

- The mean or expectation of $X \in L_1(\Omega, B)$ is $\mathbb{E}[X] := \int_\Omega X\mathbb{P} \in B$.
- The covariance of $X, Y \in L_2(\Omega, B)$ is the linear operator

$$\mathrm{Cov}[X,Y] : B^* \to B, \quad \mathrm{Cov}[X,Y](\alpha) = \mathbb{E}\big[\alpha(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])\big].$$

- The variance of $X \in L_2(\Omega, B)$ is $\mathrm{Cov}[X, X]$.

Remark.

- The square brackets $[\cdot]$ have no special meaning.
- The notation $\mathbb{E}$ is bad when there is more than one measure around.
- The variance and covariance are operators (think: matrices), but for scalar random variables, they are just numbers.

Conditional expectation

Definition

Let $X \in L_1(\Omega, B)$, and let $\mathcal{G}$ be a sigma algebra contained in $\mathcal{F}$.

- The conditional expectation of X given $\mathcal{G}$ is the unique $\mathcal{G}$-measurable random variable $\mathbb{E}[X|\mathcal{G}] \in L_1(\Omega, B)$ satisfying

$$\forall A \in \mathcal{G}: \qquad \mathbb{E}[\mathbb{1}_A E[X|\mathcal{G}]] = \mathbb{E}[\mathbb{1}_A X].$$

- Similarly, for any random variable Y, $\mathbb{E}[X|Y]$ is defined as $\mathbb{E}[X|\sigma(Y)]$, where $\sigma(Y)$ is the sigma algebra generated by Y.

Remark.

- The conditional expectation a linear contraction on $L_1(\Omega, B)$. If B is a Hilbert space, then it is an orthogonal projection on $L_2(\Omega, B)$.
- $\mathbb{E}[X|Y]$ is $\sigma(Y)$-measurable and thus a function of Y.

Computing expectations via distributions

Lemma

Let $X : \Omega \to B$ be a random variable with distribution $P = X_* \mathbb{P}$, and let $f : B \to C$ be a measurable function with values in a separable Banach space C. Then $f \circ X$ is $\mathbb{P}$-integrable if and only if f is P-integrable, and

$$\int_\Omega (f \circ X)\, \mathbb{P} = \int_B f\, P.$$

Remark. Both integrals are expectations (with respect to $\mathbb{P}$ and P, resp.)

Proof.

- For indicator functions f, this is the defining property of the image measure.
- For elementary integrands f, this follows by linearity.
- For P-integrable integrands f, this follows by approximation. □

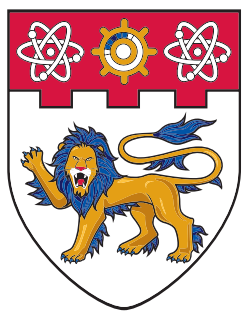
Questions to answer for yourself / discuss with friends

- Repetition: State Kolmogorov's axioms of probability.
- Repetition: What is the relation between expectation and integral?
- Check your understanding: Is the expectation of a random variable a random variable? What about the conditional expectation?
- Transfer: If the data points in supervised learning are independent, are the steps of the gradient descent independent?

MH3520:Mathematics of Deep Learning

Chapter 5

Statistical learning theory



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

Last chapter:

- Introduction to probability theory

This chapter:

- Error bounds for general learning algorithms
- As a special case, error bounds for training neural networks

Next chapter:

- Universality of neural networks

Overview of Chapter 5

1. Introduction to statistical learning
2. Empirical risk minimization and related algorithms
3. Error decompositions
4. Error trade-offs (optional)
5. Error bounds
6. Concentration inequalities
7. Bounds on the uniform generalization error

Sources for this chapter:

- Cucker, Smale (2002): On the mathematical foundations of learning. Bulletin of the American mathematical society 39.1: pp. 1–49.
- Cucker and Zhou (2007): Learning theory—An approximation theory viewpoint. Vol. 24. Cambridge University Press.

MH3520 Chapter 5

Part 1
Introduction to statistical learning

Learning

Learning or, more precisely, inductive inference:

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

Of course, this definition is very general and could be taken more or less as the goal of all natural sciences...

Learning theory versus machine learning

Goals of learning theory versus machine learning:

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

Nothing is more practical than a good theory. *[Vapnik]*

Statistical learning theory

Main assumption of statistical learning theory:

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

Definition

Transfer learning (TL) is a technique in machine learning in which knowledge learned from a task is re-used in order to boost performance on a related task. For example, for image classification, knowledge gained while learning to recognize cars could be applied when trying to recognize trucks.

Formalization

- Input and output spaces: measurable spaces $\mathcal{X}$ and $\mathcal{Y}$.
- Loss function: a measurable function $\ell\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$.
- Hypothesis class (aka. model class): a set $\mathcal{H}$ of measurable functions $h\colon \mathcal{X} \to \mathcal{Y}$.
- 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 $h \in \mathcal{H}$ with low or minimal risk (aka. test or generalization risk)

$$R(h) := \int \ell(h(x), y) P(dx, dy)$$

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

Remarks

Applications:

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

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.
- Check your understanding: What is the difference between $\mathbb{P}$ and P, and similarly between $\mathbb{E}$ and E?

MH3520 Chapter 5

Part 2

Empirical risk minimization and related algorithms

Risk versus empirical risk

Risk: Recall that...

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

$$R(h) := \int \ell(h(x), y) P(dx, dy)$$

over all h in the hypothesis class $\mathcal{H}$.

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

Empirical risk:

- As a substitute, define the empirical risk

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

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):

$$h_n \in \arg\min_{h \in \mathcal{H}} R_n(h).$$

Structural risk minimization:

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

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:

$$h_n \in \arg\min_{h \in \mathcal{H}} R_n(h) + \|h\|^2,$$

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

Algorithms (cont.)

Maximum likelihood:

$$h_n \in \arg\max_{h\in\mathcal{H}} e^{-R_n(h)} p(h) = \arg\min_{h\in\mathcal{H}} R_n(h) - \log p(h),$$

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

Posterior mean:

$$h_n = \frac{1}{Z_n} \int_{\mathcal{H}} h e^{-R_n(h)} p(h) \mu(dh),$$

where $Z_n := \int_{\mathcal{H}} e^{-R_n(h)} p(h) \mu(dh)$ is a normalizing factor.

Gibbs sampling:

$$h_n \sim \frac{1}{Z_n} e^{-R_n} p\mu.$$

Questions to answer for yourself / discuss with friends

- Repetition: What is the empirical measure, and what is empirical risk minimization?
- Repetition: How is it related to regularization and maximum likelihood estimation?
- Transfer: What optimization algorithms could be used to solve the empirical risk minimization problem?
- Transfer: Based on your knowledge in statistics, how fast do you expect the empirical risk $R_n(h)$ to converge to $R(h)$, for fixed $h \in \mathcal{H}$?

MH3520 Chapter 5

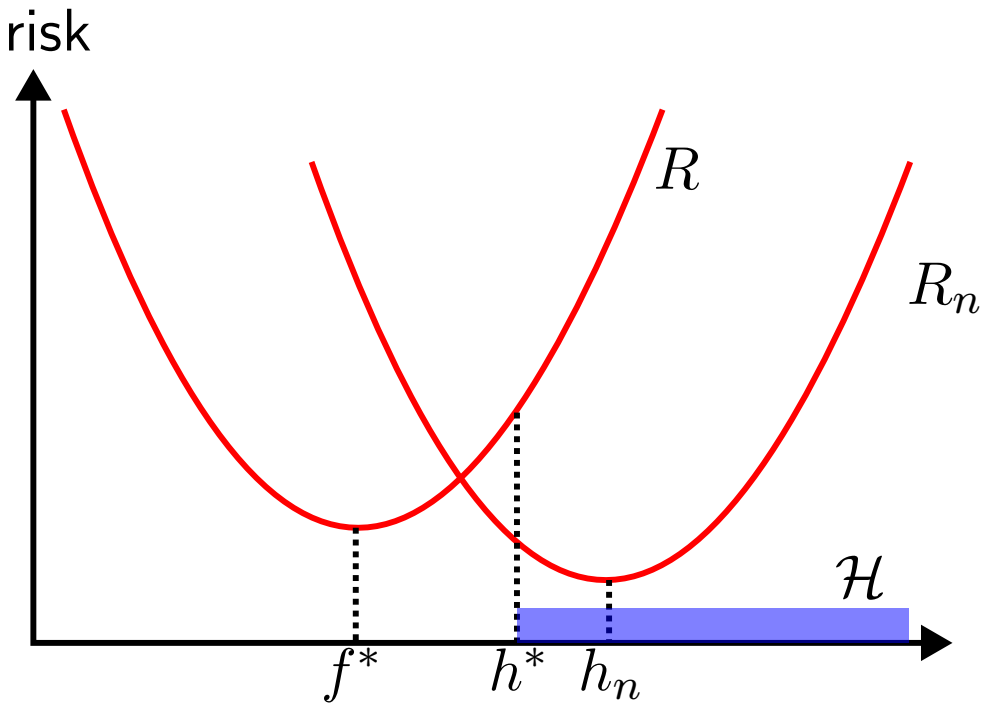
Part 3

Error decompositions

Error decompositions

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

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



Three error decompositions

Approximation and estimation error:

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

Empirical risk and generalization error:

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

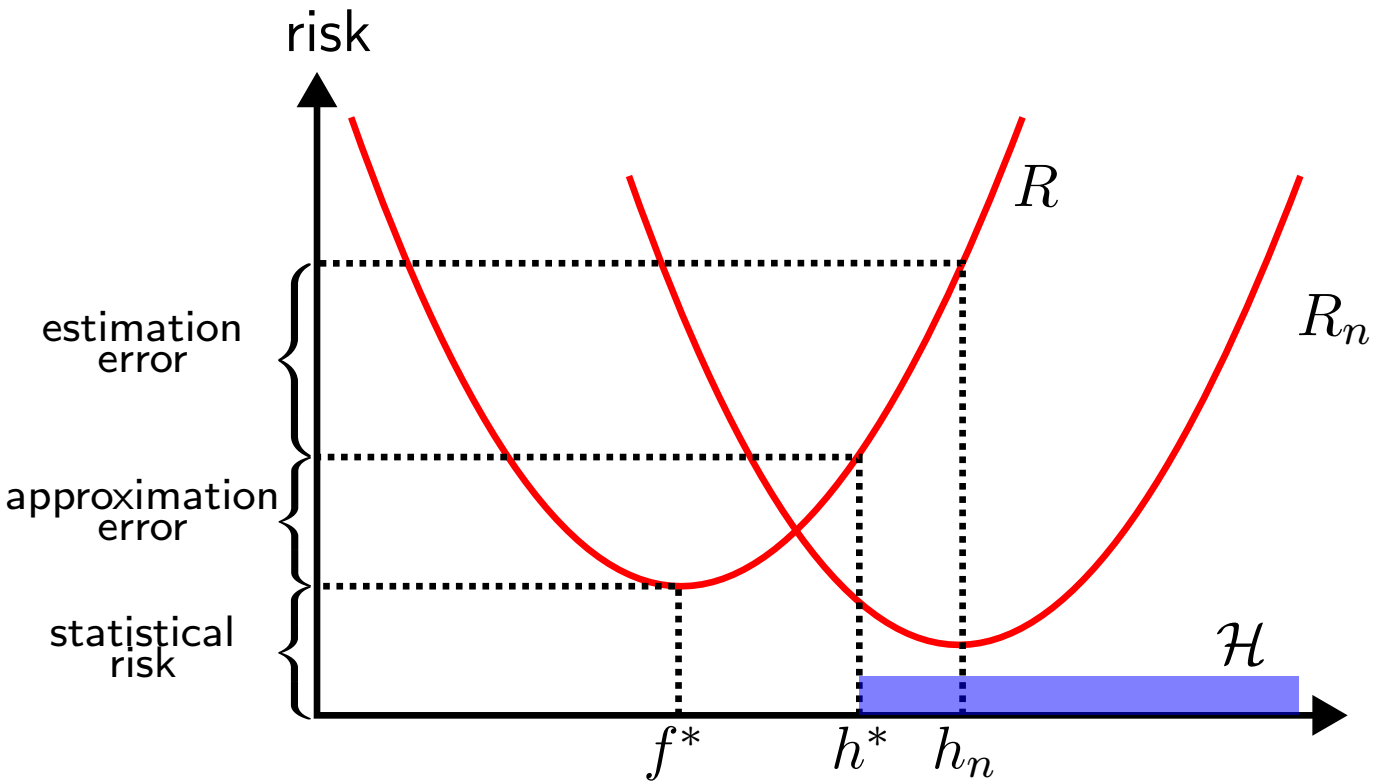
Bias and variance: for $\mathcal{Y} = \mathbb{R}$ and $\ell(y_1, y_2) = (y_1 - y_2)^2$,

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

Approximation and estimation error

Approximation and estimation error:

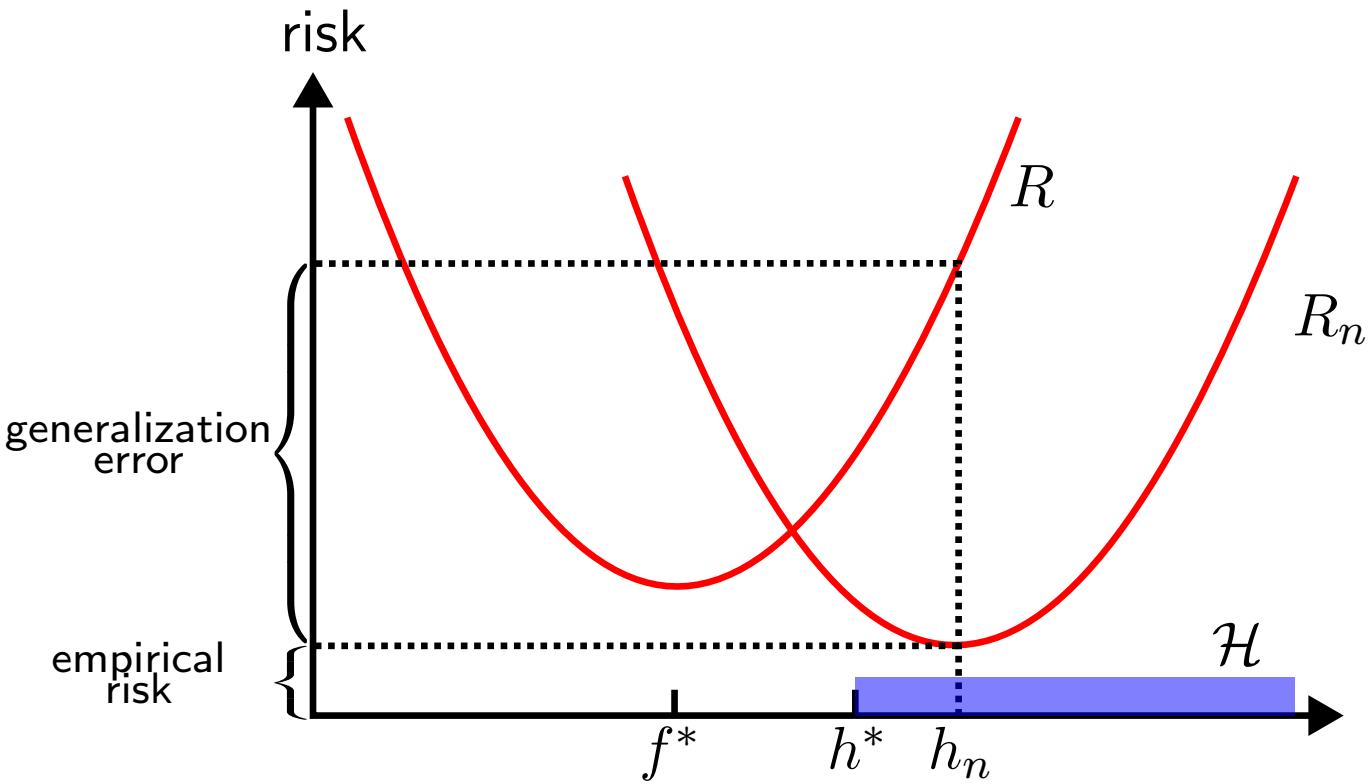
$$R(h_n) = \underbrace{R(f^*)}_{\text{statistical risk}} + \underbrace{\big(R(h^*) - R(f^*)\big)}_{\text{approximation error}} + \underbrace{\big(R(h_n) - R(h^*)\big)}_{\text{estimation error}}$$



Empirical risk and generalization error

Empirical risk and generalization error:

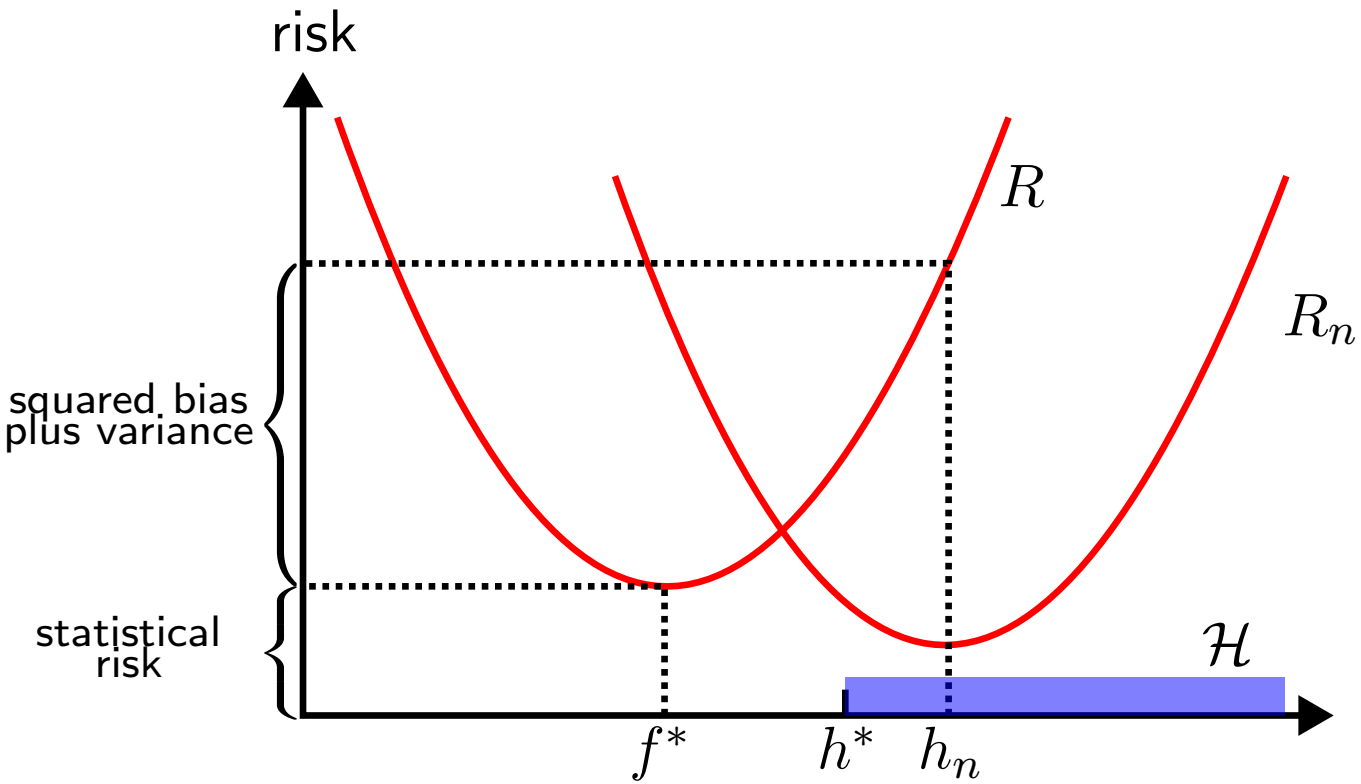
$$R(h_n) = \underbrace{R_n(h_n)}_{\text{empirical risk}} + \underbrace{\big(R(h_n) - R_n(h_n)\big)}_{\text{generalization error}}$$



Bias and variance

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

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



Proof of the bias-variance decomposition

Recall:

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

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

Conditional risk of h_n given (x, ω):

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

Expected risk of h_n: applying $\mathbb{E}$ and E on both sides yields

$$\begin{aligned} \mathbb{E}[R(h_n)] &= R(f^*) + E\big[\mathbb{E}[(h_n(x) - f^*(x))^2]\big] \\ &= R(f^*) + E\big[\underbrace{\mathbb{E}[h_n(x) - f^*(x)]^2}_{\text{squared bias}} + \underbrace{\mathbb{V}\mathrm{ar}[h_n(x)]}_{\text{variance}}\big]. \end{aligned}$$

□

Questions to answer for yourself / discuss with friends

- Repetition: Define the approximation, estimation, and generalization errors.
- Discussion: We haven't talked about the optimization error yet. Intuitively, that's the error which is due to imperfections of the numerical optimizer. How would you define it?

MH3520 Chapter 5

Part 4

Error trade-offs (optional)

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?

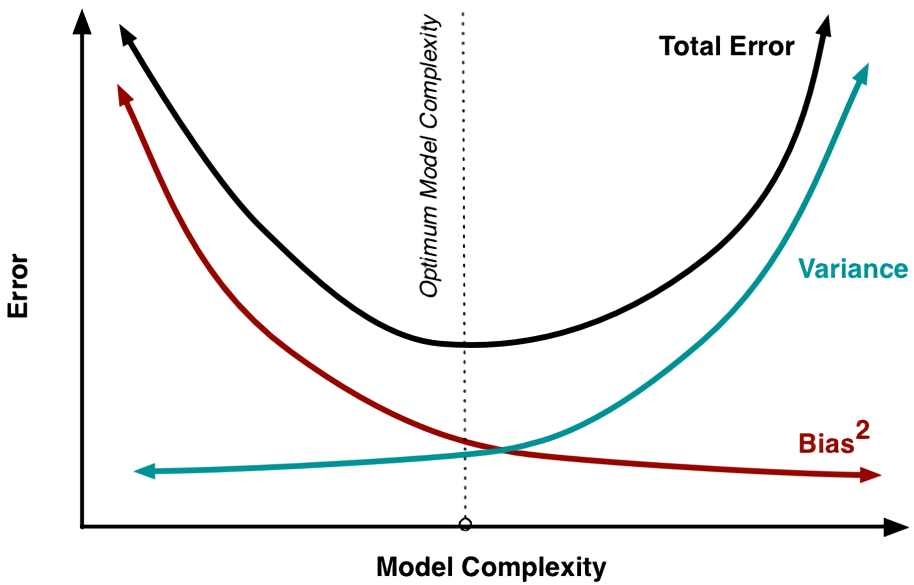
- In general, there is no trade-off in the above error decompositions with respect to $\mathcal{H}$.
- However, there may be trade-offs with respect to $\mathcal{H}$ in some upper bounds for the error (as opposed to the error itself).

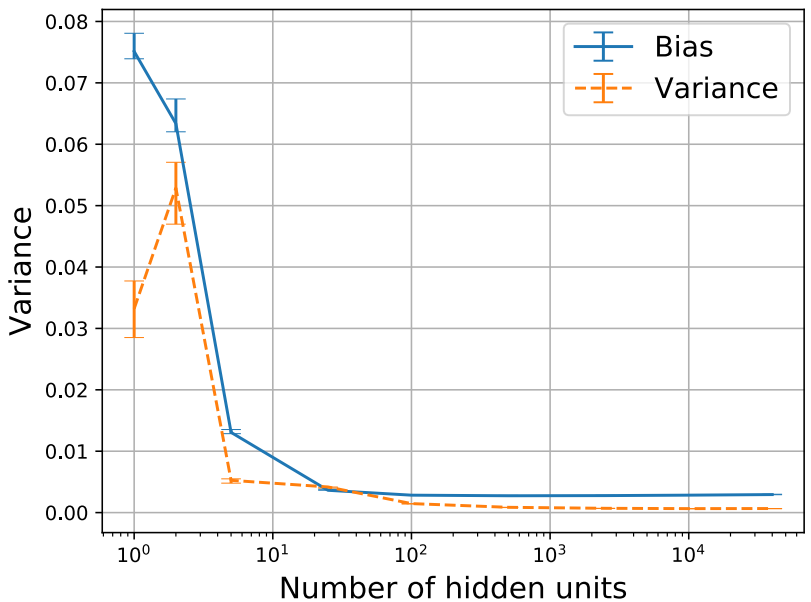
Example: bias-variance decomposition

- Conventional wisdom: The price to pay for achieving low bias is high variance—a trade-off in the choice of $\mathcal{H}$. [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). Lack of any trade-off in MNIST character recognition using sufficiently wide ReLu networks (right).

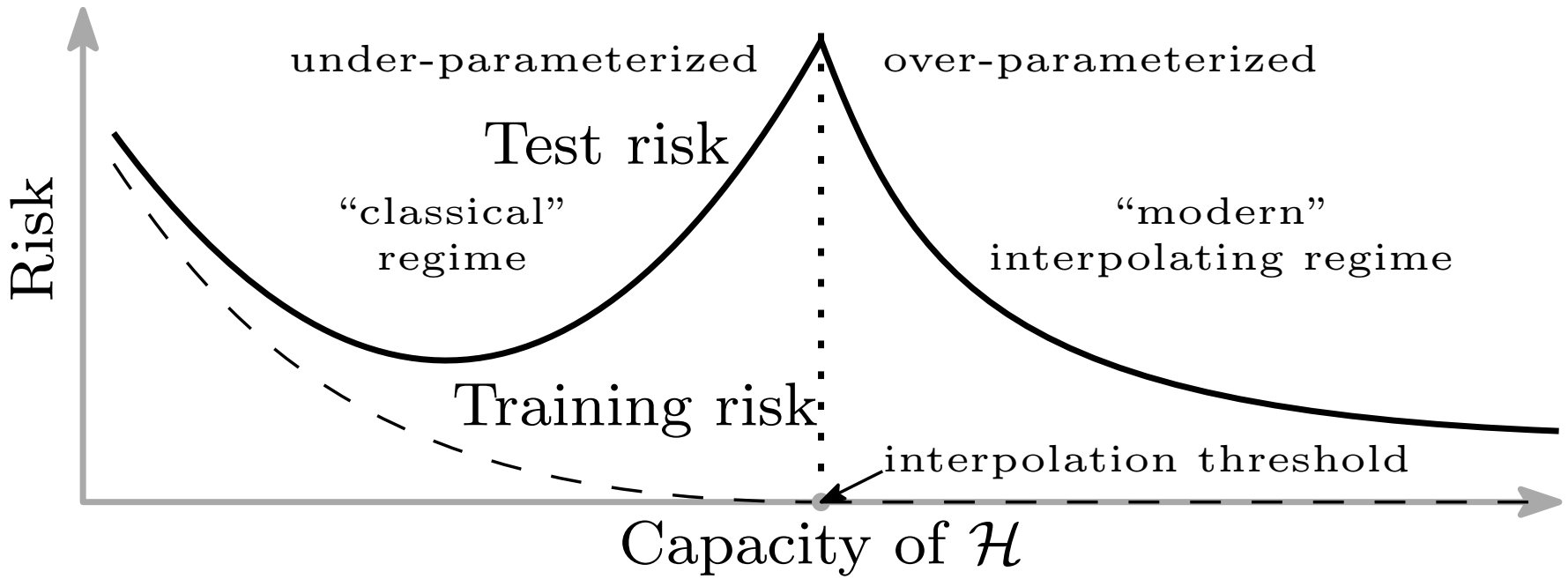




[Figures from Neal 2019]

Example: bias-variance decomposition (cont.)

Conjectured reconciliation: U-shaped risk curve in the underparameterized regime and decreasing risk in the overparameterized regime



[Figure from Belkin e.a. 2019]

Questions to answer for yourself / discuss with friends

- Repetition: What's the difference between an error decomposition and an error trade-off?
- Check your understanding: What is model complexity versus number of hidden units?
- Check your understanding: What is being referred to by the under-parameterized, interpolating, and over-parameterized regimes?
- Discussion: Can you think of a reason (or an example) why the variance might be decreasing in over-parameterized regimes?

MH3520 Chapter 5

Part 5
Error bounds

Bounding the approximation error

Notation:

- f^* solves $R(f^*) = \inf_{f:\, \mathcal{X} \to \mathcal{Y}} R(f)$, and
- h^* solves $R(h^*) = \inf_{h \in \mathcal{H}} R(h)$.
- h_n is a random element in $\mathcal{H}$.

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

- Decreases when $\mathcal{H}$ increases.
- Depends on how closely f^* can be approximated by functions in $\mathcal{H}$.
- Is the main focus of function approximation theory.

Bound for quadratic loss functions: using $E[y \mid x] = f^*(x)$,

$$\begin{aligned} 0 \le R(h^*) - R(f^*) &= E\big[(h^*(x) - f^*(x) + f^*(x) - y)^2 - (f^*(x) - y)^2\big] \\ &= E\big[(h^*(x) - f^*(x))^2 + (h^*(x) - f^*(x))(f^*(x) - y)\big] \\ &= E\big[(h^*(x) - f^*(x))^2\big] = \|h^* - f^*\|_2^2. \end{aligned}$$

Bounding the generalization error

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

- If $h_n \equiv h$ was deterministic and fixed, this would be the difference between an empirical mean and a mean.
- By the central limit theorem, it would converge to zero at rate $n^{-1/2}$.
- However, as h_n is random and highly correlated with the data, one has to resort to uniform estimates.

Uniform generalization error: $\sup_{h\in\mathcal{H}} |R_n(h) - R(h)|$

- Increases when $\mathcal{H}$ increases.
- Is the main focus of statistical learning theory.
- Is the norm of $R_n - R$ in the Banach space $B(\mathcal{H}, \mathbb{R})$ of bounded functions with the supremum norm:

$$\sup_{h\in\mathcal{H}} |R_n(h) - R(h)| = \|R_n - R\|_{B(\mathcal{H},\mathbb{R})}.$$

Bounding the estimation error

Estimation error: $R(h_n) - R(h^*)$

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

$$\ldots \leq R(h_n) - R_n(h_n) + \underbrace{R_n(h_n) - R_n(h^*)}_{\leq 0} + R_n(h^*) - R(h^*)$$

$$\leq 2 \underbrace{\sup_{h \in \mathcal{H}} |R_n(h) - R(h)|}_{\text{uniform generalization error}} .$$

- That's yet another reason why uniform estimates on the generalization error are needed.

Summary: what bounds do we need?

The statistical risk is always present if there is noise in the labels; there is nothing one can do about it. Moreover, the estimation error is bounded in terms of the uniform generalization error. We summarize our findings:

Theorem

If $h_n \in \mathcal{H}$ minimizes the empirical risk R_n, then

$$R(h_n) \leq \underbrace{R(f^*)}_{\text{statistical risk}} + \underbrace{R(h^*) - R(f^*)}_{\text{approximation error}} + 2 \underbrace{\sup_{h \in \mathcal{H}} |R_n(h) - R(h)|}_{\text{uniform generalization error}} .$$

Our next goal is to construct bounds for the uniform generalization error. If there was a central limit theorem in the Banach space $B(\mathcal{H}, \mathbb{R})$, we would be done, but it's not so simple...

Questions to answer for yourself / discuss with friends

- Repetition: What terms do you need to control to get error bounds for empirical risk minimization?
- Check your understanding: What's an example of a norm which is stronger than the L^2 norm?
- Transfer: Describe maximum likelihood estimation as an empirical risk minimization problem. What is the main challenge there: bounding the approximation or generalization error?

MH3520 Chapter 5

Part 6
Concentration inequalities

Empirical mean

Standing assumption. $(\Omega, \mathcal{F}, \mathbb{P})$ is a probability space carrying iid. random variables $X, X_1, X_2, \cdots \in L_2(\Omega, B)$ with mean μ taking values in a separable Banach space B.

Definition

The empirical mean of $X_1, \ldots, X_n$ is

$$\overline{X}_n := \frac{1}{n} \sum_{i=1}^{n} X_i.$$

Remark. Next, we will review some classical results on convergence of the empirical mean to the mean. We'll see them work nicely if B is Hilbert and break down if B is merely Banach.

Law of large numbers

Here is a quantitative law of large numbers, meaning that it asserts not merely convergence but also gives a convergence rate of $1/2$:

Theorem

If B is a Hilbert space, then the empirical mean converges in $L_2(\Omega, B)$ to the mean:

$$\|\overline{X}_n - \mu\|_{L_2(\Omega,B)} \leq \frac{1}{\sqrt{n}} \|X - \mu\|_{L_2(\Omega,B)} = O(n^{-1/2}).$$

Proof: see next slide.

Remark. As an aside, the strong law of large numbers would assert almost sure convergence $\overline{X}_n \to \mu$.

Law of large numbers (cont.)

Proof. Without loss of generality $\mu = 0$.

- By independence, $\mathbb{E}[\langle X_i, X_j \rangle] = \langle \mathbb{E}[X_i], \mathbb{E}[X_j] \rangle = 0$. (This can be seen via approximation by elementary random variables.)
- By Pythagoras, and by the identical distribution of all X_i's,

$$\mathbb{E}\big[\|\overline{X}_n\|^2\big] = \frac{1}{n^2} \sum_{i,j=1}^{n} \mathbb{E}\big[\langle X_i, X_j \rangle\big] = \frac{1}{n^2} \sum_{i=1}^{n} \mathbb{E}\big[\|X_i\|^2\big] = \frac{1}{n} \mathbb{E}\big[\|X\|^2\big].$$

- Taking the square root on both sides yields the desired estimate. □

Convergence in distribution

For a more fine-grained analysis of the convergence $\overline{X}_n \to \mu$, we need the notion of convergence in distribution:

Definition

A sequence of random variables $Y_n \in L_0(\Omega, B)$ converges in distribution to a random variable $Y \in L_0(\Omega, B)$ if for all continuous bounded functions $f : B \to \mathbb{R}$,

$$\mathbb{E}[f(Y_n)] \xrightarrow[n\to\infty]{} \mathbb{E}[f(Y)].$$

Remark. Convergence in distribution is the weakest of all notions of stochastic convergence; it is implied by convergence in $L_0(\Omega, B)$.

Central limit theorem

Here is the promised refinement of the law of large numbers, obtained by rescaling the empirical mean:

Theorem

Let B be a Hilbert space. Then, the rescaled empirical means

$$Y_n := \sqrt{n}(\overline{X}_n - \mu)$$

converge in distribution to a normal distribution.

Proof: skipped (uses the Fourier transform).

Remark. There are two things to learn from this: first, the rate of $1/2$ is sharp, and second, the rescaled empirical means are asymptotically normal.

Beyond Hilbert spaces?

Without any further assumptions, the central limit theorem and the associated convergence of order $1/2$ fail on Banach spaces:

Example

Let $B = B(\mathcal{H}, \mathbb{R})$ be the Banach space of real-valued bounded functions on a set $\mathcal{H}$, endowed with the supremum norm. Assume that $X_n(h)$, $n \in \mathbb{N}$, $h \in \mathcal{H}$, are iid. with values ± 1 with probability $1/2$. Then $X_1, X_2, \ldots$ are bounded iid. random variables with values in B and mean $\mu = 0$. If $\mathcal{H}$ is an infinite set, then $\|\overline{X}_n - \mu\|_{B(\mathcal{H},\mathbb{R})} = 1$ almost surely.

Remark. This means no convergence in any L_p, not even in L_0.

To the rescue: concentration inequalities

We'll have to lower our ambitions.

Additional assumption:

- $X(h)$ independent of $X(k)$ for $h \neq k$ was the worst possible scenario.
- To save our skin, we will assume $X(h)$ to be Lipschitz in h.
- Then, an error bound for h is also an error bound for all nearby k's.
- For this reason, it will suffice to consider finitely many h, meaning that we need bounds for $B = \mathbb{R}$ only.

Weaker conclusion:

- We will gain control over $\overline{X}_n - \mu$ in L_0 but not in L_p, $p > 0$.
- The bounds take the form of concentration inequalities, which say that the error has most mass near zero, and only very little mass in the tails.

Tails of normal distributions

To get some intuition, let us try to bound the tails of a normal distribution.

Lemma

If X is standard normally distributed, then its tails are bounded as follows:

$$\forall \epsilon > 0 : \qquad \mathbb{P}\big[|X| \geq \epsilon\big] \leq 2e^{-\frac{\epsilon^2}{2}}.$$

Proof. This follows from the one-sided bound

$$\mathbb{P}\big[X \geq \epsilon\big] \leq \inf_{t \in \mathbb{R}+} \mathbb{E}\big[e^{t(X-\epsilon)}\big] = \inf_{t \in \mathbb{R}+} e^{\frac{t^2}{2} - \epsilon t} = e^{-\frac{\epsilon^2}{2}}. \qquad \square$$

Bernstein inequality

- By the previous lemma, if $\overline{X}_n - \mu$ was $N(0, \sigma^2/n)$ distributed, then

$$\mathbb{P}\left[|\overline{X}_n - \mathbb{E}[X]| \geq \epsilon\right] \leq 2e^{-\frac{n\epsilon^2}{2\sigma^2}}.$$

- In reality, $\overline{X}_n - \mu$ is only approximately $N(0, \sigma^2/n)$ distributed, but we get a similar bound:

Theorem

Let $B = \mathbb{R}$, and suppose that $|X - \mu| \leq M$ and $\mathbb{E}[|X - \mu|^2] \leq \sigma^2$. Then, Bernstein's inequality holds for all $\epsilon > 0$:

$$\mathbb{P}\left[|\overline{X}_n - \mu| \geq \epsilon\right] \leq 2e^{-\frac{n\epsilon^2}{2(\sigma^2 + \frac{1}{3}M\epsilon)}}$$

Proof: see next slide.

Bernstein inequality (cont.)

Proof. Without loss of generality, $\mu = 0$.

- Similarly to the previous lemma, we have for all $t > 0$ that

$$\mathbb{P}\big[\overline{X}_n \geq \epsilon\big] = \mathbb{P}\Big[\sum_{i=1}^{n} X_i \geq n\epsilon\Big] \leq \mathbb{E}\big[e^{t(\sum_{i=1}^{n} X_i - n\epsilon)}\big] = \Big(e^{-\epsilon t}\mathbb{E}\big[e^{tX}\big]\Big)^n.$$

- We bound the moment generating function of X:

$$\mathbb{E}\big[e^{tX}\big] = 1 + \mathbb{E}[tX] + \sum_{k\geq 2} \frac{t^k}{k!}\mathbb{E}[X^2 X^{k-2}] \leq 1 + \sum_{k\geq 2} \frac{t^k}{k!}\sigma^2 M^{k-2}$$

$$= 1 + \underbrace{\frac{\sigma^2}{M^2}\big(e^{tM} - 1 - tM\big)}_{=:g(t)} \leq e^{g(t)}.$$

- Minimization by setting the derivative to zero yields

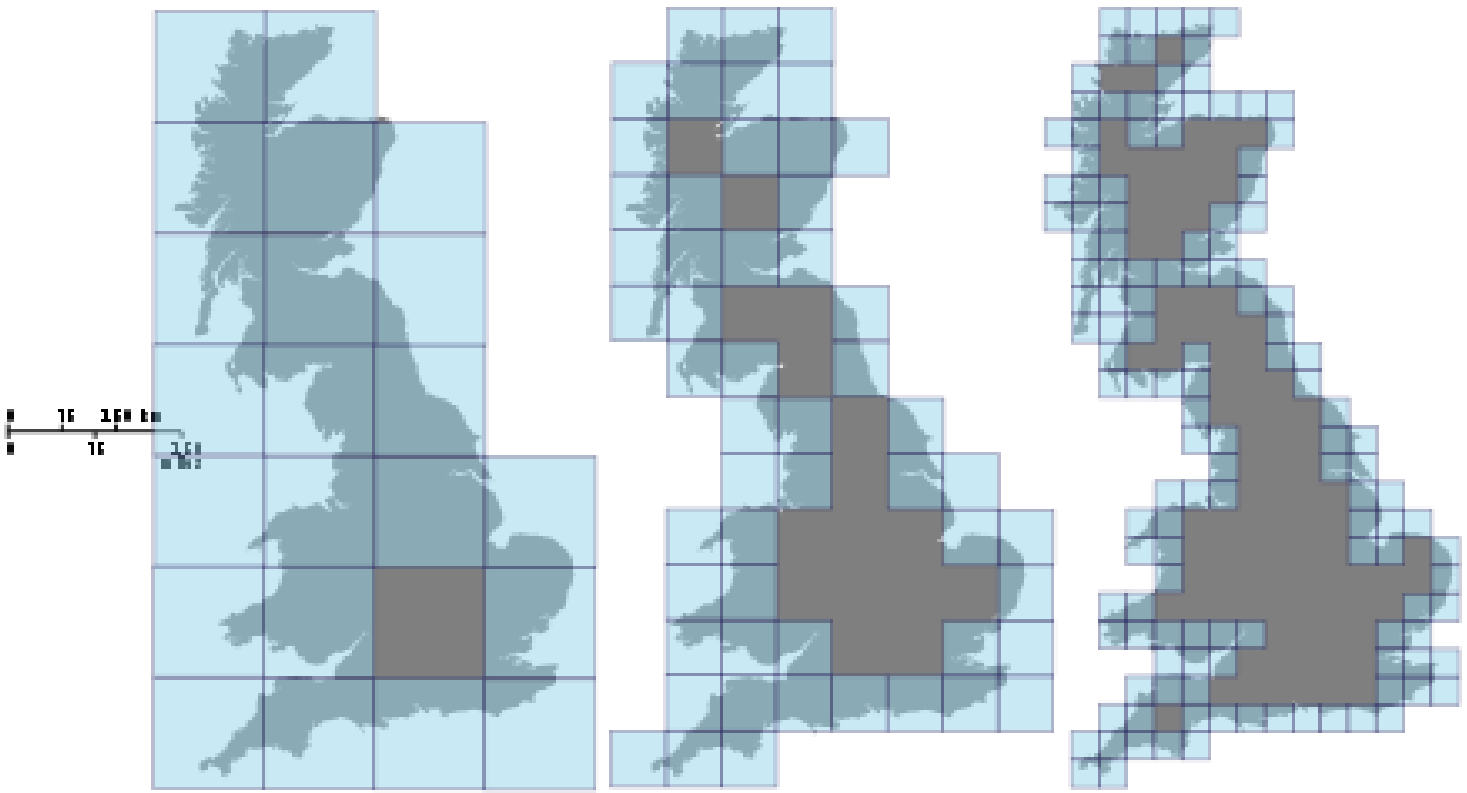
$$\mathbb{P}\big[\overline{X}_n \geq \epsilon\big] \leq \inf_{t>0} e^{-n\epsilon t + n g(t)} = e^{-\frac{n\epsilon^2}{2\sigma^2}\varphi(M\epsilon/\sigma^2)},$$

where $\varphi(\lambda) = 2\lambda^{-}2[(1+\lambda)\log(1+\lambda) - \lambda] \geq (1 + \frac{1}{3}\lambda)^{-1}$. □

Covering numbers

Definition

The covering number $\mathcal{N}(\mathcal{H}, \epsilon)$ of a metric space $\mathcal{H}$ is the minimal number $N \in \mathbb{N}$ such that N discs of radius ϵ cover all of $\mathcal{H}$.



[P. Grohs]

Covering numbers and compactness

Theorem

A metric space is totally bounded if all of its covering numbers are finite, i.e., if for every $\epsilon > 0$, it can be covered by finitely many ϵ-balls.

Theorem

A metric space is compact if and only if it is complete and totally bounded.

Remark. In this sense, covering numbers can be seen as a quantification of compactness.

Covering numbers

Let's consider covering numbers of subsets $\mathcal{H}$ of a Banach space B.

Counter-example (non-compact hypothesis classes)

$\mathcal{N}(\mathcal{H}, \epsilon) = \infty$ for all small ϵ unless $\mathcal{H}$ has compact closure in B.

Example (balls in finite dimensions)

Let $\mathcal{H}$ be a ball of radius r in a d-dimensional linear subspace of B. Then, $\mathcal{N}(\mathcal{H}, \epsilon) \leq (4r/\epsilon)^d$.

Example (balls in function spaces)

Let $B = C_b(\mathcal{X}, \mathbb{R})$ be the Banach space of bounded continuous functions on a bounded domain $\mathcal{X} \subseteq \mathbb{R}^d$ with smooth boundary. Let $\mathcal{H}$ be a ball of radius r in the Sobolev space $H^s(\mathcal{X}, \mathbb{R})$ or the Hölder space $C^s(\mathcal{X}, \mathbb{R})$. Then, $\log \mathcal{N}(\mathcal{H}, \epsilon) \leq (cr/\epsilon)^{d/s}$, for some $c > 0$.

Uniform Bernstein inequality

We'll get tail bounds for finitely many h's using Bernstein's inequality and derive tail bounds for all nearby k's using a Lipschitz assumption:

Theorem

Let $B = B(\mathcal{H}, \mathbb{R})$ for some metric space $\mathcal{H}$, and let L, M, σ be real numbers such that for all $h, k \in \mathcal{H}$,

$$|X(h) - \mu(h)| \leq M, \qquad \text{(bounded)}$$
$$|X(h) - X(k)| \leq L\ d(h,k) \qquad \text{(Lipschitz)}$$
$$\mathbb{E}[|X(h) - \mu(h)|^2] \leq \sigma^2. \qquad (2^{\text{nd}} \text{ moment})$$

Then, the uniform Bernstein inequality holds, i.e., for all $\epsilon > 0$:

$$\mathbb{P}\left[\sup_{h \in \mathcal{H}} |\overline{X}_n(h) - \mu(h)| \geq \epsilon\right] \leq \mathcal{N}(\mathcal{H}, \tfrac{\epsilon}{4L}) 2e^{-\frac{n\epsilon^2}{4(2\sigma^2 + \frac{1}{3}M\epsilon)}}$$

Proof of the uniform Bernstein inequality

Proof.

- Cover $\mathcal{H}$ by $\mathcal{N}(\mathcal{H}, \frac{\epsilon}{2L})$ balls B_j of radius $\frac{\epsilon}{4L}$ centered at h_j.
- By the Lipschitz assumption, for any $h \in B_j$,

$$|\overline{X}_n(h) - \mu(h) - \overline{X}_n(h_j) + \mu(h_j)| \leq \epsilon/2.$$

- Thus, by Bernstein's inequality with $\epsilon/2$ in place of ϵ,

$$\begin{aligned}
&\mathbb{P}\left[\sup_{h\in\mathcal{H}} |\overline{X}_n(h) - \mathbb{E}[X(h)]| \geq \epsilon\right] \\
&\quad\leq \sum_j \mathbb{P}\left[\sup_{h\in B_j} |\overline{X}_n(h) - \mathbb{E}[X(h)]| \geq \epsilon\right] \\
&\quad\leq \sum_j \mathbb{P}\left[|\overline{X}_n(h_j) - \mathbb{E}[X(h_j)]| \geq \frac{\epsilon}{2}\right] \\
&\quad\leq \mathcal{N}(\mathcal{H}, \tfrac{\epsilon}{4L}) 2e^{-\frac{n\epsilon^2}{4(2\sigma^2 + \frac{1}{3}M\epsilon)}}
\end{aligned}$$

□

Implications and outlook

- Convergence in L_0: If n is sufficiently large such that the RHS of the uniform Bernstein inequality is at most ϵ, then

$$\| \overline{X}_n - \mu \|_{L_0(\Omega, B(\mathcal{H}, \mathbb{R}))} \leq \epsilon,$$

  i.e., X_n converges to μ in probability. The convergence rate depends on the small-ϵ behavior of $\mathcal{N}(\mathcal{H}, \epsilon)$.

- Outlook: There are some important refinements (e.g. better rates for convex $\mathcal{H}$). Covering numbers have been worked out for many hypothesis classes (Euclidean spaces and many function spaces). We next apply the results to statistical learning.

Questions to answer for yourself / discuss with friends

- Repetition: How fast does the empirical mean of iid. random variables converge to the mean?
- Repetition: What is the covering number of a metric space?
- Check your understanding: Does the central limit theorem hold on $B(\mathcal{H}, \mathbb{R})$ if $\mathcal{H}$ is a finite set? What is the covering number of a finite set?
- Check your understanding: How does the RHS of the uniform Bernstein inequality behave for fixed n and small ϵ? And what about fixed ϵ and large n?

MH3520 Chapter 5

Part 7

Bounds on the uniform generalization error

Assumptions

To be as specific as possible, we work with a quadratic loss. We assume that our hypothesis class consists of bounded functions, and we endow it with the supremum norm. Most of this can be generalized.

Standing assumption:

- Input and output spaces: topological space $\mathcal{X}$, real numbers $\mathcal{Y} := \mathbb{R}$.
- Loss function: $\ell(y, y') = (y - y')^2$.
- Hypothesis class (aka. model class): a set $\mathcal{H}$ of bounded measurable functions $h\colon \mathcal{X} \to \mathcal{Y}$, endowed with the supremum norm.
- Observations: independent random variables $(X, Y), (X_1, Y_1), \ldots, (X_n, Y_n)$, defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, distributed according to a probability measure P on $\mathcal{X} \times \mathcal{Y}$.

Uniform generalization error

- Recall that the empirical and expected risk of $h \in \mathcal{H}$ are defined as

$$R_n(h) = \sum_{i=1}^{n} \ell(h(X_i), Y_i), \qquad R(h) = \mathbb{E}[\ell(h(X), Y)].$$

- By the central limit theorem or the quantitative law of large numbers, the generalization error with fixed $h \in \mathcal{H}$ is of order $n^{-1/2}$:

$$\big\|R_n(h) - R(h)\big\|_{L_2(\Omega,\mathbb{R})} = O(n^{-1/2}).$$

- The uniform generalization error is defined as

$$\sup_{h\in\mathcal{H}} |R_n(h) - R(h)| = \big\|R_n - R\|_{B(\mathcal{H},\mathbb{R})},$$

where $B(\mathcal{H}, \mathbb{R})$ is the Banach space of bounded functions $\mathcal{H} \to \mathbb{R}$ with the supremum norm.

Bounding the uniform generalization error

Here is the foundational result of statistical learning theory:

Theorem

Assume that $|h(x) - y| \leq M$ *holds* P*-almost surely, and let* $\sigma^2 := \sup_{h \in \mathcal{H}} \mathrm{Var}[(h(x) - y)^2]$. *Then, for all* $\epsilon > 0$,

$$\mathbb{P}\left[\sup_{h \in \mathcal{H}} |R_n(h) - R(h)| \geq \epsilon\right] \leq \mathcal{N}(\mathcal{H}, \tfrac{\epsilon}{8M}) 2e^{-\frac{n\epsilon^2}{4(2\sigma^2 + \frac{1}{3}M^2\epsilon)}}.$$

Proof. Apply the uniform Bernstein inequality with...

- $X_n(h)$ replaced by $\ell(h(X_n), Y_n))$,
- M replaced by M^2 because $\ell(h(x), y) \leq M^2$, and
- L replaced by $4M$ because

$$\big(h(x)-y\big)^2 - \big(k(x)-y\big)^2 = \underbrace{\big(h(x) - k(x)\big)}_{|\ldots| \leq \|h-k\|_\infty} \underbrace{\big(h(x) + k(x) - 2y\big)}_{|\ldots| \leq 4M}. \qquad \square$$

Bounding the estimation error

Recall that the estimation error $R(h_n) - R(h^*)$ is non-negative and bounded by twice the uniform generalization error. However, this bound can be sharpened.

Theorem

Assume that h_n minimizes the empirical risk R_n over a convex hypothesis set $\mathcal{H}$, let $|h(x) - y| \leq M$ hold P-almost surely, and let $\sigma^2 := \sup_{h \in \mathcal{H}} \mathrm{Var}[(h(x) - y)^2]$. Then, for all $\epsilon > 0$,

$$\mathbb{P}\left[R(h_n) - R(h^*) \geq \epsilon\right] \leq \mathcal{N}(\mathcal{H}, \tfrac{\epsilon}{24M}) e^{-\frac{n\epsilon}{288M^2}}.$$

Remark.

- You get this new bound, with ϵ^2 replaced by ϵ, by setting $\sigma^2 = 0$.
- Intuitively, σ^2 quantifies the statistical risk, and the statistical risk cancels out in the difference $R(h_n) - R(h^*)$.

From L_0 to L_p bounds

Lemma

For any $X \in L_p(\Omega, B)$,

$$\mathbb{E}[\|X\|^p] = \int_0^\infty p\epsilon^{p-1}\mathbb{P}[\|X\| \geq \epsilon]\, d\epsilon.$$

Proof. If $\|X\|$ has density f and distribution F, integration by parts yields

$$\mathbb{E}[\|X\|^p] = \int_0^\infty \epsilon^p f(\epsilon)\, d\epsilon = \underbrace{\epsilon^p(1-F(\epsilon))\big|_0^\infty}_{=0} + \int_0^\infty p\epsilon^{p-1}(1-F(\epsilon))\, d\epsilon.$$

□

Remark. This applies well to the bounds in the previous two theorems, which involve Gaussian or exponential densities.

From L_0 to L_p bounds

Recall that $\mathcal{N}(\mathcal{H}, \epsilon)$ is bounded if $\mathcal{H}$ is a finite set, $\approx \epsilon^{-d}$ if $\mathcal{H}$ is a ball in a d-dimensional normed space, and $\approx e^{c\epsilon^{-d/s}}$ if $\mathcal{H}$ is a ball in a function space with regularity s on a bounded d-dimensional domain.

Corollary (Uniform generalization error)

$$\|R_n - R\|_{L_p(\Omega, B(\mathcal{H},\mathbb{R}))} \lesssim \begin{cases} n^{-1/2}, & \text{if } \mathcal{N}(\mathcal{H}, \epsilon) \text{ is bounded} \\ n^{(-1+d/p)/2}, & \text{if } \mathcal{N}(\mathcal{H}, \epsilon) \lesssim \epsilon^{-d}, \\ \infty, & \text{if } \mathcal{N}(\mathcal{H}, \epsilon) \approx e^{c\epsilon^{-d/s}} \end{cases}$$

Corollary (Estimation error)

$$\|R(h_n) - R(h^*)\|_{L_p(\Omega,\mathbb{R})} \lesssim \begin{cases} n^{-1}, & \text{if } \mathcal{N}(\mathcal{H}, \epsilon) \text{ is bounded} \\ n^{-1+d/p}, & \text{if } \mathcal{N}(\mathcal{H}, \epsilon) \lesssim \epsilon^{-d}, \\ \infty, & \text{if } \mathcal{N}(\mathcal{H}, \epsilon) \approx e^{c\epsilon^{-d/s}} \end{cases}$$

Putting it all together

If there is no approximation error, i.e., $f^* \in \mathcal{H}$, then...

- We have shown a quantitative version of the statement that $R(h_n)$ converges to $R(f^*)$ in probability, i.e.,

$$\forall \epsilon > 0 : \qquad \lim_{n \to \infty} \mathbb{P}[|R(h_n) - R(f^*)| \geq \epsilon] = 0.$$

- In words, h_n is probably approximately correct (PAC). Similarly, the uniform Bernstein inequality is called a PAC bound.
- Some of these L_0 bounds can be turned into L_p bounds.

If there is an approximation error, i.e., $f^* \notin \mathcal{H}$, then...

- The above statements don't hold, unless we consider increasing hypothesis classes H_n such that the approximation error $\inf_{h \in H_n} R(h) - R(f^*)$ tends to zero as $n \to \infty$.
- How fast will will it tend to zero? This leads into function approximation theory.

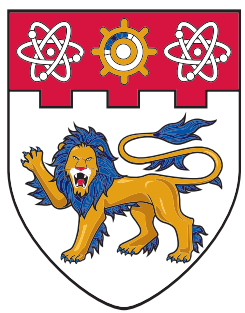
Questions to answer for yourself / discuss with friends

- Repetition: Describe the PAC bound on the uniform generalization error provided by statistical learning theory.
- Check your understanding: We used the supremum norm on $\mathcal{H}$. Could we alternatively use some weaker or stronger norm?
- Discussion: Can you spot any points where the error analysis of statistical learning theory leaves room for improvements?

MH3520:Mathematics of Deep Learning

Chapter 6

Basics of functional analysis



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

Last chapters:

- Numerical optimization $\rightsquigarrow$ training neural networks
- Probability theory $\rightsquigarrow$ error bounds for learning algorithms

This chapter: basics of functional analysis

- Topological vector spaces and linear mappings between them
- Prime example: function spaces and integral operators

Next chapter:

- Perceptrons are universal, i.e., dense in certain function spaces.

What is functional analysis?

. . . the single most successful mathematical theory of the 20th century!

- Algebra studies sets endowed with operations such as addition, multiplication, inverse, etc.
- Topology studies sets where continuous passing from some elements to others makes sense.
- Functional analysis studies sets which have both algebraic and topological structure (for instance, topological vector spaces).

There are applications in many fields of pure and applied mathematics (e.g. differential equations, harmonic analysis, numerical analysis, inverse problems) and in the natural sciences (e.g. quantum mechanics)

Overview of Chapter 6

1. Topology
2. Topological vector spaces
3. A zoo of function spaces
4. Convexity and the Hahn–Banach theorem
5. Completeness and the uniform boundedness principle (Optional)

Sources for this chapter:

- Helemskii (2006): Lectures and exercises on functional analysis. Mathematical Monographs, Volume 233. American Mathematical Society.

MH3520 Chapter 6

Part 1
Topology

Topology

Definition

A topology on a set E is a collection $\mathcal{O}$ of sets, which are called open, s.t.

- The empty set $\emptyset$ and the full set E are open, i.e.,

$$\emptyset \in \mathcal{O}, \qquad \Omega \in \mathcal{O}.$$

- The union of open sets is open, i.e., for any index set I,

$$A_i \in I \text{ for all } i \in I \qquad \Longrightarrow \qquad \bigcup_{i \in I} A_i \in \mathcal{O}.$$

- The finite intersection of open sets is open, i.e., for any $n \in \mathbb{N}$,

$$A_1, \ldots, A_n \in \mathcal{O} \qquad \Longrightarrow \qquad \bigcap_{i=1}^{n} A_i \in \mathcal{O}.$$

The tuple $(E, \mathcal{O})$ is called a topological space.

Examples of topological spaces

Every metric space and normed space is a topological space:

Example

The open sets (defined as unions of open balls) in a normed space or metric space form a topology; it is the smallest topology containing all open balls.

Here is the smallest possible topology:

Example

The trivial topology on E is $\mathcal{O} = \{\emptyset, E\}$.

Here is the largest one, which is commonly used on countable sets:

Example

The discrete topology on E is the power set $\mathcal{O} = 2^E = \{A : A \subseteq E\}$.

Neighborhoods

Definition

A neighborhood of a point x is a set U such that $x \in O \subseteq U$ for some open set O.

Note that a set is open if and only if it is a neighborhood of all of its points. Here is a technical detail:

Definition

A topological space is called Hausdorff if any two distinct points have disjoint neighborhoods.

We will always consider Hausdorff spaces, unless mentioned otherwise.

Example

Every normed or metric space is Hausdorff. However, spaces of measurable functions (without equivalence almost everywhere) with L^p norms (or rather semi-norms!) are non-Hausdorff.

Continuity

Definition

A function $f : E_1 \to E_2$ between topological space $(E_1, \mathcal{O}_1)$ and $(E_2, \mathcal{O}_2)$ is continuous if

$$\forall O_2 \in \mathcal{O}_2 : \qquad f^{-1}(O_2) \in \mathcal{O}_1.$$

Remark. This notion of continuity extends the usual one for metric and normed spaces.

Definition

A sequence $x : \mathbb{N} \to E$ converges to a point x in a topological space $(E, \mathcal{O})$ if

$$\forall O \in \mathcal{O} \text{ such that } x \in O : \quad \exists N \in \mathbb{N} : \quad \forall n \geq N : \qquad x_n \in \mathcal{O}.$$

Remark. More generally, one may replace $\mathbb{N}$ by a directed set I. Then, $x : I \to E$ is called a net. The motivation is that convergence of sequences does not identify the topology, but convergence of nets does.

New topological spaces from old ones

Definition

The product of topological spaces $(E_i)_{i\in I}$ is the set $E := \prod_{i\in I} E_i$ with the smallest topology such that all projections $\mathrm{pr}_i : E \to E_i$ are continuous.

Remark. Finite product of normed spaces are normed, and countable products of metric spaces are metric spaces:

$$\|x\| := \sum_{i=1}^{n} \|x_i\|_i, \qquad d(x,y) := \sum_{i=1}^{\infty} 2^{-i} \frac{d_i(x_i,y_i)}{1+d_i(x_i,y_i)}.$$

Definition

The topology on a subset F of a topological space is defined as the smallest topology such that the inclusion $i : F \to E$ is continuous.

Remark. Subspaces of normed spaces are normed, and subsets of metric spaces are metric spaces.

Interior, closure, and boundary

Definition

A set is closed if its complement is open.

Definition

On any topological space, . . .

- The interior A° of a set A is its largest open subset.
- The closure $\overline{A}$ of a set A is its smallest closed superset.
- The boundary ∂A of a set A is its closure minus its interior.

Remark. In metric or normed spaces, the closure of a set A is the set of all limits of sequences in A.

Definition

A dense subset of a topological space E is a set whose closure is all of E.

Compactness

A cover of a set is a family of subsets whose union is all of the set. A cover is open if it consists of open sets and finite if it consists of finitely many sets. A sub-cover is a sub-family which is a cover.

Definition

A set is compact if every open cover has a finite sub-cover.

Theorem

- *A metric space is compact if and only if it is complete and totally bounded,*[3] *if and only if every sequence has a converging subsequence.*
- *A subset of a compact space is compact if and only if it is closed.*
- *The continuous image of a compact set is compact.*
- *The product of compact sets is compact.*

[3] Total boundedness means that all covering numbers are finite.

Questions to answer for yourself / discuss with friends

- Repetition: What is the definition of a topology, and what does it have to do with continuity and convergence?
- Check your understanding: True or false: a converging sequence is a continuous function $\mathbb{N} \cup \{\infty\} \to E$.
- Check your understanding: Complete the sentence: A set A in a metric space E is dense if for every point $x \in E$, ...
- Discussion: Let's try to prove this together: the continuous image of a compact set is compact.

MH3520 Chapter 6

Part 2

Topological vector spaces

Topological vector spaces

The object of interest in functional analysis are topological vector spaces and continuous linear functions (aka. maps) between them:

Definition

A topological vector space is a vector space E endowed with a topology such that addition $E \times E \to E$ and scalar multiplication $\mathbb{R} \times E \to E$ are continuous.

Example

Every normed vector space is a topological vector space: continuity of addition follows from the triangle inequality, and continuity of scalar multiplication follows from the absolute homogeneity of the norm.

Example

If the norm is replaced by a metric, continuity of addition and scalar multiplication are not automatic; one speaks of metric vector spaces.

Continuity and boundedness

Lemma

A linear map between topological vector spaces is continuous if and only if it is continuous at zero (or any other point).

Proof. Let $A : E \to F$ be linear.

$$\begin{aligned} A \text{ is continuous at } x &\iff A(\cdot - x) \text{ is continuous at } 0 \\ &\iff A(\cdot) - A(x) \text{ is continuous at } 0 \\ &\iff A(\cdot) \text{ is continuous at } 0. \end{aligned}$$

□

Lemma

A linear map between normed vector spaces is continuous if and only if it is bounded, i.e., the image of the unit ball is a bounded set.

Dual spaces

Definition

The (topological) dual space E^* of a topological vector space E is the set of continuous linear functionals $E \to \mathbb{R}$.

Example

The dual of a normed space is a Banach space when endowed with the operator norm.

Example

Hilbert spaces are self-dual, i.e., a Hilbert space E is isomorphic to its dual E^* via the Riesz isomorphism.

Topologies on dual spaces

We have already encountered the strong topology:

Definition

The strong topology on the dual E^* of a normed space E is given by the operator norm $\|\alpha\| := \sup\{|\alpha(x)| : x \in E, \|x\| \leq 1\}$. Then, E^* is a Banach space with this norm.

There is also another natural topology on a dual:

Definition

The weak* topology on the dual E^* of a topological vector space E is the product topology on

$$E^* = \prod_{x \in E} \mathbb{R}, \qquad \alpha \in E^* \leftrightsquigarrow (\alpha(x))_{x \in E}.$$

Convergence $\alpha_n \to \alpha$ in the weak* topology on E^* is equivalent to point-wise convergence, i.e., $\alpha_n(x) \to \alpha(x)$, for all $x \in E$.

Topologies on pre-dual spaces

Every $\alpha \in E^*$ is by definition continuous $E \to \mathbb{R}$. Here is the smallest topology with this property:

Definition

The weak topology on a topological vector space E is the smallest topology such that all $\alpha \in E^*$ are continuous.

Remark. In a pair (E, E^*), we call E the pre-dual and E^* the dual. Beware of the asymmetry: E^* is uniquely determined by E but not vice versa. Moreover, every space is a pre-dual, but not every space is a dual.

Example

The weak and weak* topologies can be strictly smaller than the norm topology (aka. strong topology). For example, the unit vectors e_n in ℓ^2 converge weakly (but not strongly) to zero. To see this, recall that every continuous linear functional on ℓ^2 has the Riesz representation $\langle \cdot, x \rangle_{\ell_2}$ for some $x \in \ell^2$, and note that $\langle e_n, x \rangle_{\ell^2} = x_n \to 0$. □

Finite versus infinite dimensions

More than one topology on the same space E or E^*!? Yes, that's very common in infinite dimensions, but impossible in finite dimensions:

Theorem

On an n-dimensional vector space, there exists one and only one topology such that addition and scalar multiplication are continuous.

This explains why topology is rarely discussed for finite-dimensional vector spaces—there is not much to talk about.

Questions to answer for yourself / discuss with friends

- Repetition: What is a topological vector space? What is its dual space? What is the weak versus strong topology?
- Check your understanding: What's the difference between a collection $(x_i)_{i\in I}$ of elements in E and a function $x : I \to E$? What spaces do they belong to?
- Check your understanding: Does $\|\cdot\|_1 \leq \|\cdot\|_2$ imply $\mathcal{O}_1 \subseteq \mathcal{O}_2$ for the corresponding topologies?
- Transfer: Can you give some examples of topological vector spaces?

MH3520 Chapter 6

Part 3

A zoo of function spaces

Signed measures

Assumption. $(\Omega, \mathcal{F})$ is a measurable space.

Definition

A signed measure is a countably additive function $\mu : \mathcal{F} \to \mathbb{R}$ satisfying $\mu(\emptyset) = 0$.

Definition

$\mathcal{M}(\Omega)$ is the Banach space of signed measures μ with finite total variation norm $\|\mu\|_{\mathcal{M}(\Omega)} = \|\mu\|_1$ defined as

$$\sup\left\{\sum_{i=1}^{n} |\mu(A_i)| : A_1, \ldots, A_n \in \mathcal{F} \text{ disjoint}, n \in \mathbb{N}\right\} < \infty.$$

Remark. Every signed measure μ is the difference $\mu_+ - \mu_-$ between two unsigned measures, and $\|\mu\|_1 = \mu_+(\Omega) + \mu_-(\Omega)$.

Continuous functions

Assumption. K is a compact metric space.

Definition

$C(K)$ is the Banach space of continuous functions

$$f : K \to \mathbb{R}, \qquad \|f\|_{C(K)} := \|f\|_\infty := \sup_{x \in K} |f(x)|.$$

Theorem (Riesz–Kakutani)

- $C(K)$ *is separable.*
- *The dual of* $C(K)$ *is the Banach space* $\mathcal{M}(K)$*, i.e., every continuous linear functional on* $C(K)$ *is given by integration with respect to some measure.*

Continuous bounded functions

Assumption. X is a Hausdorff topological space, and B is a Banach space.

Definition

$C_b(X, B)$ is the Banach space of continuous bounded functions

$$f : X \to B, \qquad \|f\|_{C_b(X,B)} := \|f\|_\infty := \sup_{x \in X} \|f(x)\|_B < \infty.$$

Remark.

- $C_b(\mathbb{R}, \mathbb{R})$ is not separable despite $\mathbb{R}$ being separable.
- The dual of $C_b(X, \mathbb{R})$ consists of measures which are merely finitely additive.
- $C_c(X, B)$ is the subspace of compactly supported continuous functions $X \to B$. It may be incomplete unless X is compact.

Continuously differentiable functions

Assumption. X is an open subset of a normed space E, B is a Banach space, and $k \in \mathbb{N}$.

Definition

$C_b^k(X, B)$ is the Banach space of k times continuously differentiable functions

$$f : X \to B, \qquad \|f\|_{C_b^k(X,B)} := \sum_{j=0}^{k} \| d^j f\|_{C_b(X, L^{(j)}(E,B))} < \infty.$$

Remark. C_b^1 implies Lipschitz.

Hölder functions

Assumption. X is an open subset of a normed space E, B is a Banach space, and $s \in (0, \infty) \setminus \mathbb{N}$, i.e., s is a real but not a natural number.

Definition

$C_b^s(X, B)$ is the Banach space of $k := \lfloor s \rfloor$ times continuously differentiable functions $f : X \to B$ with

$$\|f\|_{C_b^s(X,B)} := \|f\|_{C_b^k(X,B)} + \sup_{x \neq y \in X} \frac{\| d^k f(y) - d^k f(x) \|_{L^{(k)}(E,B)}}{\|y - x\|^{s-k}} < \infty.$$

Remark. C_b^s functions with $s \in (0, 1)$ can be quite rough—think of a sample path of (fractional) Brownian motion.

Bounded smooth functions with bounded derivatives

Assumption. X is an open subset of a normed space E, and B is a Banach space.

Definition

$C_b^\infty(X,B)$ is the complete metric vector space of smooth functions $f : X \to B$ with bounded derivatives of all orders, endowed with the metric

$$d(f,g) := |\!|\!| f-g |\!|\!|_{C_b^\infty(X,B)}, \quad |\!|\!| f |\!|\!|_{C_b^\infty(X,B)} := \sum_{j=0}^{\infty} 2^{-j} \frac{\| d^j f \|}{1+\| d^j f \|},$$

where the norm on the RHS is in $C_b(X, L^{(j)}(E,B))$.

Remark. There is no norm for this topology, only a metric.

Smooth functions and distributions

Assumption. E is Euclidean space, and B is a Banach space.

Definition

$C^\infty(E,B)$ is the complete metric vector space of smooth functions $f : E \to B$, endowed with the metric

$$d(f,g) := [\![f-g]\!]_{C^\infty(E,B)}, \quad [\![f]\!]_{C^\infty(E,B)} := \sum_{r,j=0}^{\infty} 2^{-(i+j)} \frac{\| d^j f \|_r}{1 + \| d^j f \|_r},$$

where the norm on the RHS is in $C_b(B_r(0), L^{(j)}(E,B))$.

Definition

Distributions are continuous linear functionals on the space $C_c^\infty(E,\mathbb{R})$ of compactly supported smooth functions, i.e., elements of $C_c^\infty(E,\mathbb{R})^*$.

L^p functions

Assumption. $(\Omega, \mathcal{F}, \mu)$ is a measure space.

Definition

For any $p \in [1, \infty)$, $L^p(\Omega)$ is the Banach space of measurable functions

$$f : \Omega \to \mathbb{R}, \qquad \|f\|_{L^p(\Omega)} := \Big(\int_\Omega |f|^p \mu \Big)^{1/p} < \infty,$$

modulo equality μ-almost everywhere.

Theorem

Let $p \in [1, \infty)$ and $q \in (1, \infty]$ with $\frac{1}{p} + \frac{1}{q} = 1$.

- *$L^p(\Omega)$ is separable if Ω is a separable metric space endowed with its Borel sigma algebra.*
- *The dual of $L^p(\Omega)$ is $L^q(\Omega)$, i.e., every linear functional on $L^p(\Omega)$ is given by integration with respect to some function in $L^q(\Omega)$.*

Sobolev functions

Assumption. $\mathbb{R}^d$ is the d-dimensional Euclidean space with the Borel measure, $k \in \mathbb{N}$, and $p \in [1, \infty)$.

Definition

$W^{k,p}(\mathbb{R}^d)$ is the Banach space of functions $f : \mathbb{R}^d \to \mathbb{R}$ with (distributional) derivatives up to order k in $L^p(\Omega)$, and

$$\|f\|_{W^{k,p}(\mathbb{R}^d)} := \sum_{j=0}^{k} \| d^j f\|_{L^p(\mathbb{R}^d, L^{(j)}(\mathbb{R}^d, \mathbb{R}))} < \infty.$$

Remark.

- $W^{k,2}(\mathbb{R}^d)$ is a Hilbert space and is denoted by $H^k(\mathbb{R}^d)$.
- The dual of $W^{k,p}$ is $W^{-k,q}$—a space of distributions.

Besov spaces

Assumption. $\mathbb{R}^d$ is the d-dimensional Euclidean space with the Borel measure, $\alpha \in (-\infty, \infty)$, and $p, q \in [1, \infty]$.

Definition

$B^\alpha_{p,q}$ is a Banach space of distributions on $\mathbb{R}^d$, whose definition you do not want to see. It consists of (equivalence classes of) functions if $\alpha \geq 0$.

Besov spaces encompass many of the above-mentioned spaces:

Theorem

The following spaces are equal and carry equivalent norms:

- $C^\alpha(\mathbb{R}^d) = B^\alpha_{\infty,\infty}(\mathbb{R}^d)$ *if* $\alpha \in (0, \infty) \setminus \mathbb{N}$.
- $L^p(\mathbb{R}^d) = B^0_{p,p}(\mathbb{R}^d)$.
- $H^\alpha(\mathbb{R}^d) = W^{\alpha,2}(\mathbb{R}^d) = B^\alpha_{2,2}(\mathbb{R}^d)$.

Questions to answer for yourself / discuss with friends

- Repetition: Continuous functions, continuously differentiable functions, smooth functions, L^p-functions, Sobolev functions, ...
- Discussion: What kind of sequence spaces do you get by using the natural numbers as the domain of the function space?
- Discussion: Let's think about how to prove completeness of the function spaces in our little zoo.

MH3520 Chapter 6

Part 4

Convexity and the Hahn–Banach theorem

Hahn–Banach extension theorem

We have already encountered extensions from dense subspaces (an easy task). Extensions from non-dense subspaces are an entirely different story:

Theorem (Hahn–Banach extension)

If E is a normed space and V a linear subspace, then any continuous linear functional $\alpha : V \to \mathbb{R}$ can be extended to a linear functional $E \to \mathbb{R}$ with the same operator norm.

Remark.

- This requires convexity: normed spaces are locally convex, i.e., every zero-neighborhood contains a convex zero-neighborhood, and the Hahn–Banach theorem generalizes to arbitrary locally convex spaces.
- Completeness is not needed anywhere.
- The proof uses the lemma of Zorn to find a maximal extension and argues that the maximal extension must be defined on all of E.

Proof of the Hahn–Banach extension theorem

The proof uses two lemmas, which are shown subsequently.

Proof of the Hahn–Banach extension theorem.

- If such extensions exists on all W_i in an increasing family of subspaces $(W_i)_{i\in I}$, then an extension exists on $\bigcup_i W_i$.
- By the lemma of Zorn (see next), there exists a maximal extension, defined on some subspace W
- If there was any $y \notin W$, then the lemma on extensions by one dimension (see next) would allow us to extend to $W \oplus (\mathbb{R}y)$.
- This would contradict the maximality of W. Thus, $W = E$. □

Proof of the Hahn–Banach extension theorem

Lemma (Extensions by one dimension)

The theorem holds true if $E = V \oplus \mathbb{R}y$.

Proof. Without loss of generality $\|\alpha\|_{V^*} = 1$.

- We need to find $\alpha(y) \in \mathbb{R}$ such that

$$\forall x \in V: \qquad -\|x + \lambda y\| \leq \alpha(x) + \lambda\alpha(y) \leq \|x + \lambda y\|.$$

- This is automatic for $\lambda = 0$. Otherwise, dividing by λ, replacing x/λ by x, and subtracting $\alpha(x)$ yields

$$\forall x \in V: \qquad -\|x + y\| - \alpha(x) \leq \alpha(y) \leq \|x + y\| - \alpha(x). \qquad (*)$$

- By the triangle inequality, for any $x_1, x_2 \in V$,

$$\alpha(x_1) - \alpha(x_2) = \alpha(x_1 - x_2) \leq \|x_1 - x_2\| \leq \|x_1 + y\| + \|x_2 + y\|.$$

- Thus, in $(*)$, $\sup(\text{LHS}) \leq \inf(\text{RHS})$, enabling a choice of $\alpha(y)$. □

Digression: Zorn's lemma

If you are building a mathematical object in stages and find that (a) you have not finished even after infinitely many stages, and (b) there seems to be nothing to stop you continuing to build, then Zorn's lemma may well be able to help you:

Lemma (Zorn's lemma)

A partially ordered set[1] *with the property that every chain*[2] *has an upper bound,*[3] *contains at least one maximal element.*[4]

Remark.

- The proof is indirect and uses the axiom of choice.
- In Zermelo–Fraenkel set theory, it is equivalent to the axiom of choice.

[1] A set with a reflexive, antisymmetric, and transitive binary relation $\leq$

[2] A totally ordered subset

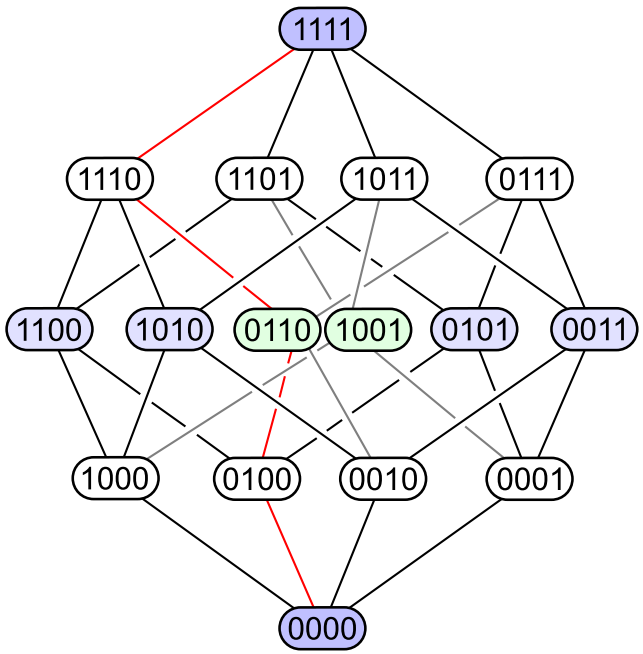
[3] An element that is greater or equal to any element of the chain

[4] An element that is not smaller than any other element

Sketch of proof of Zorn's lemma

Sketch of proof.

- Every partially ordered set contains a chain, which has no strict upper bound:
  - Suppose for contradiction that every chain C had a strict upper bound, denoted by $u(C)$.
  - Defining the function $u(C)$ requires the axiom of choice.
  - Construct a chain C^* starting from some x_0 by adding inductively the strict upper bound u of all previous elements in the chain.
  - By definition, C^* contains $u(C^*)$. However, this contradicts that $u(C^*)$ is a strict upper bound for C^*.
- Let's take a chain without strict upper bound. By assumption, it has an upper bound. This upper bound is maximal.



Implications of the Hahn–Banach theorem

The Hahn–Banach extension theorem ensures that there are 'many' linear functionals: for instance, sufficiently many for distinguishing points in E.

Corollary

If E is normed, then E^* separates points of E, i.e., for any $x, y \in E$ exists $\alpha \in E^*$ with $\alpha(x) \neq \alpha(y)$.

Proof. Extend $\alpha : \mathbb{R}(y-x) \to \mathbb{R}$, $\alpha(y-x) := 1$, to all of E. □

Beyond normed spaces (more precisely, beyond locally convex spaces), the dual may fail to separate points and may even be trivial:

Example

If E is a metric vector space, it may happen that $E^* = 0$. For instance, this happens for $E = L^p([0,1])$ with $p \in [0,1)$.

Implications of the Hahn–Banach theorem (cont.)

Another corollary of Hahn–Banach is that the norm of a vector can be computed by testing with linear functionals:

Corollary

If E is normed, then

$$\forall x \in E: \qquad \|x\| = \sup\left\{|\alpha(x)| : \alpha \in E^*, \|\alpha\| \leq 1\right\}.$$

Proof.

- For any $\alpha \in E^*$ with $\|\alpha\| \leq 1$, one has $|\alpha(x)| \leq \|\alpha\|\|x\| \leq \|x\|$.
- It remains to construct α such that equality holds above.
- Extend $\alpha : \mathbb{R}x \to \mathbb{R}$, $\alpha(x) := \|x\|$, to all of E via Hahn–Banach.
- Then, $\alpha \in E^*$ with $\|\alpha\| = 1$ and $|\alpha(x)| = \|x\|$. □

Implications of the Hahn–Banach theorem (cont.)

The Hahn–Banach theorem implies a very useful criterion for testing whether a linear subspace is dense:

Corollary (very useful criterion of Dense)

A linear subspace V of a normed space E is dense if and only if every continuous linear functional $E \to \mathbb{R}$ which vanishes on V vanishes everywhere.

Proof.

- If V is not dense, then there exists some $y \in E \setminus \overline{V}$, where $\overline{V}$ is the closure of V. Define a linear functional

$$\alpha : \overline{V} \oplus \mathbb{R}y \to \mathbb{R}, \qquad \alpha(x + \lambda y) = \lambda.$$

- α is continuous because its kernel $\overline{V}$ is closed.
- By Hahn–Banach, α extends to a non-zero continuous linear functional on E which vanishes on V.

□

Questions to answer for yourself / discuss with friends

- Repetition: State the Hahn–Banach extension theorem.
- Check your understanding: In Euclidean coordinates, how would you construct a Hahn–Banach extension?
- Useless knowledge: Zorn is the last name of a 20th century mathematician from Germany. Do you know what his name means?
- Discussion: Can you give a finite-dimensional example of a dense subspace?

MH3520 Chapter 6

Part 5

Completeness and the uniform boundedness principle (Optional)

Uniform boundedness principle

Theorem (Banach–Steinhaus)

Let $A_i : E \to F$ be a family of continuous linear functions from a Banach space E to a normed space F, indexed by $i \in I$ for some index set I. Then, point-wise boundedness

$$\forall x \in E : \qquad \sup_{i \in I} \|A_i(x)\| < \infty$$

implies uniform boundedness:

$$\sup_{i \in I} \|A_i\| = \sup_{i \in I} \sup_{\|x\| \leq 1} \|A_i(x)\| < \infty.$$

Remark.

- This requires completeness but no convexity (there are generalizations to complete metric vector spaces, which may not be locally convex).
- The proof is elementary. [Sokal 2011]

No uniform boundedness principle without completeness

Counter-example

Let c_{00} be the set of real-valued sequences with finitely many non-zero entries, endowed with the supremum norm. Then c_{00} is incomplete (why?), and the uniform boundedness principle does not hold: For instance, the family

$$A_n(x) = n x_n{}^{a}, \qquad n \in \mathbb{N}, \quad x \in c_{00},$$

is point-wise bounded but not uniformly bounded.

---

[a] x_n denotes the n-th entry of x

Proof of the uniform boundedness theorem

Here is an auxiliary lemma.

Lemma

Let E and F be normed spaces. For any $A \in L(E,F)$, $x \in E$, and $r > 0$,

$$\|A\|r := \sup_{y \in B_r(0)} \|A(y)\| \leq \sup_{y \in B_r(x)} \|A(y)\|.$$

Proof. By the triangle inequality in the form $\|a-b\| \leq \|a\| + \|b\|$,

$$\|Ty\| \leq \frac{1}{2}\big(\|T(x+y)\| + \|T(x-y)\|\big) \leq \max\big\{\|T(x+y)\|, \|T(x-y)\|\big\}.$$

Now, take $\sup_{y \in B_r(0)}$ on both sides. □

Proof of the uniform boundedness theorem (cont.)

Proof of the uniform boundedness principle.

- Suppose that $\sup_{i\in I} \|A_i\|\infty$.
- Choose $(A_n)_{n\in\mathbb{N}}$ such that $\|A_n\| \geq 4^n$.
- Set $x_0 := 0$, and for $n \geq 1$ use the lemma with $r := 3^{-n}$ to choose inductively $x_n \in E$ such that

$$\|x_n - x_{n-1}\| \leq 3^{-n} \qquad \text{and} \qquad \frac{2}{3}\|A_n\|3^{-n} \leq \|A_n x_n\|.$$

- Then (x_n) is Cauchy, hence convergent to some $x \in E$, and

$$\|x - x_n\| \leq \sum_{j=n}^{\infty} \|x_{j+1} - x_j\| \leq \sum_{j=n}^{\infty} 3^{-(j+1)} = \frac{1}{2}3^{-n}.$$

- It follows that

$$\|A_n x\| \geq \frac{1}{6}3^{-n}\|A_n\| \geq \frac{1}{6}(4/3)^n \to \infty.$$

□

Implication: open mapping theorem

Theorem

Let $A : E \to F$ be a continuous surjective linear function between Banach spaces E and F. Then A is open, i.e., it maps open sets to open sets.

Remark.

- Equivalently, images of open balls contain open balls. It suffices to consider centered balls.
- The open mapping theorem follows in an elementary way from the uniform boundedness theorem. [Eldredge 2014]

Notation. In the sequel, $U_r(0)$ and $V_r(0)$ denote the centered open balls of radius r in E and F, respectively.

Proof of the open mapping theorem

Lemma

If $A : E \to F$ is a surjection between Banach spaces E and F, then

$$\overline{A(U_1(0))} \supseteq V_r(0), \qquad \text{for some } r > 0.$$

Proof. Solve $y = Ax$ by a sequence of regularized optimization problems:

$$G := \big\{(y_n)_{n\in\mathbb{N}} : y_n \in F\big\}, \qquad \|y\|_G := \sup_{n\in\mathbb{N}} \inf_{x\in E} \big(\|x\| + n\|y_n - Ax\|\big),$$

$$S_n : F \to G, \qquad S_n(y) := (0, \dots, 0, y, 0, \dots).$$

- Each S_n is bounded: setting $x = 0$ in $\|\cdot\|_G$ yields $\|S_n(y)\| \le n\|y\|$.
- The family (S_n) is point-wise bounded: choosing $x \in A^{-1}(y)$ in $\|\cdot\|_G$ yields $\|S_n(y)\| \le \|x\|$, independently of n.
- By the uniform boundedness principle, $\|S_n\| \le \frac{1}{r}$, for some $r > 0$.
- If $\|y\| < r$, then $\|S_n y\| \le \|S_n\|\|y\| < 1$, and there exists x_n with $\|x_n\| + n\|y - Ax_n\| < 1$. Thus, $y = \lim_{n\to\infty} Ax_n \in \overline{A(U_1(0))}$. □

Proof of the open mapping theorem (cont.)

Proof of the open mapping theorem:

- We claim that $A(U_1(0) \supseteq V_{r/2}(0)$. Let $y \in V_{r/2}(0)$.
- The lemma implies for all $n \in \mathbb{N}$ that $\overline{A(U_{2^{-n}}(0)} \supseteq V_{r2^{-n}}(0)$.
- Set $x_0 = 0$ and $y_0 = y$. Inductively, for each $n \geq 1$,

$$\text{define } y_n := y_{n-1} - Ax_{n-1}, \qquad \text{note that } \|y_n\| < r2^{-n},$$
$$\text{find } x_n \text{ such that } \|x_n\| < 2^{-n} \text{ and } \|y_n - Ax_n\| < 2^{-(n+1)}.$$

- Then, $x := \sum_n x_n$ converges absolutely, $\|x\| \leq \sum_n 2^{-n} < 1$, and $Ax = \lim_{n\to\infty} Ax_1 + \cdots + Ax_{n-1} = \lim_{n\to\infty} y - y_n = y$. □

Implication: closed graph theorem

Corollary

A continuous bijective linear function between Banach spaces has a continuous inverse.

Proof. A is open if and only if A^{-1} is continuous. □

Corollary (Closed graph theorem)

A linear function between Banach spaces is continuous if and only if it has a closed graph.

Proof. $\Rightarrow$ is trivial. We show $\Leftarrow$.

- Let $G \subseteq E \times F$ be the graph of $A : E \to F$. Then G is Banach.
- The projection $\mathrm{pr}_E |G : G \subseteq E \times F \to E$ is continuous bijective.
- By the open mapping theorem, $\mathrm{pr}_E |G$ has a continuous inverse.
- Thus, $A = \mathrm{pr}_F \circ (\mathrm{pr}_E |G)^{-1}$ is continuous. □

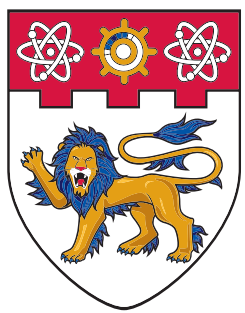
Questions to answer for yourself / discuss with friends

- Repetition: What is the uniform boundedness principle, and under what assumptions does it hold?
- Repetition: What are the closed graph and open mapping theorems?
- Check your understanding: Verify that a continuous mapping necessarily has a closed graph.
- Discussion: Does the notion of completeness make sense on general topological vector spaces?

MH3520:Mathematics of Deep Learning

Chapter 7

Universality of multi-layer perceptrons



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

Last chapter:

- Introduction to functional analysis

This chapter:

- Multi-layer perceptrons are universal in the sense that they can approximate any given function to arbitrary accuracy.

Next chapters:

- Approximation theory: the same statement with quantitative control over the approximation error

Universality of multi-layer perceptrons

- Universality means that any continuous function can be approximated uniformly on compacts by multi-layer perceptrons.
- This is a density result: multi-layer perceptrons are dense in $C(K)$, for each compact set K.
- Thanks to Hahn–Banach, density of linear subspaces can be tested by linear functionals.
- This leads to the universal approximation theorem, which asserts that perceptrons with discriminatory activation function are universal.
- We use the Stone–Weierstrass theorem to find examples of discriminatory activation functions, including the rectified linear unit.

Overview of Chapter 7

1. Universality of multi-layer perceptrons
2. Banach algebras and the Stone–Weierstrass theorem
3. Integral transforms
4. Examples of discriminatory activation functions

Sources for this chapter:

- Helemskii (2006): Lectures and exercises on functional analysis. Mathematical Monographs, Volume 233. American Mathematical Society.
- Petersen (2022): Neural Network Theory. Lecture Notes, University of Vienna. `pc-petersen.eu/Neural_Network_Theory.pdf`

MH3520 Chapter 7

Part 1

Universality of multi-layer perceptrons

Universality

Definition

A set of multi-layer perceptrons with input dimension d and output dimension 1 is universal if it is dense in $C(K)$, for all compact sets $K \subseteq \mathbb{R}^d$.

Remark.

- Our goal is to show for certain activation functions that multi-layer perceptrons with a single hidden layer are universal.
- Universality cannot hold for all activation functions, as shown by the following example.

Counter-example

There is no universality for multi-layer perceptrons with polynomial activation function and bounded depth because these are polynomials of bounded degree.

Discriminatory activation functions

Definition

A measurable function $\rho : \mathbb{R} \to \mathbb{R}$ is called discriminatory if for any $d \in \mathbb{N}$ and compact set $K \subset \mathbb{R}^d$, the zero measure is the only signed measure μ on K which satisfies

$$\forall a \in \mathbb{R}^d, \quad \forall b \in \mathbb{R} : \qquad \int_K \rho\big(\langle a, x\rangle + b\big)\mu(dx) = 0.$$

Remark. In words, a function is discriminatory if the only signed measure which annihilates all it dilations and translations is the zero measure.

Counter-example

Polynomial functions are non-discriminatory because their dilations and translations are polynomials of the same or lesser degree. This imposes finitely many constraints on an infinite-dimensional measure space of measures, allowing for non-zero annihilating measures.

Universal approximation theorem

Theorem (Cybenko)

If ρ is discriminatory, then the set of multi-layer perceptrons of the form

$$h(x) = \sum_{i=1}^{n} w_i \rho(\langle a_i, x \rangle + b_i), \qquad n \in \mathbb{N}, \quad w_i \in \mathbb{R}, \quad a_i, b_i \in \mathbb{R}^d,$$

is universal.

Remark.

- For deeper networks, one has to assume additionally that the lower layers can represent or approximate the identity.
- Universality is necessary for deep learning to have any chance to work.
- However, it is not sufficient as an explanation for why deep learning works so well—many other function classes such as polynomials, trigonometric polynomials, etc. are also universal.

Proof of the universal approximation theorem

Proof.

- Let $\mathcal{H}$ be the set of multi-layer perceptrons of the form in the theorem, seen as a linear subspace of $C(K)$, for some compact set $K \subset \mathbb{R}^d$.
- Recall that the dual of $C(K)$ is the Banach space $\mathcal{M}(K)$ of signed measures with finite total variation.
- Any $\mu \in \mathcal{M}(K)$ which satisfies

$$\forall h \in \mathcal{H} : \qquad \int_K h(x)\mu(dx) = 0$$

  satisfies in particular that

$$\forall a \in \mathbb{R}^d, \quad \forall b \in \mathbb{R} : \qquad \int_K \rho\big(\langle a, x\rangle + b\big)\mu(dx) = 0.$$

- As ρ is discriminatory, the only such μ is the zero measure.
- By the Hahn–Banach theorem, $\mathcal{H}$ is dense in $C(K)$. □

Questions to answer for yourself / discuss with friends

- Repetition: Recount the universal approximation theorem and its proof.
- Complete the sentence: If universality holds, then for any given continuous function f...
- Discussion: Can you give an example of a discriminatory and/or a non-discriminatory activation function?
- Discussion: Are there similar results for multiple output dimensions?
- Discussion: How does Cybenko's universality theorem differ from the Stone–Weierstrass approximation theorem?

MH3520 Chapter 7

Part 2

Banach algebras and the Stone–Weierstrass theorem

Banach algebras

Definition

A Banach algebra is a Banach space A endowed with a multiplication, which is a continuous bilinear map $A \times A \to A$ satisfying the associativity identity $(ab)c = a(bc)$.

Example

$C(K)$ is a Banach algebra, for any compact space K.

Example

$L(E, E)$ is a Banach algebra, for any Banach space E.

Stone–Weierstrass theorem

Theorem (Stone–Weierstrass)

Let K be a compact topological space, and let $A \subseteq C(K, \mathbb{R})$ satisfy

- *A is an algebra, i.e., closed under linear combinations and products,*
- *A contains all constant functions, and*
- *A is point-separating, i.e., for any $x \neq y \in K$ exists $f \in A$ such that $f(x) \neq f(y)$.*

Then, A is dense in $C(K, \mathbb{R})$.

Remark.

- Proof: `https://web.math.utk.edu/~freire/teaching/m447f16/StoneWeierstrassNotes.pdf`
- We next consider some implications.

Polynomial approximation

Corollary (Weierstrass)

Polynomials are dense in the Banach space $C([0,1],\mathbb{R})$.

Proof. The polynomials form an algebra, which contains all constant functions and separates points. □

Remark. There is a simple probabilistic proof of this corollary: if $X_1, X_2, \ldots$ are iid. $\mathrm{Ber}(x)$-distributed, then

$$\mathbb{E}[f(\overline{X}_n)] = \underbrace{\sum_{i=0}^{n} f\left(\frac{i}{n}\right)\binom{n}{i}x^i(1-x)^{n-i}}_{\text{polynomial}} \xrightarrow[n\to\infty]{} f(x),$$

and one can argue that this convergence is uniform in x.

Trigonometric approximation

A trigonometric polynomial is a function of the form

$$f : [0,1] \to \mathbb{R}, \qquad f(x) = a_0 + \sum_{k=1}^{n} \Big(a_k \cos(2\pi k x) + b_k \sin(2\pi k x) \Big).$$

Corollary (Weierstrass)

Trigonometric polynomials are dense in the Banach space $C([0,1], \mathbb{R})$.

Proof. The trigonometric polynomials form an algebra, which contains all constant functions and separates points. □

Questions to answer for yourself / discuss with friends

- Repetition: What does the Stone–Weierstrass theorem state in its general form? What does it state for polynomials or trigonometric polynomials?
- Check your understanding: Can the compactness condition on the domain of $C(K)$ be dropped?
- Transfer: What is the implication of Stone–Weierstrass for Fourier series and splines?

MH3520 Chapter 7

Part 3

Integral transforms

Fourier transform

Definition

The Fourier transform of a signed Borel measure $\mu \in \mathcal{M}(\mathbb{R}^d)$ is

$$\hat{\mu} : \mathbb{R}^d \to \mathbb{C}, \qquad \hat{\mu}(y) = \int_{\mathbb{R}^d} \exp(-i\langle x, y\rangle)\mu(dx).$$

Remark.

- The Fourier transform of an integrable function $f : \mathbb{R}^d \to \mathbb{R}$ is defined as the Fourier transform of $f\mu$, where μ is the Lebesgue measure.
- The Fourier transform of a random variable $X : \Omega \to \mathbb{R}^d$ is the Fourier transform of its distribution, i.e., $\hat{X}(y) = \mathbb{E}[\exp(-i\langle X, y\rangle)]$.

Theorem

The Fourier transform is injective, i.e., μ can be recovered from $\hat{\mu}$.

Proof of the injectivity of the Fourier transform

Proof. Let $\mu \in \mathcal{M}(\mathbb{R}^d)$ satisfy $\hat{\mu} = 0$.

- Given $\epsilon > 0$ and $f \in C_c(\mathbb{R}^d)$, choose r so large that f is supported in $K := [-r, r]^d$ and that $\|\mathbb{1}_{\mathbb{R}^d \setminus K}\mu\|_1 \leq \epsilon$.
- By Stone–Weierstrass, there is a trigonometric polynomial $p : \mathbb{R}^d \to \mathbb{C}$ of period $2r$ with $\|f - p\|_{C(K)} \leq \epsilon$. Thus,

$$\begin{aligned}\left|\int_{\mathbb{R}^d} f\mu\right| &\leq \left|\int_K (f-p)\mu\right| + \left|\int_{\mathbb{R}^d\setminus K} \underbrace{(f-p)}_{=p}\mu\right| + \left|\underbrace{\int_{\mathbb{R}^d} p\mu}_{=0}\right| \\ &\leq \underbrace{\|f-p\|_{C(K)}}_{\leq\epsilon}\|\mu\|_{\mathcal{M}(\mathbb{R}^d)} + \underbrace{\|p\|_{C_b(\mathbb{R}^d)}}_{\leq\epsilon+\|f\|_{C(K)}}\underbrace{\|\mu\|_{\mathcal{M}(\mathbb{R}^d\setminus K)}}_{\leq\epsilon}.\end{aligned}$$

- As ϵ was arbitrary, $\int f\mu = 0$. As f was arbitrary, μ vanishes on compacts. By dominated convergence, μ vanishes everywhere. □

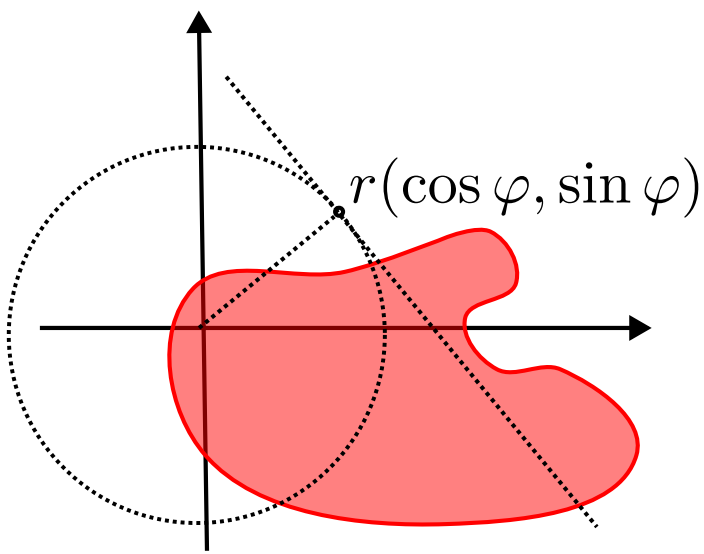
Radon transform for functions on $\mathbb{R}^2$

The Radon transform of a function is the collection of all its line integrals:

Definition

The Radon transform of a compactly supported continuous function $f \in C_c(\mathbb{R}^d, \mathbb{R})$ is the function $\overline{f} : (0, \infty) \times [0, 2\pi)$,

$$\overline{f}(r, \varphi) = \int_{-\infty}^{\infty} f\big(r(\cos\varphi, \sin\varphi) + t(-\sin\varphi, \cos\varphi)\big)\, dt$$



Radon transform for measures

- The Radon transform in higher dimensions is defined by integration over hyperplanes, i.e., co-dimension 1 subspaces.
- For measures in place of functions, we thicken the hyperplane:

Definition

The Radon transform of a signed Borel measure $\mu \in \mathcal{M}(\mathbb{R}^d)$ is

$$\overline{\mu} : \mathbb{R}^d \times \mathbb{R} \times \mathbb{R} \to \mathbb{R}, \qquad \overline{\mu}(a, b_1, b_2) = \mu\Big(\{x \in B : \langle a, x\rangle \in (b_1, b_2]\Big).$$

Theorem

The Radon transform is injective, i.e., μ can be recovered from $\overline{\mu}$.

Proof of the injectivity of the Radon transform

Proof.

- Let μ be a signed measure with vanishing Radon transform.
- For all indicator functions $g = \mathbb{1}_{(b_1,b_2]}$,

$$\int_B g(\langle a, x\rangle)\mu(dx) = 0.$$

- By linearity, this holds for all elementary integrands g.
- By approximation, this holds for all bounded measurable g.
- In particular,

$$\int_B \exp(-i\langle a, x\rangle)\mu(dx) = 0.$$

- By the injectivity of the Fourier transform, $\mu = 0$. □

Questions to answer for yourself / discuss with friends

- Repetition: Describe the Fourier transform and Radon transform of a measure.
- Check your understanding: Why did we have to thicken the hyperplane?
- Transfer: What about the Laplace transform, is it well defined and injective on signed measures?

MH3520 Chapter 7

Part 4

Examples of discriminatory activation functions

Sigmoidal functions

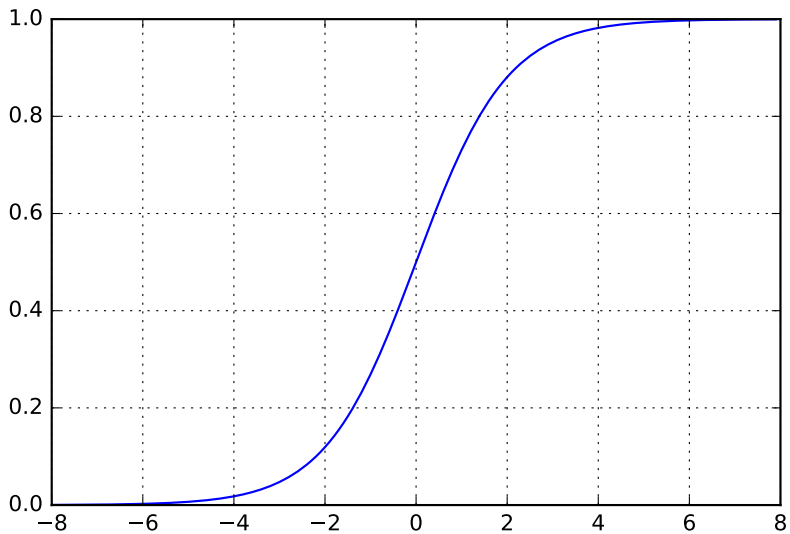
Definition

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 $x \to -\infty$.

Theorem (Cybenko)

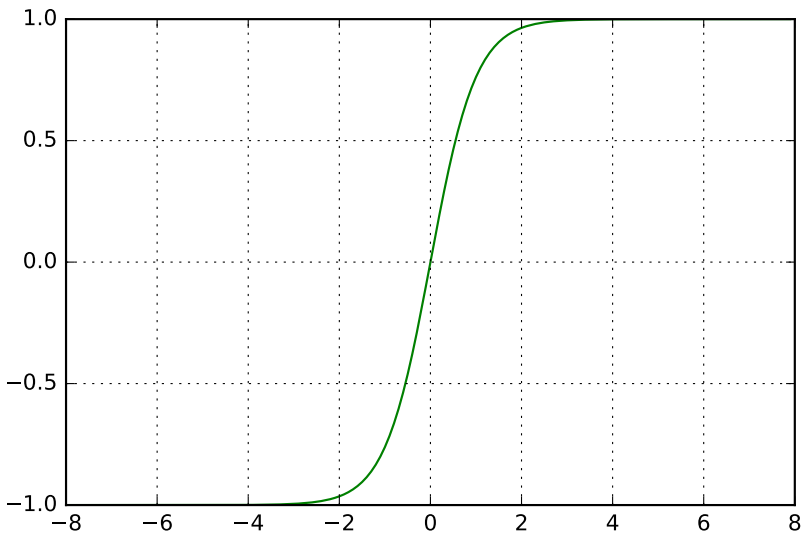
Sigmoidal activation functions are discriminatory.

Examples of sigmoidal functions



Logistic (aka. sigmoidal) function

$$\rho(x) = (1 + e^{-x})^{-1}$$



Hyperbolic tangent

$$\rho(x) = \tanh(x)$$

Proof that sigmoidal functions are discriminatory

Proof.

- Let $\mu \in \mathcal{M}(K)$ such that $\int_K \rho(\langle a, x\rangle - b)\mu(dx) = 0$ for $a \in \mathbb{R}^d, b \in \mathbb{R}$
- As ρ is sigmoidal, one has for any $\theta \in \mathbb{R}$ that

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

- Thus, by dominated convergence,

$$\mu(\{\langle a, x\rangle > b\}) + \rho(\theta)\mu(\{\langle a, x\rangle = b\})$$
$$= \lim_{\lambda\to\infty} \int_K \rho(\lambda(\langle a, x\rangle - b) + \theta) d\mu(x) = 0.$$

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

$$\forall a \in \mathbb{R}^d, \quad \forall b \in \mathbb{R} : \qquad \mu(\{\langle a, x\rangle > b\}) = 0.$$

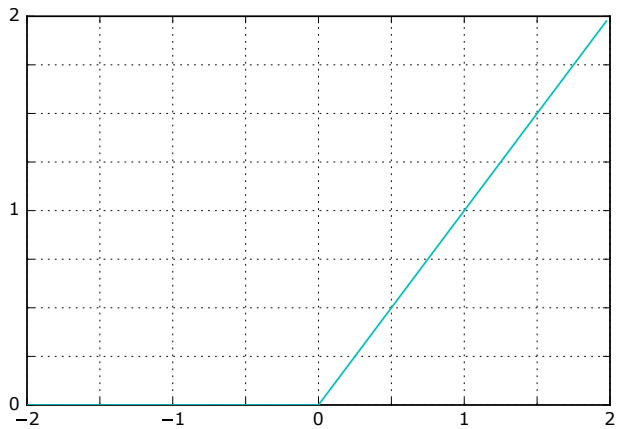
- By linearity, $\mu(\{\langle a, x\rangle \in (b_1, b_2]\}) = 0$, i.e., the Radon transform of μ vanishes. As the Radon transform is injective, $\mu = 0$. □

Rectified linear units

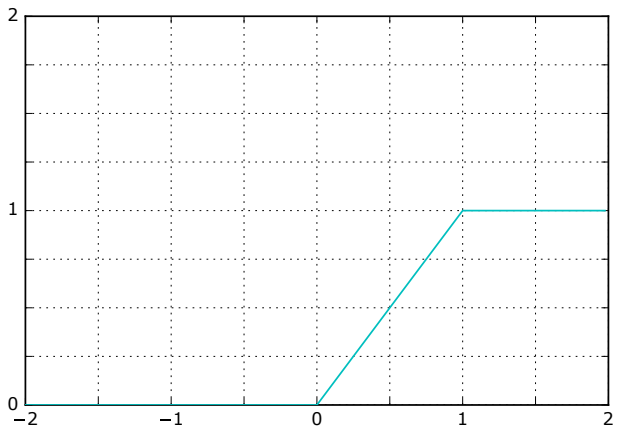
Theorem

The rectified linear unit $x \mapsto \max\{0, x\}$ is discriminatory.

Proof. The function $\eta(x) := \rho(x) - \rho(x-1)$ is sigmoidal, thus discriminatory.



$$\rho(x) = \max\{0, x\}$$



$$\eta(x) = \rho(x) - \rho(x-1)$$

Then for any μ vanishes on ρ will then vanishes on η. Since η is discriminatory then $\mu = 0$, which implies ρ is discriminatory. □

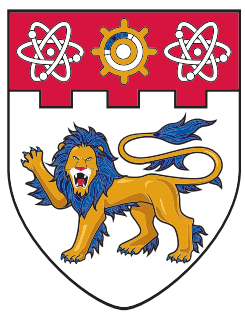
Questions to answer for yourself / discuss with friends

- Repetition: Give some examples of discriminatory activation functions.
- Check your understanding: Are sigmoidal functions bounded?
- Check your understanding: Is being discriminatory necessary *and* sufficient for universality?
- Discussion: Are multi-layer perceptrons dense in $L^p(K)$ for finite p? What about $p = \infty$?

MH3520:Mathematics of Deep Learning

Chapter 8

Basics of approximation theory



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

Last chapter: universality of multi-layer perceptrons

- Any continuous function can be approximated uniformly on compacts by multi-layer perceptrons.

This chapter: approximation theory

- Quantitative control over the approximation error
- Approximation rates for Banach frames and orthonormal bases
- Main result: high approximation rate if a sequence decays fast or a function is smooth

Next chapters: approximation by multi-layer perceptrons

- Approximation rates for multi-layer perceptrons
- Deduced from approximation rates for splines or wavelets

Approximation theory

Approximation theory:

- How well can a possibly complicated function be approximated by some class of simpler, easier to compute functions?
- The approximating functions could be multi-layer perceptrons, polynomials, trigonometric polynomials, splines, wavelets, or anything else one can compute with.
- Approximation theory connects the best-approximation rate to the degree of smoothness of the target function.

Relations to other fields:

- Origins in numerical analysis
- Related to harmonic analysis (useful classes of simple functions) and coding theory (information-theoretic perspective)
- Applications in signal processing (e.g. audio, image, and video codecs)

Overview of Chapter 8

1. Approximation theory
2. Banach frames and orthonormal bases
3. Approximation in sequence spaces
4. Fourier approximation (Optional)

Sources for this chapter:

- Pietsch (1981): Approximation spaces. Journal of Approximation Theory 32, 115–134.
- DeVore and Lorentz (1993): Constructive approximation. Springer.
- Christensen (2016): An introduction to frames and Riesz bases, 2nd edition. Birkhäuser.

MH3520 Chapter 8

Part 1

Approximation theory

Approximation theory

Approximation theory:

- The goal is to approximate $f \in X$ by $h \in H_n \subseteq X$
- The approximation error is measured in X
- Typically, X is a function space.

Example

$X = C([0,1])$, and for each $n \in \mathbb{N}$, H_n is a set of...

- polynomials of degree n,
- trigonometric polynomials of degree n,
- piece-wise polynomials of fixed degree on n intervals,
- n-term linear combination of wavelets, curvelets, ridgelets,
- multi-layer perceptrons with n non-zero weights, etc.

Remark. H stands for hypothesis class. The hypothesis classes H_n are often defined in terms of dictionaries; see next.

Dictionaries and non-sparse approximation

A dictionary is a collection of elements in X.

Example

Some common dictionaries are monomials, trigonometric monomials, spline bases, wavelet frames and, in neural network theory, dilations and translations of the activation function.

The most common use of dictionaries is approximation by the hypothesis class spanned by the first n dictionary items:

Definition

For non-sparse approximation from a dictionary $(\phi_n)_{n\in\mathbb{N}}$, one uses the sets $H_n = \mathtt{H}_n(\phi)$ of linear combinations from the first n dictionary items:

$$\mathtt{H}_n(\phi) = \left\{ \sum_{i=1}^{n} c_i \phi_i : c_i \in \mathbb{R} \right\}.$$

Sparse approximation

We next discard the constraint that only the first n dictionary items are allowed. However, we retain the sparsity constraint that only n coefficients are non-zero.

Definition

For sparse approximation from a dictionary $(\phi_\lambda)_{\lambda\in\Lambda}$, one uses the sets $H_n = \Sigma_n(\phi)$ of n-term linear combinations of dictionary items:

$$\Sigma_n(\phi) = \left\{ \sum_{\lambda\in\Lambda} c_\lambda \phi_\lambda : c_\lambda \neq 0 \text{ at most } n \text{ times} \right\}.$$

Remark. Note that $H_n = \Sigma_n(\phi)$ is nonlinear; it merely satisfies $H_n + H_n \subseteq H_{2n}$. Hence, one speaks of nonlinear approximation.

Sparse approximation with polynomial search

Sparse approximation requires a search through the entire dictionary for finding the n most relevant items—algorithmically, an impossible task.

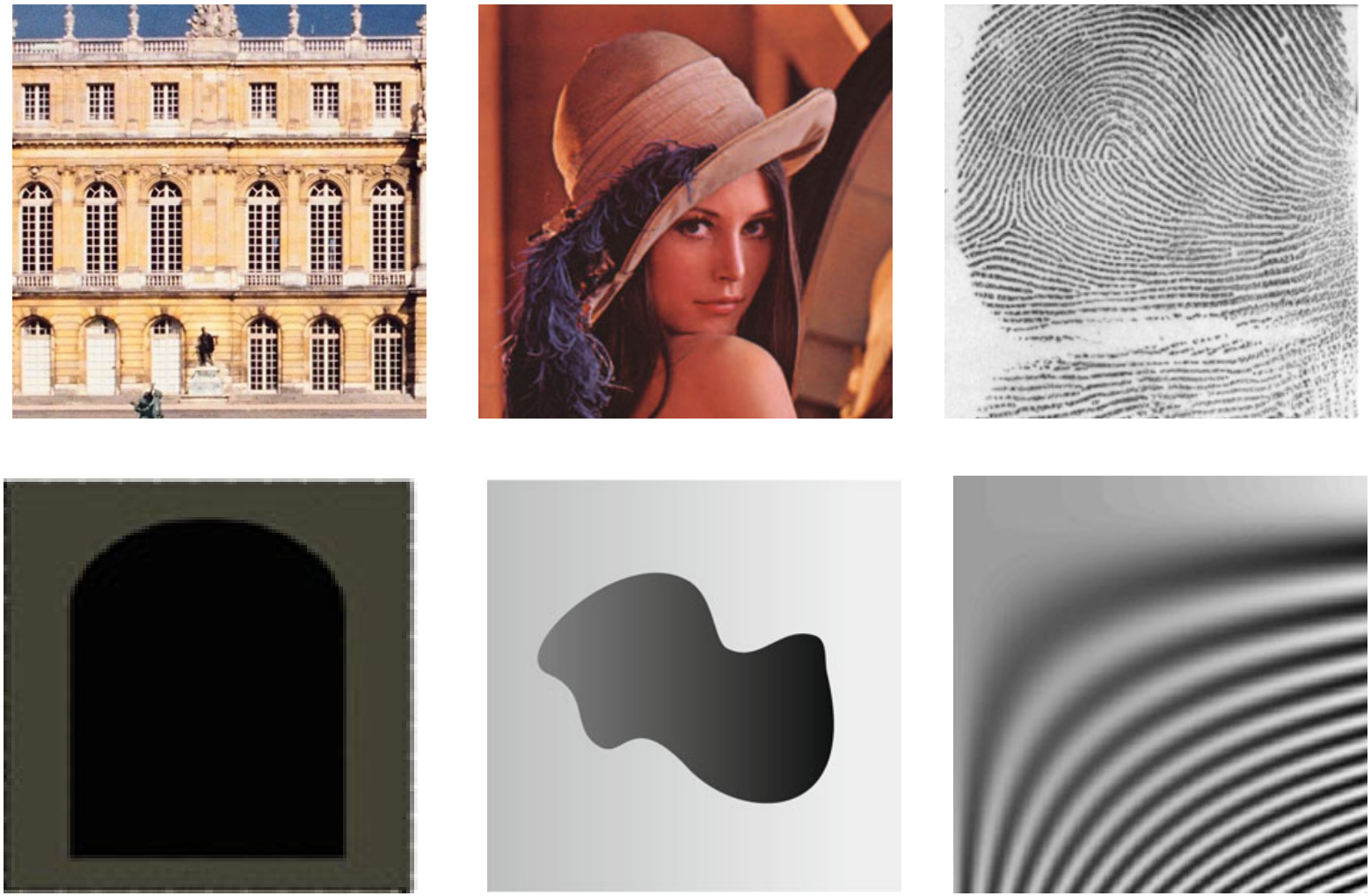
Definition

For sparse approximation with polynomial search from a dictionary $(\phi_n)_{n\in\mathbb{N}}$, one specifies a univariate polynomial π and defines

$$\Sigma_n^\pi(\phi) = \left\{ \sum_{i=1}^{\pi(n)} c_i \phi_i : c_\lambda \neq 0 \text{ at most } n \text{ times} \right\}.$$

Searching through $\pi(n)$ dictionary items is implementable but can still be quite costly. Improvements are studied in compressed sensing.

Example: dictionaries in image processing



Real-world images (top) can be expressed in terms of dictionaries of simpler image elements (bottom). [Dahlke, Fig. 5.1–3]

Quasi-norms

Technical detail: For some of the 'norms' which naturally occur in approximation theory, the triangle inequality holds merely up to a constant:

Definition

A quasi-norm on a linear space X is a function $\|\cdot\| : X \to [0, \infty)$ s.t.

- Non-degeneracy: $\|x\| = 0$ if and only if $x = 0$,
- Absolute homogeneity: $\|\lambda x\| = |\lambda| \|x\|$, and
- Quasi-triangle inequality: $\|x + y\| \leq C\big(\|x\| + \|y\|\big)$, for some $C \geq 1$.

Example

For any measure space $(\Omega, \mathcal{F}, \mu)$ and separable Banach space B, the space $L_p(\Omega, B)$ with $p \in (0, 1)$ carries the quasi-norm

$$\|f\|_{L_p(\Omega)} = \left(\int_\Omega \|f\|_p \mu \right)^{1/p}.$$

Quasi-normed spaces as metric spaces

The p-th power of the L_p quasi-norm satisfies the triangle inequality. This is a general feature:

Lemma

There exists an equivalent quasi-norm, still denoted by $\|\cdot\|$, such that

$$\|x+y\|_p \le \|x\|^p + \|y\|^p, \qquad \text{where } p := (1+\log_2 C)^{-1}.$$

Remark. Thus, a quasi-normed space is a metric vector space, and completeness is well defined. However, local convexity fails, in general.

Approximation error

Standing assumption.

- $(X, \|\cdot\|_X)$ is a quasi-normed space
- $\{0\} = H_0 \subseteq H_1 \subseteq H_2 \subseteq \dots$ are subsets of X
- We write H as a short-hand for $(H_n)_{n\in\mathbb{N}}$.

Definition

For any $n \in \mathbb{N}$, the n-th approximation error of $f \in X$ from H_n is

$$E_n(f) := \inf_{h\in H_n} \|f - h\|_X.$$

Remark.

- To specify X and H, we also write $E_n(f, X, H)$.
- $E_0(f) = \|f\|_X$, and $E_n(f)$ is decreasing in n.
- We next quantify how quickly $E_n(f)$ decays to zero.

Approximation rate

We focus on polynomial approximation rates:

Definition

The approximation rate of a set $Y \subseteq X$ is defined as

$$a^*(Y) := \sup\left\{\alpha > 0 : \sup_{f\in Y}\sup_{n\in\mathbb{N}} n^\alpha E_n(f) < \infty\right\}.$$

For a single $f \in X$, we set $a^*(f) := a^*(\{f\})$.

Remark.

- To specify X and H, we also write $a^*(Y, X, H)$.
- $a^*(f) = \alpha$ iff $E_n(f) = O(n^{-\beta})$ for all $\beta < \alpha$ and no $\beta > \alpha$.

Approximation space

We next consider the set of all $f \in X$ with approximation rate α:

Definition

For any approximation rate $\alpha \in (0, \infty)$, the approximation space $A^\alpha(X, H)$ consists of all $f \in X$ such that

$$\|f\|_{A^\alpha(X,H)} := \sup_{n \geq 1} n^\alpha E_{n-1}(f) < \infty.$$

Remark. The notation suggests that this is a norm or quasi-norm. This is indeed the case under mild conditions; see next.
This implies, $\forall n \geq 1$, $E_{n-1}(f) \leq n^\alpha \|f\|_{A^\alpha(X,H)}$

Approximation quasi-norm

Standing assumption.

- $(X, \|\cdot\|_X)$ is a quasi-normed space
- $\{0\} = H_0 \subseteq H_1 \subseteq H_2 \subseteq \dots$ are subsets of X
- There exists a constant $c \in \mathbb{N}$ such that for all $n \in \mathbb{N}$ and $\lambda \in \mathbb{R}$,

$$\lambda \cdot H_n \subseteq H_n, \qquad H_n + H_n \subseteq H_{cn}.$$

Lemma

$A^\alpha(X, H)$ is a linear quasi-normed space. Moreover, if X is normed and H_n are linear, then $A^\alpha(X, H)$ is normed.

Approximation quasi-norm: proof

Proof.

- Absolute homogeneity of $\|\cdot\|_{A^\alpha(X,H)}$ follows from

$$E_n(\lambda f) = |\lambda| E_n(f).$$

- The quasi-triangle inequality for $\|\cdot\|_{A^\alpha(X,H)}$ follows from

$$\begin{aligned} E_{cn}(f+g,H) &= \inf_{h\in H_{cn}} \|f+g-h\|_X \\ &\le C \inf_{h,k\in H_n} \|f-h\|_X + \|g-k\|_X = C\big(E_n(f,H)+E_n(g,H)\big). \end{aligned}$$

- This implies that $A^\alpha(X,H)$ is a linear quasi-normed space.
- X normed and H_n linear implies $c = C = 1$, and one gets a norm. □

Embeddings and completeness

Lemma

$A^\alpha(X,H) \subseteq A^\beta(X,H) \subseteq X$ *with continuous embeddings if* $\alpha > \beta$.

Proof. The $A^\alpha(X,H)$ quasi-norm is monotonic in α and dominates the quasi-norm in X because $\|f\|_X = E_0(f)$. □

Lemma

$A^\alpha(X,H)$ *is complete if* X *is complete.*

Proof.

- Let (f_m) be Cauchy in $A^\alpha(X,H)$.
- Then, the $A^\alpha(X,H)$-norm of all f_m's is bounded by some M.
- If X is complete, f_m converges in X to some f.
- Then, the $A^\alpha(X,H)$-norm of f is also bounded by M. □

Representation and denseness

I'll next show some more advanced properties of approximation spaces to highlight the elegance of the construction.

Theorem (Representation)

$f \in A^\alpha(X,H)$ if and only if it can be represented by $f_n \in H_{2^n}$ such that

$$f = \sum_{n\in\mathbb{N}} f_n, \qquad \sup_{n\in\mathbb{N}} 2^{\alpha n}\|f_n\|_X < \infty.$$

The infimum of the RHS over all representations (f_n) of f defines an equivalent quasi-norm on $A^\alpha(X,H)$.

Corollary (Denseness)

$\bigcup_{n\in\mathbb{N}} H_n$ is dense in $A^\alpha(X,H)$.

Iteration and interpolation

Theorem (Iteration)

$$A^{\alpha}(A^{\beta}(X,H),H) = A^{\alpha+\beta}(X,H).$$

Interpolation is a method of assigning to spaces X and Y an intermediate space $(X,Y)_{\theta,p}$ parameterized by $\theta \in (0,1)$ and $p \in (0,\infty]$.

Example

$$\big(C([0,1]), C^1([0,1])\big)_{\theta,\infty} = C^{\theta}([0,1]),$$
$$\big(L_p([0,1]), W^{1,p}([0,1])\big)_{\theta,p} = W^{\theta,p}([0,1]).$$

Theorem (Interpolation)

$$\big(A^{\alpha}(X,H), A^{\beta}(X,H)\big)_{\theta,\infty} = A^{(1-\theta)\alpha+\theta\beta}(X,H).$$

Direct and inverse estimates

A typical task in approximation theory is to investigate for some quasi-normed subspace Y of X (think of Hölder or Sobolev functions) whether...

- a direct estimate holds:

$$\|\cdot\|_{A^\alpha(X,H)} \lesssim \|\cdot\|_Y, \quad \text{implying that} \quad Y \subseteq A^\alpha(X,H),$$

- or an inverse estimate holds:

$$\|\cdot\|_Y \lesssim \|\cdot\|_{A^\alpha(X,H)} \quad \text{implying that} \quad A^\alpha(X,H) \subseteq Y.$$

The terminology has historical reasons: direct estimates are shown by constructing good approximations for any given $f \in Y$, whereas inverse estimates involve some reverse-engineering of this process.

Questions to answer for yourself / discuss with friends

- Repetition: Recall the definitions of approximation error, approximation rate, and approximation space.
- Check your understanding: Is $A^\alpha(X, H)$ large for large or small α? Similarly for H?
- Check your understanding: What's the approximation rate of $A^\alpha(X, H)$?
- Check your understanding: Is the set $\Sigma_n(\phi)$, which consists of n-term linear combinations in the dictionary ϕ, a linear space?

MH3520 Chapter 8

Part 2

Banach frames and orthonormal bases

What are good dictionaries?

A good dictionary should provide continuous operations for translating functions $f \in X$ into dictionary coefficients $(c_k) \in E$:

Definition

Let X be a Banach space, and let E be a Banach space of real-valued sequences. A Banach frame for X with respect to E is given by

- Analysis: A continuous linear operator $A\colon X \to E$, and
- Synthesis: A continuous linear operator $S\colon E \to X$, such that
- Reconstruction: the relation $S \circ A = \mathrm{Id}_X$ must hold.

Remark:

- S is surjective but may be non-injective; several coefficient sequences $(c_k) \in E$ may represent the same function $f \in X$.
- A is injective but may be non-surjective; for instance, A could select particularly sparse coefficient sequences (c_k).
- Every separable Banach space has a Banach frame.

What are good sequence spaces?

The definition of Banach frames is very general. To be of practical use, the sequence space must be 'nice'. Here is a minimal requirement.

Definition

The sequence space E is called nice[4] if for all sequences a and b,

$$\begin{aligned} \forall n |a_n| \le |b_n| \text{ and } b \in E \quad &\Longrightarrow \quad a \in E \text{ and } \|a\|_E \le \|b\|_E, \\ b \in E \quad &\Longrightarrow \quad \lim_{n\to\infty} \|b - \mathbb{1}_{\{1,\dots,n\}} b\|_E = 0. \end{aligned}$$

Remark. These are very mild conditions, which are met by all classical sequence spaces ℓ_p but not for ℓ_∞.

[4] Own definition; it means that E is solid and has the unit vectors as Schauder basis.

Now, we have some good dictionaries

Banach frames with nice sequence spaces are good dictionaries:

Theorem

Let $A : X \to E$ and $S : E \to X$ be a Banach frame, for a nice sequence space E with unit vectors e_k, $k \in \mathbb{N}$. Then, $\phi_k := S(e_k)$ is a dictionary, also known as frame, and one has an atomic decomposition

$$\forall f \in X : \qquad f = \sum_{k\in\mathbb{N}} c_k \phi_k,$$

with coefficients $c_k := e_k^*(A(f))$ depending continuously on f.

Proof.

$$c = \sum_{k\in\mathbb{N}} e_k^*(c) e_k, \quad f = SAf = S\left(\sum_{k\in\mathbb{N}} e_k^*(Af) e_k\right) = \sum_{k\in\mathbb{N}} \underbrace{e_k^*(Af)}_{c_k} \underbrace{Se_k}_{\phi_k}.$$

□

Remark. We will see further good properties of such dictionaries later on.

Some overly small and overly large dictionaries

Here's what happens when the dictionary is too small:

Example

If the span of the dictionary is not dense in X, then some $f \in X$ cannot be approximated by linear combinations of dictionary items.

And here is what happens when the dictionary is too large:

Example

If the dictionary itself is dense in X, then any $f \in X$ can be approximated to arbitrary accuracy by 1-term linear combinations. However, this one-hot encoding needs lots of storage and cannot be implemented algorithmically.

Bases are overly large dictionaries

Definition

A basis of a vector space is a linearly independent subset which spans all of the vector space.

Remark.

- Every vector space has a basis (by the lemma of Zorn).
- Only finite linear combinations of basic elements are allowed. This is problematic in infinite dimensions.

Example

Every basis in an infinite-dimensional Banach space is uncountable.

Thus, bases are even worse than dense dictionaries: they are way too big.

Orthonormal bases are good dictionaries

Definition

An orthonormal basis in a Hilbert space is a countable set of orthonormal vectors with dense linear span.

Lemma

An orthonormal basis provides an isometric isomorphism from the Hilbert space to ℓ_2. In particular, it is a Banach frame.

Remark. In this sense, ℓ_2 is the only separable Hilbert space.

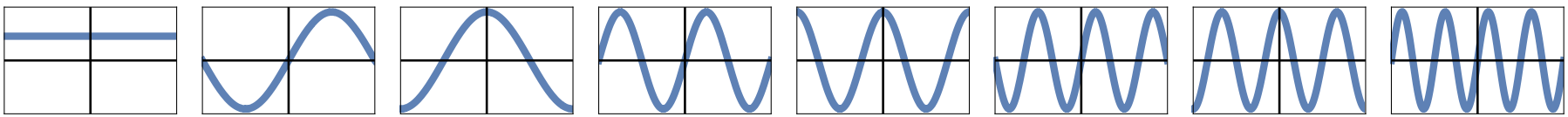
Proof. Let (b_n) be an orthonormal basis in a Hilbert space X.

- Set $A : X \to \ell_2$, $Af(n) := \langle f, b_n \rangle$
- Then A is an isometric isomorphism.
- Its inverse $S = A^{-1}$ is then also an isometric isomorphism. □

Examples of orthonormal bases

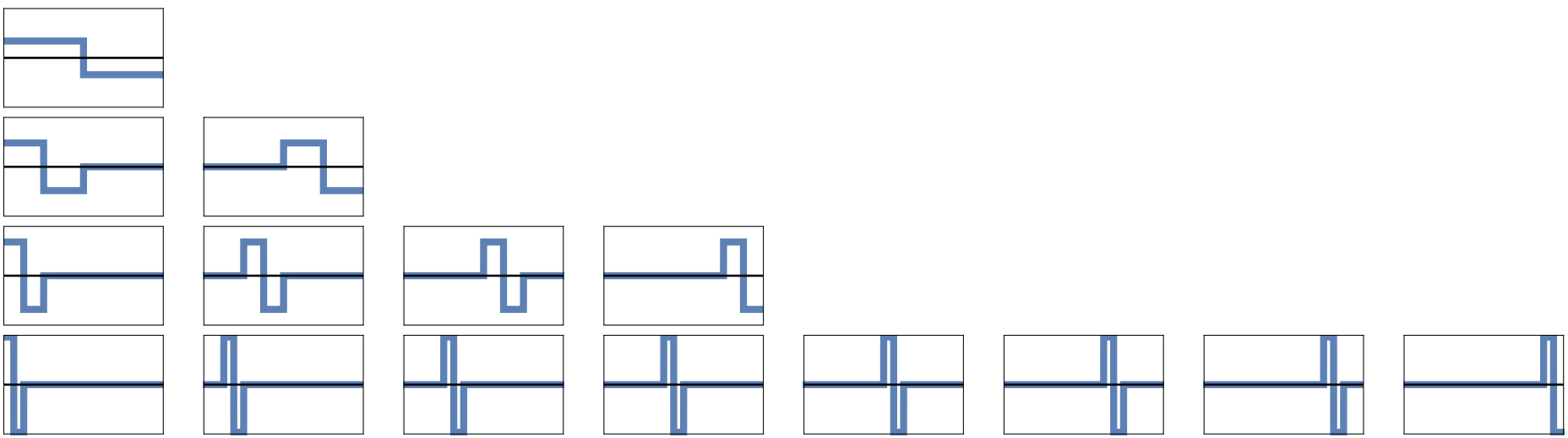
Example

The trigonometric monomials are an orthonormal basis in $L^2([0,1])$.



Example

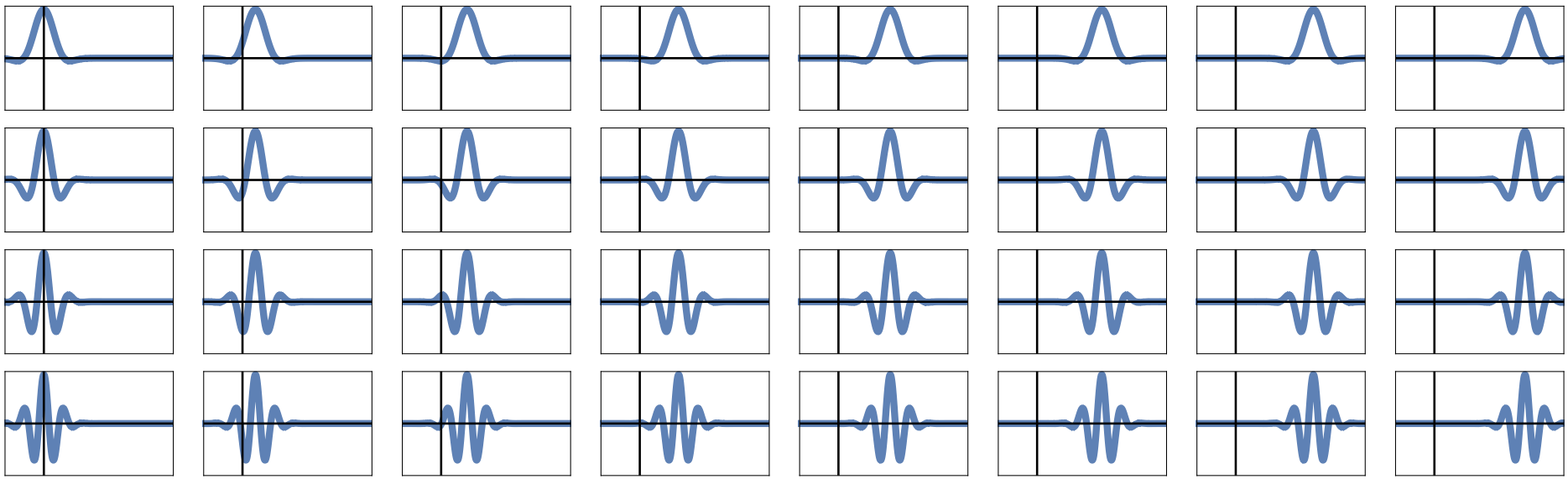
The Haar wavelet system is an orthonormal basis in $L^2([0,1])$.



Examples of Banach frames

Example

The Gabor system is a Banach frame in $L^2(\mathbb{R})$ but not an orthonormal basis.



Remark. The Gabor system performs a time-frequency analysis, also known as short-time Fourier transform, which computes an amplitude for each time *and* each frequency.

Approximation by Banach frames

Assumption. Banach frame $A : X \to E$, $S : E \to X$, where E is a nice sequence space with unit vectors (e_k), and associated frame $\phi = S(e_k)$.

Theorem

For any $\alpha \in (0, \infty)$ and $f \in X$,

$$\|f\|_{A^\alpha(X, \mathtt{H}(\phi))} \lesssim \|Af\|_{A^\alpha(E, \mathtt{H}(e))},$$

and similarly for Σ or Σ^π in place of $\mathtt{H}$.

Remark.

- In words: if Af is well approximated by finite sequences, then f is well approximated by finite linear combinations of dictionary items.
- This reduces approximation in function spaces to approximation in sequence spaces.

Proof for $\mathtt{H}$ (and similarly for Σ and Σ^π):

$$E_n(f, X, \mathtt{H}(\phi)) = E_n(SAf, X, S\mathtt{H}(e)) \leq \|S\| E_n(Af, E, \mathtt{H}(e)). \qquad \square$$

The last step is from the natural construction.

Approximation by Banach frames

Summarized in terms of approximation rates, the result reads as:

Corollary

For any $\alpha \in (0, \infty)$ *and* $Y \subseteq X$,

$$a^*(Y, X, \mathtt{H}(\phi)) \geq a^*(A(Y), E, \mathtt{H}(e)).$$

Remark.

- In words: for any Banach frame, the approximation rate in the function space is at least the approximation rate in the sequence space
- Equality holds when $A = S^{-1}$. For instance, this is the case for orthonormal bases, seen as Banach frames with an ℓ_2 sequence space.
- It also holds for sparse approximation Σ or Σ^π in place of $\mathtt{H}$.

Questions to answer for yourself / discuss with Friends

- Repetition: What is a Banach frame, and what is an orthonormal basis?
- Check your understanding: Are the elements of an orthonormal basis linearly independent? What about Banach frames?
- Discussion: What is better: large or small frames?
- Transfer: How could Banach frames be useful for numerical computation?
- Discussion: Have you encountered examples of orthonormal bases Banach frames elsewhere?

MH3520 Chapter 8

Part 3
Approximation in sequence spaces

Approximation in sequence spaces

Overview

- We look at sparse and non-sparse approximation in sequence spaces.
- The motivation is that any result for sequence spaces transfers to function spaces via Banach frames.
- Loosely speaking, we get weighted ℓ_p spaces as approximation spaces.
- You will see the beauty of it when you discover how the same principles are at work not only in sequence spaces but (via Banach frames) also in many well-known function spaces.

Technical detail

- So far, we have measured the approximation error in ℓ_∞, but some computations are easier in ℓ_q with finite q.
- For instance, $q = 2$ works well for approximation in ℓ_2 or, equivalently, in a Hilbert space with a chosen orthonormal basis.

Sequence spaces

Sequences are merely functions defined on $\mathbb{N}$. For example:

Definition

For any $q \in [0, \infty]$ and any measure μ on $(\mathbb{N}, 2^{\mathbb{N}})$,

$$\ell_q(\mu) := L_q(\mu).$$

If no measure is specified, it is understood to be the counting measure.

Standing assumption.

- $\alpha \in (0, \infty)$ and $q \in (0, \infty]$
- $\mu := \sum_n \delta_n / n$ is a weighted counting measure
- $e_1, e_2, e_3, \ldots$ are the unit vectors in sequence space $\mathbb{R}^{\mathbb{N}}$
- $\{0\} = H_0 \subseteq H_1 \subseteq H_2 \subseteq \ldots$ are subsets of a quasi-normed space X such that for all λ and some c: $\lambda \cdot H_n \subseteq H_n$ and $H_n + H_n \subseteq H_{cn}$.

Auxiliary inequalities

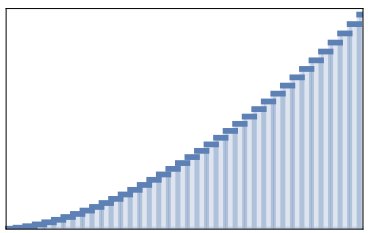
We will frequently use the following two elementary inequalities.

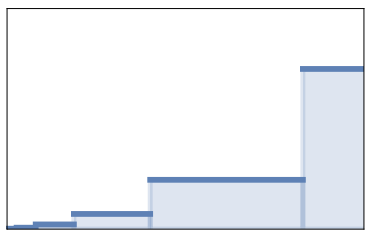
Lemma

If (a_n) is a non-negative monotonically decreasing sequence, then

$$\|(n^\alpha a_n)\|_{\ell_q} \asymp \|(2^{\alpha(n-1)} a_{2^{n-1}})\|_{\ell_q}.$$

Proof: easy, by direct comparison.





Hardy's inequality

Lemma (Hardy's inequality)

If (a_n) is a non-negative sequence and $b_n \le \|(\mathbb{1}_{\{n,n+1,\dots\}} a_n)\|_{\ell_p}$, then

$$\left\|(2^{\alpha n} b_n)\right\|_{\ell_q} \lesssim \left\|(2^{\alpha n} a_n)\right\|_{\ell_q}.$$

Proof. By monotonicity in p, it suffices to consider $p < q$. Then, there is $r > 0$ such that $p/q + p/r = 1$. Choose $\beta \in (0, \alpha)$. Then, Hölder's inequality yields

$$\begin{aligned} b_n &\le \|(\mathbb{1}_{\{k\ge n\}} 2^{\beta k} a_k 2^{-\beta k})\|_{\ell_p} \le \|(\mathbb{1}_{\{k\ge n\}} 2^{\beta k} a_k)\|_{\ell_q} \|(\mathbb{1}_{\{k\ge n\}} 2^{-\beta k})\|_{\ell_r} \\ &\lesssim \|(\mathbb{1}_{\{k\ge n\}} 2^{\beta k} a_k)\|_{\ell_q} 2^{-\beta n}. \end{aligned}$$

Therefore, by interchanging summation over n and k, one obtains

$$\begin{aligned} \left\|(2^{\alpha n} b_n)\right\|_{\ell_q} &\lesssim \left\|(2^{(\alpha-\beta)n} \|(\mathbb{1}_{\{k\ge n\}} 2^{\beta k} a_k)\|_{\ell_q})\right\|_{\ell_q} \\ &= \left\|(2^{\beta k} a_k \|(2^{(\alpha-\beta)n} \mathbb{1}_{\{k\ge n\}})\|_{\ell_q})\right\|_{\ell_q} \lesssim \left\|(2^{\beta k} a_k 2^{(\alpha-\beta)k})\right\|_{\ell_q}. \end{aligned}$$

□

Approximation spaces with summability parameter

Definition

The approximation space $A_q^\alpha(X,H)$ with approximation rate α and summability parameter q consists of all $f \in X$ such that

$$\|f\|_{A_q^\alpha(X,H)} := \|(n^\alpha E_{n-1}(f))\|_{\ell_q(\mu)} < \infty.$$

Remark. Everything is similar to before:

- Special case: $A^\alpha = A_\infty^\alpha$.
- Lexicographic ordering: $A_q^\alpha \subseteq A_r^\beta$ iff $\alpha > \beta$ or $\alpha = \beta$ and $q \le r$.
- Iteration: $A_q^\alpha(A_r^\beta(X,H),H) = A_q^{\alpha+\beta}(X,H)$.
- Interpolation: $\left(A_q^\alpha(X,H), A_r^\beta(X,H)\right)_{\theta,s} = A_s^{(1-\theta)\alpha+\theta\beta}(X,H)$.

Non-sparse approximation

Theorem

The non-sparse approximation space $A_q^\alpha(\ell_p, \mathtt{H}(e))$ *coincides with the Besov space* $\mathfrak{b}_q^\alpha(\ell_p)$ *of sequences* c *with finite quasi-norm*

$$\begin{aligned}\|c\|_{\mathfrak{b}_q^\alpha(\ell_p)} &:= \left\|(n^\alpha \|\mathbb{1}_{\{n,n+1,\dots\}} c\|_{\ell_p})\right\|_{\ell_q(\mu)} \\ &\asymp \left\|(2^{\alpha(n-1)} \|\mathbb{1}_{\{2^{n-1},\dots,2^n-1\}} c\|_{\ell_p})\right\|_{\ell_q}.\end{aligned}$$

For $p = q$, $\|c\|_{\mathfrak{b}_p^\alpha(\ell_p)} \asymp \|(n^\alpha c_n)\|_{\ell_p}$ *is equivalent to a weighted* ℓ_p *norm.*

Remark.

- All Besov spaces are defined this way: p-norm at the fine level, followed by weighted q-norm at the coarse level.
- We may use the 'standard' notation, $\mathfrak{b}_{p,q}^\alpha$
- The next optional reading section shows a famous theory related. Let ϕ be Fourier series Hypothesis in $[0,1]^d$: $A_q^\alpha(L_p([0,1]^d), \mathtt{H}(\phi)) \equiv B_q^{\alpha d}(L_p([0,1]^d))$

Non-sparse approximation: proof

Proof.

- Thanks to a) the monotonicity of the approximation error and b) Hardy's inequality,

$$\begin{aligned}
\|c\|_{A_q^\alpha(\ell_p)} &= \big\|(n^\alpha E_{n-1}(c)\|_{\ell_p})\big\|_{\ell_q(\mu)} \\
&= \big\|(n^\alpha \|\mathbb{1}_{\{n,n+1,\dots\}} c\|_{\ell_p})\big\|_{\ell_q(\mu)} \\
&\overset{\text{a)}}{\asymp} \big\|(2^{\alpha(n-1)} \|\mathbb{1}_{\{2^{n-1},2^{n-1}+1,\dots\}} c\|_{\ell_p})\big\|_{\ell_q} \\
&\overset{\text{b)}}{\asymp} \big\|(2^{\alpha(n-1)} \|\mathbb{1}_{\{2^{n-1},\dots,2^n-1\}} c\|_{\ell_p})\big\|_{\ell_q} = \|c\|_{\mathfrak{b}_q^\alpha(\ell_p)}.
\end{aligned}$$

- For $p = q$, the last line reads as

$$\begin{aligned}
\|c\|^p_{\mathfrak{b}_p^\alpha(\ell_p)} &= \sum_{n=1}^\infty \sum_{k=2^{n-1}}^{2^n-1} \left|2^{\alpha(n-1)} c_k\right|^p \\
&\asymp \sum_{n=1}^\infty \sum_{k=2^{n-1}}^{2^n-1} \left|k^\alpha c_k\right|^p = \sum_{n=1}^\infty \left|n^\alpha c_n\right|^p = \|(n^\alpha c_n)\|^p_{\ell_p}.
\end{aligned}$$

□

Non-increasing rearrangement

For sparse approximation, it is useful to reorder a given sequence c by decreasing absolute values:

Definition

The non-increasing rearrangement c^* of a sequence c is the sequence $|c|$ permuted in non-increasing sequence order.

Remark. Here is why this is important: As the best-approximation from Σ_n picks the n most significant values of a sequence,

$$E_n(c, \ell_\infty, \Sigma(e)) = c^*_{n+1}.$$

Sparse approximation

Theorem

The sparse approximation space $A_q^\alpha(\ell_p, \Sigma(e))$ *is the Lorentz space* $\ell_{r,q}$ *with* $r := 1/(\alpha + 1/p)$, *which by definition consists of all sequences* c *with*

$$\|c\|_{\ell_{r,q}} = \left\|(n^{1/r} c_n^*)\right\|_{\ell_q(\mu)} \asymp \left\|(2^{(n-1)/r} c_{2^{n-1}}^*)\right\|_{\ell_q} < \infty$$

Remark.

- For $q = r$, the Lorentz space $\ell_{r,q}$ is simply ℓ_r, and the summability parameter $r := 1/(\alpha + 1/p)$ determines the approximation rate α.
- As $\mathtt{H}_n(e) \subseteq \Sigma_n(e)$, the non-sparse approximation space is embedded in the sparse approximation space.

Take-away. For any $\alpha_- < \alpha$, by lexicographic ordering,

$$\|c\|_{\ell_{1/(\alpha_- + 1/p)}} \lesssim \|c\|_{A^\alpha(\ell_p, \Sigma(e))} \lesssim \|c\|_{\ell_{1/(\alpha+1/p)}}.$$

Sparse approximation: proof

Proof.

- As an intermediate step, one has $\ell_p = A_p^{1/p}(\ell_\infty, \Sigma(e))$ because

$$\begin{aligned}\|c\|_{\ell_p} &= \|c^*\|_{\ell_p} = \|(n^{1/p} c_n^*)\|_{\ell_p(\mu)} \\ &= \|(n^{1/p} E_{n-1}(c))\|_{\ell_p(\mu)} = \|c\|_{A_p^{1/p}(\ell_\infty, \Sigma(e))}.\end{aligned}$$

- Therefore, by the iteration theorem,

$$\begin{aligned}\|c\|_{A_q^\alpha(\ell_p, \Sigma(e))} &= \|c\|_{A_q^\alpha(A_p^{1/p}(\ell_\infty, \Sigma(e)), \Sigma(e))} \asymp \|c\|_{A_q^{\alpha+1/p}(\ell_\infty, \Sigma(e))} \\ &= \|(n^{\alpha+1/p} E_{n-1}(c))\|_{\ell_q(\mu)} = \|(n^{\alpha+1/p} c_n^*)\|_{\ell_q(\mu)} = \|c\|_{\ell_{r,q}}.\end{aligned}$$

□

Sparse approximation with polynomial search

For polynomial-depth search, one picks the n largest values among the first $\pi(n)$ values of the sequence:

Theorem

The sparse-with-polynomial-search approximation space $A^\alpha(\ell_p, \Sigma^\pi(e))$ *with* $\pi(n) := n^s$ *for some* $s \in (0, \infty)$ *contains the space of all sequences* c *with finite quasi-norm*

$$\begin{aligned}\|c\|_{A^\alpha_q(\ell_p, \Sigma^\pi(e))} &\lesssim \|c\|_{A^\alpha_q(\ell_p, \mathtt{H}_\pi(e))} + \|c\|_{A^\alpha_q(\ell_p, \Sigma(e))} \\ &= \|c\|_{\mathfrak{b}^{\alpha/s}_q(\ell_p)} + \|c\|_{\ell_{1/(\alpha+1/p),q}} < \infty.\end{aligned}$$

Proof. For any such c,

- Approximation from $\mathtt{H}_{\pi(n)}(e)$ incurs an error of order $O(n^\alpha)$, and
- For the resulting sequence, whose $\ell_{1/(\alpha+1/p),q}$ norm is at most that of c, approximation from $\Sigma_n(e)$ incurs an error of order $O(n^\alpha)$. □

Approximation rates

Summarized in terms of approximation rates, our results read as:

Corollary

The approximation rates of $F \subseteq \ell_p$, with $\pi(n) := n^s$ for some $s > 0$, are:

$$a^*(F, \ell_p, \mathtt{H}(e)) \geq \alpha \quad \text{if} \quad \sup_{c \in F} \|(n^\alpha c_n)\|_{\ell_p} < \infty,$$

$$a^*(F, \ell_p, \mathtt{\Sigma}(e)) \geq \alpha \quad \text{if} \quad \sup_{c \in F} \|c\|_{\ell_{\frac{1}{\alpha+1/p}}} < \infty,$$

$$a^*(F, \ell_p, \mathtt{\Sigma}^\pi(e)) \geq \alpha \quad \text{if} \quad \sup_{c \in F} \|(n^{\alpha/s} c_n)\|_{\ell_p} + \|c\|_{\ell_{\frac{1}{\alpha+1/p}}} < \infty.$$

- Thus, in the end, all we need are weighted ℓ_p spaces...
- Then the take-away is that high approximation rates correspond to heavy weights or stringent summability conditions.

Questions to answer for yourself / discuss with friends

- Repetition: How does 'regularity' of a sequence translate into sparse or non-sparse approximation rates?
- Check your understanding: Does the space ℓ_p increase or decrease in p? What about $L^p([0,1])$? What about $L^p(\mathbb{R})$?
- Transfer: What are the implications for approximation by Banach frames?

MH3520 Chapter 8

Part 4

Fourier approximation (Optional)

Fourier approximation

Overview

- Fourier approximation is approximation by trigonometric polynomials.
- It is a classical topic in approximation theory, where the scale of approximation space dovetails nicely with the scale of Besov space.
- Here, it serves as a motivating and introductory example.
- It is most elegantly formulated for complex-valued functions.

Main results

- We use the Fourier transform as a Banach frame, which maps function to sequences.
- In sequence space, high approximation rates correspond to heavy weights or stringent summability conditions.
- In function space, this translates to high smoothness.

Fourier basis and Fourier transform

Standing assumption.

- All function spaces are complex-valued unless mentioned otherwise.
- We fix $d \in \mathbb{N}$, $\alpha \in [0, \infty)$, $p, q \in (0, \infty]$, and a norm $\|\cdot\|$ on $\mathbb{R}^d$.

Lemma

$L^2([0,1]^d)$ has an orthonormal basis of trigonometric monomials

$$\phi_k(x) = e^{2\pi i \langle k, x \rangle}, \qquad x \in [0,1]^d, \qquad k \in \mathbb{N}_0^d,$$

hence a Banach frame. We call $\|k\|$ the degree of the monomial.

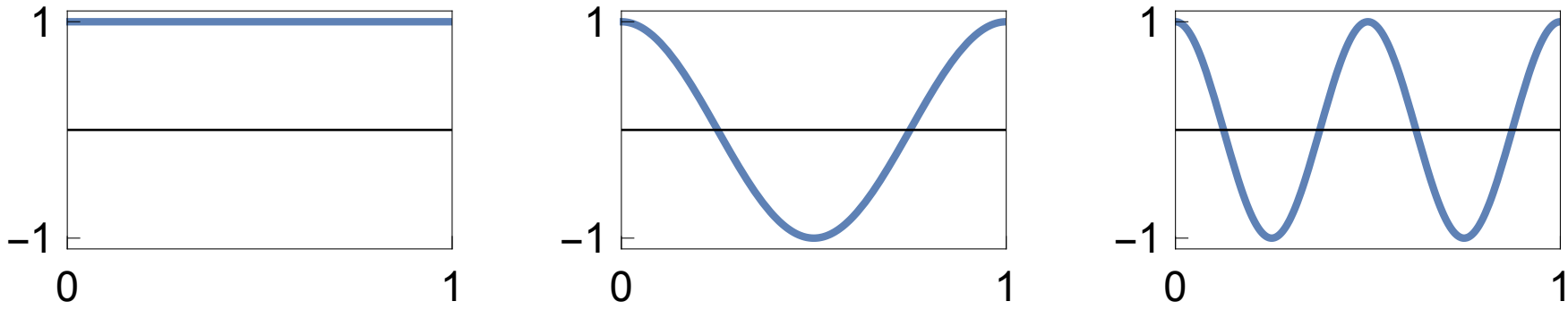
Proof. Orthogonality is easily verified by hand, and denseness of the linear span follows from Stone–Weierstrass (Banach algebra). □

Definition

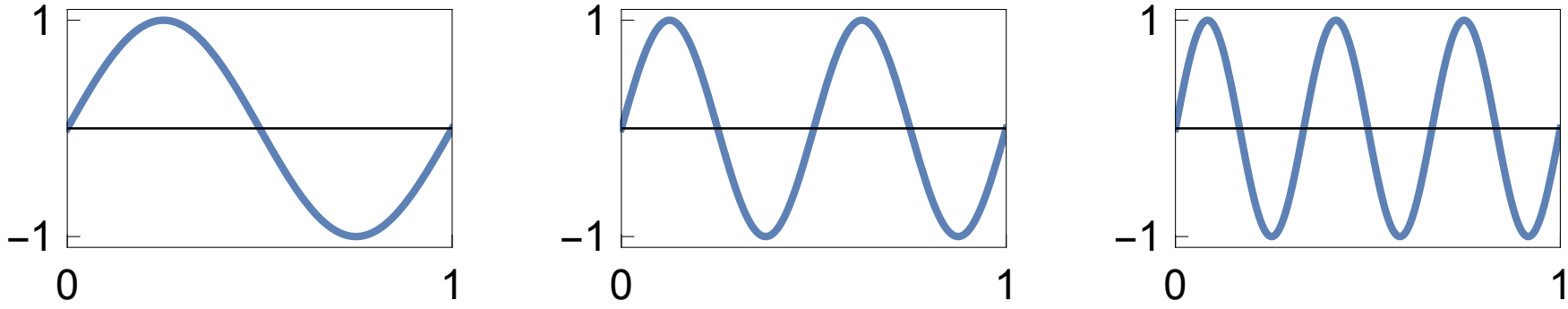
The analysis operator of the Banach frame is called Fourier transform

$$\mathcal{F} : L^2([0,1]^d) \to \ell_2(\mathbb{N}_0^d), \qquad \mathcal{F}(f)_n = \langle f, \phi_k \rangle_{L^2([0,1]^d)}.$$

Univariate trigonometric monomials



Real parts = even trigonometric monomials $x \mapsto \cos(2\pi kx)$



Imaginary parts = odd trigonometric monomials $x \mapsto \sin(2\pi kx)$

Besov spaces

Definition

The Besov space $B_q^\alpha(L_p([0,1]^d))$ consists of function $f \in L_p([0,1]^d)$ with finite quasi-norm

$$\begin{aligned}\|f\|_{B_q^\alpha(L_p([0,1]^d))} &= \left\|\left(n^\alpha \|\mathcal{F}^{-1}\mathbb{1}_{\{\|\cdot\|\geq n-1\}}\mathcal{F}f\|_{L_p([0,1]^d)}\right)\right\|_{\ell_q(\mu)} \\ &\asymp \left\|\left(2^{\alpha(n-1)}\|\mathcal{F}^{-1}\psi_{n-1}\mathcal{F}f\|_{L_p([0,1]^d)}\right)\right\|_{\ell_q},\end{aligned}$$

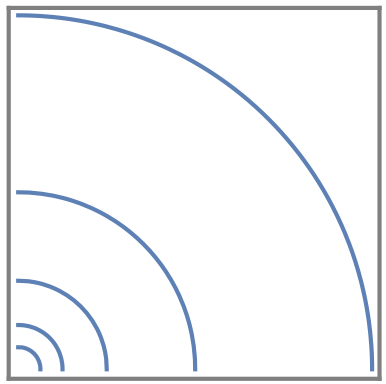
where $\mathcal{F}$ is the Fourier transform, and $\psi_n : \mathbb{N}_0^d \to \{0,1\}$ form a circular/annular partition of unity (see next slide).

Remark. All Besov norms are defined this way: p-norm at the fine level, weighted q-norm at the coarse level.

Tiling of the frequency domain

- Besov spaces use a circular/annular tiling of the frequency domain:

$$1 = \sum_{n=0}^{\infty} \psi_n,$$
$$\psi_0 = \mathbb{1}_{\{k:\, \|k\|<1\}},$$
$$\psi_n = \mathbb{1}_{\{k:\, 2^{n-1}\leq\|k\|<2^n\}}.$$



- For the 1-norm, $\|k\| = k_1 + \cdots + k_d$ is the total degree .
- For the ∞-norm, $\|k\| = \max\{k_1, \ldots, k_d\}$ is the maximal degree.
- Any choice is ok because all norms in finite dimension are equivalent.

Fourier approximation

Theorem

The approximation space $A_q^\alpha(L_p([0,1]^d), \mathtt{H}(\phi))$ *coincides with the Besov space* $B_q^{\alpha d}(L_p([0,1]^d))$.

Remark.

- Amazingly, Besov regularity coincides with approximation capability.
- There is a curse of dimensionality (i.e., adverse dependence on d).
- One obtains a real-valued version if ϕ_n's are replaced by cosines (or sines).

Fourier approximation: proof

Proof. Set $\mu := \sum_n \delta_n/n$.

- As approximation in $X := L_p([0,1])$ by $\mathtt{H}_n(\phi)$ sets the first n frequencies to zero, and frequencies are counted starting from zero,

$$E_n(f) = \|\mathcal{F}^{-1}\mathbb{1}_{\{n,n+1,\dots\}}\mathcal{F}f\|_X, \quad \text{and}$$
$$E_{2^{d(n-1)}}(f) = \|\mathcal{F}^{-1}(\psi_n + \psi_{n+1} + \dots)\mathcal{F}f\|_X.$$

- By a) the monotonicity of $E_n(f)$ and b) Hardy's inequality,

$$\begin{aligned}
\|f\|_{A_q^\alpha(X,\mathtt{H}(\phi))} &\overset{\text{def}}{=} \left\|(n^\alpha E_{n-1}(f))\right\|_{\ell_q(\mu)} \asymp \|f\|_X + \left\|(n^\alpha E_n(f))\right\|_{\ell_q(\mu)} \\
&\overset{\text{a)}}{\asymp} \|f\|_X + \left\|(2^{\alpha d(n-1)} E_{2^{d(n-1)}}(f))\right\|_{\ell_q(\mu)} \\
&= \|f\|_X + \left\|(2^{\alpha d(n-1)} \|\mathcal{F}^{-1}(\psi_n + \psi_{n+1} + \dots)\mathcal{F}f\|_X\right\|_{\ell_q} \\
&\asymp \left\|(2^{\alpha d(n-1)} \|\mathcal{F}^{-1}(\psi_{n-1} + \psi_n + \dots)\mathcal{F}f\|_X\right\|_{\ell_q} \\
&\overset{\text{b)}}{\asymp} \left\|(2^{\alpha d(n-1)} \|\mathcal{F}^{-1}\psi_{n-1}\mathcal{F}f\|_X\right\|_{\ell_q} = \|f\|_{B_q^{\alpha d}(X)}.
\end{aligned}$$

□

Besov spaces with $p = q$

For $p = q$ we get an L_p space with a weight applied in the Fourier domain:

Lemma

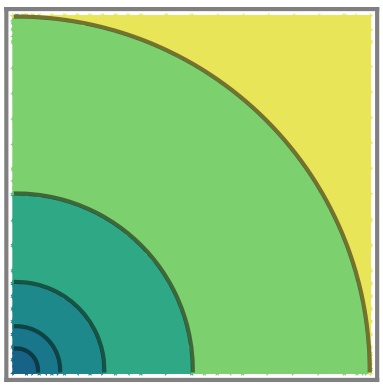
For $p = q$, one has $\|f\|_{B_p^\alpha(L_p([0,1]^d))} \asymp \|\mathcal{F}^{-1}(1 + \|\cdot\|^\alpha)\mathcal{F}f\|_{L_p([0,1]^d)}$.

Proof of the lemma.

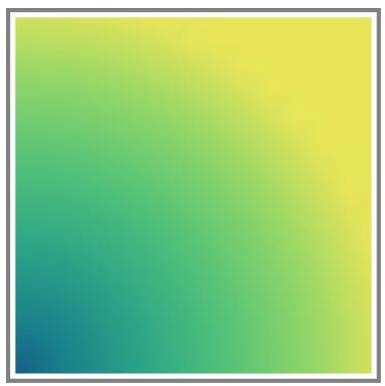
- Swap ℓ_p summation and L_p integration to get

$$\|f\|_{B_q^\alpha(L_p([0,1]^d))} = \left\|\mathcal{F}^{-1}\|2^{\alpha(n-1)}\psi_{n-1}\|_{\ell_p}\mathcal{F}f\right\|_{L_p([0,1]^d)}.$$

- Use $2^{\alpha(n-1)}\psi_{n-1}(k) \asymp (1 + \|k\|^\alpha)\psi_{n-1}(k)$ and $\sum_n \psi_n = 1$. □



Weight $\sum_n 2^{\alpha(n-1)}\psi_{n-1}(k)$



Weight $1 + \|k\|^\alpha$

Relations to other function spaces

Besov spaces were invented to create order in the messy zoo of functions.

Theorem

The following embeddings and isomorphisms hold:

- $C^\alpha([0,1]^d) \subseteq B^\alpha_\infty(L^\infty([0,1]^d))$ *with equality for* $\alpha \notin \mathbb{N}$.
- $L_p([0,1]^d) = B^0_p(L_p([0,1]^d))$.
- $H^\alpha([0,1]^d) = W^{\alpha,2}([0,1]^d) = B^\alpha_2(L^2([0,1]^d))$.

Proof for $H^\alpha([0,1]) = B^\alpha_2(L^2([0,1]))$ and $\alpha \in \mathbb{N}$ only:As the Fourier transform is an isometry $L^2([0,1]) \to \ell_2$ and $(\mathcal{F}f')_k = 2\pi i k(\mathcal{F}f)_k$,

$$\begin{aligned}
\|f\|^2_{B^\alpha_2(L^2([0,1]))} &= \|\mathcal{F}^{-1}(1+\|\cdot\|^\alpha)\mathcal{F}f\|^2_{L^2([0,1])} \\
&= \|(1+\|\cdot\|^\alpha)\mathcal{F}f\|^2_{\ell_2} \asymp \|\mathcal{F}f\|^2_{\ell_2} + \|(k^\alpha(\mathcal{F}f)_k)\|^2_{\ell_2} \\
&\asymp \|f\|^2_{L^2([0,1])} + \|\, d^\alpha f\|^2_{L^2([0,1])} = \|f\|^2_{H^\alpha([0,1])}. \qquad \square
\end{aligned}$$

Approximation rates

Summarized in terms of approximation rates, our result reads as:

Corollary

The approximation rate of F by trigonometric monomials satisfies

$$a^*(Y, L^2([0,1]^d), \mathtt{H}(\phi)) \geq \alpha \qquad \textit{if} \qquad \sup_{f \in Y} \|f\|_{H^{\alpha d}([0,1]^d)} < \infty,$$

$$a^*(Y, L^\infty([0,1]^d), \mathtt{H}(\phi)) \geq \alpha \qquad \textit{if} \qquad \sup_{f \in Y} \|f\|_{C^{\alpha d}([0,1]^d)} < \infty,$$

Remark.

- This relates Sobolev or Hölder smoothness to the approximation rate.
- Little is known for approximations from $\Sigma(\phi)$.

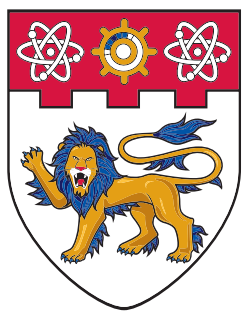
Questions to answer for yourself / discuss with friends

- Repetition: At what rate can Hölder C^α or Sobolev H^α functions be approximated by trigonometric polynomials?
- Check your understanding: Are Besov spaces $B^\alpha_q(L_p)$ ordered lexicographically?
- Check your understanding: How bad is the curse of dimensionality: if n trigonometric polynomials are sufficient in dimension 1, do you need dn, n^d or d^n trigonometric polynomials in dimension d?
- Discussion: Can you give an example of a function which can/cannot be approximated by trigonometric polynomials at rate, say, 3?

MH3520:Mathematics of Deep Learning

Chapter 9

Splines and approximation by wide networks



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

Last chapter:

- Approximation theory: quantitative control over the approximation error

This chapter:

- Approximation theory for neural networks
- Application: neural networks are at least as good as splines

Next chapter:

- Approximation theory for deep neural networks

Overview

Size of neural networks:

- We will quantify the size of a network by counting the number of neurons, weights, etc.
- This is well defined for neural networks but not for perceptrons because of over-parameterization

Dictionary learning:

- This is a general transfer-of-approximation result, which says:
- If perceptrons approximate a dictionary well, and the dictionary approximates a function class well, then perceptrons approximate the function class well.

Spline approximation:

- Splines are known to approximate a whole scale of functions well.
- We already know them—it's the scale of Besov functions.
- The implication is: anything splines can do, networks can do as well.

Overview of Chapter 9

1. Spaces of neural networks
2. Approximation spaces of multi-layer perceptrons
3. Dictionary learning
4. Spline approximation
5. Dictionary learning with splines

Sources for this chapter:

- Petersen (2022): Neural Network Theory. Lecture Notes, University of Vienna. `pc-petersen.eu/Neural_Network_Theory.pdf`
- Gribonval Kutyniok Nielsen Voigtlaender (2022): Approximation spaces of deep neural networks. Constructive approximation 55:1, 259–367.

MH3520 Chapter 9

Part 1

Spaces of neural networks

Definition of neural networks

Definition

A neural network with L layers is a family of matrix-vector tuples

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

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

Remark. According to this definition, neural networks are the coefficients of multi-layer perceptrons. Let's make this precise...

Realizations of neural networks

Definition

The realization of a neural network

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

with activation function $\rho\colon \mathbb{R} \to \mathbb{R}$ is the multi-layer perceptron

$$\mathrm{R}(\Phi) = T_L \circ \rho \circ T_{L-1} \circ \cdots \circ \rho \circ T_1,$$

where ρ is understood to act component-wise, and where

$$\forall l \in \{1, \ldots, L\}: \qquad T_l : \mathbb{R}^{N_{l-1}} \to \mathbb{R}^{N_l}, \qquad T_l(x) = A_l x + b_l.$$

Remark.

- Every multilayer perceptron is the realization of a neural network.
- However, several neural networks might have the same realization.
- This is called over-parameterization.

Size of a neural network

Let $\|\cdot\|_0$ be the number of non-zero entries of a matrix or vector.

Definition

For any neural network Φ, in the above notation,

- The depth or number of layers is L,
- The architecture is $S := (N_L, \ldots, N_0)$ and width $\max\{N_L, \ldots, N_0\}$,
- The input dimension is N_0, and the output dimension is N_L,
- The number of neurons is $N_L + \cdots + N_0$
- The number of hidden neurons is $N := N_{L-1} + \cdots + N_1$,
- The number of weights is $W := \|A_L\|_0 + \cdots + \|A_1\|_0$
- The number of biases is $B := \|b_L\|_0 + \cdots + \|b_1\|_0$
- The number of coefficients is $C := W + B$
- The size of the coefficients is the supremum K of $|A_{l,i,j}|, |b_{l,i}|$

These notions are ill defined for perceptrons due to over-parameterization.

Parallelization

We'll next discuss several operations on neural networks.

Definition

The parallelization of $f : \mathbb{R}^d \to \mathbb{R}^m$ and $g : \mathbb{R}^d \to \mathbb{R}^n$ is the function

$$(f, g) : \mathbb{R}^d \to \mathbb{R}^{m+n}, \qquad (f, g)(x) = (f(x), g(x)).$$

Remark. In parallelization, the inputs of f and g are shared. When they are not, one speaks of full parallelization $(x, y) \mapsto (f(x), g(y))$.

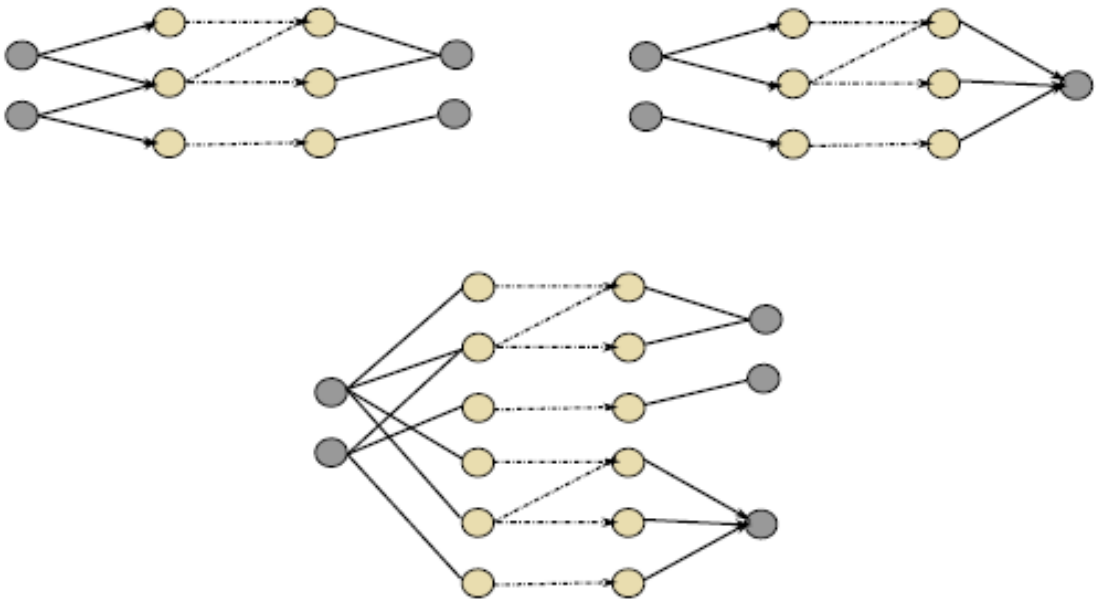
Lemma

For any neural networks Φ_1, Φ_2 with depth L and input dimension d, there exists a neural network $P(\Phi_1, \Phi_2)$ such that

$$R(P(\Phi_1, \Phi_2)) = (R(\Phi_1), R(\Phi_2)).$$

Moreover, $P(\Phi_1, \Phi_2)$ can be chosen to have depth L, input dimension d, $N(P(\Phi_1, \Phi_2)) \leq \sum N(\Phi_i)$, and $W(P(\Phi_1, \Phi_2)) \leq \sum_W (\Phi_i)$.

Parallelization: visual proof



Two networks (top) and their parallelisation (bottom) [Petersen]

Parallelisation: proof

Proof.

- The networks Φ_1 and Φ_2 are of the form

$$\Phi^1 = \left((A_L^1, b_L^1), \ldots, (A_1^1, b_1^1)\right), \qquad \Phi^2 = \left((A_L^2, b_L^2), \ldots, (A_1^2, b_1^2)\right).$$

- Their full parallelization (with unshared inputs) is the neural network

$$FP(\Phi^1, \Phi^2) := \left(\left(\begin{pmatrix} A_L^1 & 0 \\ 0 & A_L^2 \end{pmatrix}, \begin{pmatrix} b_L^1 \\ b_L^2 \end{pmatrix} \right), \ldots, \left(\begin{pmatrix} A_1^1 & 0 \\ 0 & A_1^2 \end{pmatrix}, \begin{pmatrix} b_1^1 \\ b_1^2 \end{pmatrix} \right) \right).$$

- Their parallelization has a modified bottom-most layer:

$$P(\Phi^1, \Phi^2) := \left(\left(\begin{pmatrix} A_L^1 & 0 \\ 0 & A_L^2 \end{pmatrix}, \begin{pmatrix} b_L^1 \\ b_L^2 \end{pmatrix} \right), \ldots, \left(\begin{pmatrix} A_1^1 \\ A_1^2 \end{pmatrix}, \begin{pmatrix} b_1^1 \\ b_1^2 \end{pmatrix} \right) \right).$$

□

Composition

Definition

The composition of $f : \mathbb{R}^d \to \mathbb{R}^k$ and $g : \mathbb{R}^k \to \mathbb{R}^l$ is the function

$$g \circ f : \mathbb{R}^d \to \mathbb{R}^l, \qquad g \circ f(x) = g(f(x)).$$

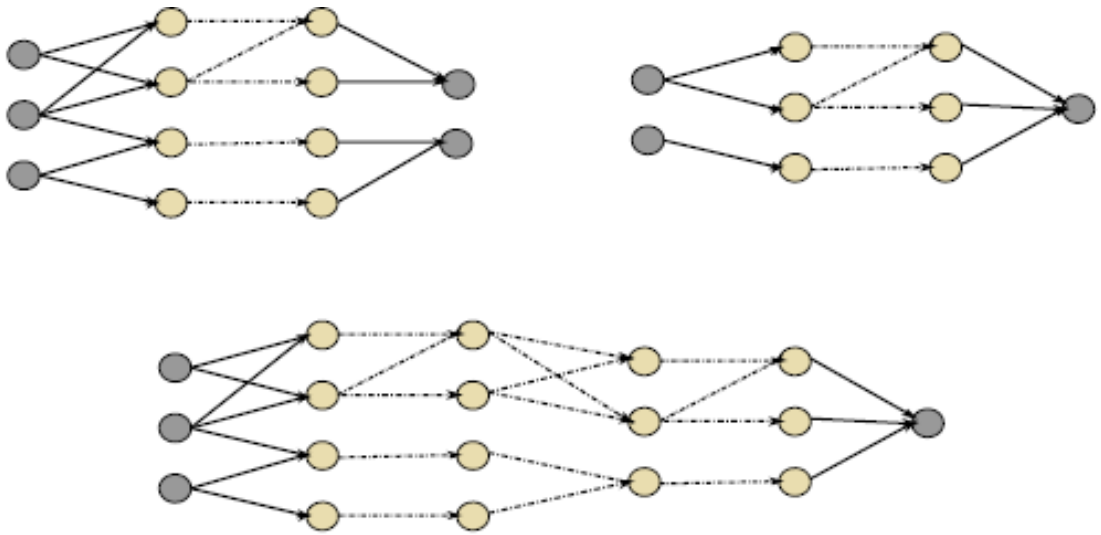
Lemma

For any neural networks Φ_2 and Φ_1 with input dimension of Φ_2 equal to output dimension of Φ_1, there exists a neural network $\Phi_2 \bullet \Phi_1$ such that

$$R(\Phi_2 \bullet \Phi_1) = R(\Phi_2) \circ R(\Phi_2).$$

Moreover, $\Phi_2 \bullet \Phi_1$ can be chosen s.t. $L(\Phi_2 \bullet \Phi_1) = L(\Phi_2) + L(\Phi_1) - 1$, $N(\Phi_2 \bullet \Phi_1) \leq N(\Phi_2) + N(\Phi_1)$, and
$W(\Phi_2 \bullet \Phi_1) \leq W(\Phi_1) + \max\{N(\Phi_1), d\}W(\Phi_2)$.

Composition: visual proof



Two neural networks (top) and their composition (bottom) [Petersen]

Composition: proof

Proof.

- The networks Φ_2 and Φ_1 are of the form

$$\Phi^2 = \big((A^2_{L_2}, b^2_{L_2}), \ldots, (A^2_1, b^2_1)\big)\,, \qquad \Phi^1 = \big((A^1_{L_1}, b^1_{L_1}), \ldots, (A^1_1, b^1_1)\big)\,.$$

- Define $\Phi^2 \bullet \Phi^1$ as the following network with $L_1 + L_2 - 1$ layers:

$$\big((A^2_{L_2}, b^2_{L_2}), \ldots, (A^2_2, b^2_2), (A^2_1 A^1_{L_1}, A^2_1 b^1_{L_1} + b^2_1), (A^1_{L_1-1}, b^1_{L^1-1}), \ldots, (A^1_1, b^1_1)\big).$$

- Distinguishing the cases $L(\Phi_1) = 1$ versus $L(\Phi_1) > 1$ leads to

$$\|A^2_1 A^1_L\|_0 \le \|A^2_1\|_0 \max_i \|e_i^\top A^1_L\|_0 \le \|A^2_1\|_0 \max\{N(\Phi_1), d\}. \qquad \square$$

Affine transformations

Corollary

Let Φ be a neural network with input dimension d and output dimension k. Then, for any affine maps $P = (A, b) : \mathbb{R}^{d_1} \to \mathbb{R}^d$ and $Q = (B, c) : \mathbb{R}^k \to \mathbb{R}^{k_1}$, there exists a neural network Ψ such that

$$R(\Psi) = Q \circ R(\Phi) \circ P.$$

Moreover, Ψ can be chosen such that $L(\Psi) = L(\Phi)$, $N(\Psi) \leq N(\Phi)$, and $W(\Psi) \leq \max_j \|Be_j\|_0 \cdot W(\Phi) \cdot \max_i \|e_i^\top A\|_0$.

Proof:

- Define Ψ as

$$((B, c)) \bullet \Phi \bullet ((A, b)).$$

- Use the estimates

$$\|BA\|_0 \leq \max_j \|Be_j\|_0 \|A\|_0, \qquad \|BA\|_0 \leq \|B\|_0 \max_i \|e_i^\top A\|_0.$$

Linear combinations

Corollary

For any networks $\Phi_1, \ldots, \Phi_n$ with equal depths, input dimensions, and output dimensions, and for any $c_1, \ldots, c_n \in \mathbb{R}$, there exists a neural network Ψ such that

$$R(\Psi) = \sum_i c_i R(\Phi_i).$$

Moreover, Ψ can be chosen such that $L(\Psi) = L(\Phi_i)$, $N(\Psi) \leq \sum_i N(\Phi_i)$, and $W(\Psi) \leq \sum_i W(\Phi_i)$.

Proof:

- Let k be the output dimension of Φ_i and define

$$A = \begin{pmatrix} c_1 \operatorname{Id}_k & \ldots & c_n \operatorname{Id}_k \end{pmatrix} \in \mathbb{R}^{k \times (kn)}, \qquad b = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \in \mathbb{R}^k.$$

- Define the neural network Φ by

$$\Phi = ((A, b)) \bullet P(\Phi_1, \ldots, \Phi_n).$$

□

Questions to Answer for Yourself / Discuss with Friends

- Repetition: What are neural networks, and how do they differ from multi-layer perceptrons?
- Check your understanding: Is $\|\cdot\|_0$ a metric?
- Discussion: How could one define the width, depth, number of neurons, etc. of a multi-layer perceptron?
- Discussion: Can you think of any further operations on neural networks?
- Transfer: What measure of a network's complexity is related to the number of floating point operations needed to compute the output given the input?

MH3520 Chapter 9

Part 2

Approximation spaces of multi-layer perceptrons

Hypothesis classes of multi-layer perceptrons

Standing assumption.

- X is a quasi-normed space of functions $\mathbb{R}^d \supseteq \Omega \to \mathbb{R}^k$
- Activation function ρ, number of layers L

Definition

$\mathtt{W}_n(X, \rho, L)$ denotes the elements of the function space X which are realizations of neural networks Φ with activation function ρ, at most n non-zero weights, and exactly L layers.

Remark. The letter $\mathtt{W}$ relates to weights.

Approximation spaces of multi-layer perceptrons

Lemma

The hypothesis classes $\mathtt{W}_n(X, \rho, L)$ *give rise to quasi-normed approximation spaces*

$$A_q^\alpha(X, \mathtt{W}(X, \rho, L)).$$

Proof. The hypothesis classes satisfy

$$\lambda \mathtt{W}_n(X, \rho, L) \subseteq \mathtt{W}_n(X, \rho, L),$$
$$\mathtt{W}_n(X, \rho, L) + \mathtt{W}_n(X, \rho, L) \subseteq \mathtt{W}_{2n}(X, \rho, L).$$

□

Number of weights versus number of coefficients

Coefficients are either weights or biases, i.e., $C = W + B$. Counting weights is equivalent to counting coefficients:

Theorem

If $\mathtt{C}_n(X, \rho, L)$ denotes multi-layer perceptrons with at most n coefficients (including both weights and biases), then

$$A_q^\alpha(X, \mathtt{C}(X, \rho, L)) = A_q^\alpha(X, \mathtt{W}(X, \rho, L)).$$

Proof. This is a consequence of the following lemma. □

Lemma

For any network Φ with output dimension k, there exists a compressed network Ψ with the same realization and number of layers such that

$$N(\Psi) \leq N(\Phi), \qquad W(\Psi) \leq W(\Psi) + B(\Psi) \leq k + 2W(\Phi).$$

Number of weights versus number of coefficients: proof

Sketch of proof.

- Assume for contradiction that the claim fails for some network Φ, and assume Φ to have a minimal number of neurons.
- If every non-input neuron of Φ is connected to at least one neuron in the layer below, then $\|\!|b_l|\!\|_0 \leq N_l \leq \|\!|A_l|\!\|_0$. Thus, $B(\Phi) \leq W(\Phi)$, and the claim holds true for $\Psi = \Phi$, a contradiction.
- Thus, at least one neuron of Φ is not connected to any layer underneath. Intuitively, one can remove it without changing $R(\Phi)$, a contradiction to the minimality of Φ's number of neurons.
- When removing the neuron, some care is needed to distinguish between hidden neurons versus output neurons and between layers of size 1 versus size greater than 1. □

Number of weights versus number of neurons

Counting weights is inequivalent to counting hidden neurons for $L > 2$, i.e., unless there is only a single hidden layer:

Theorem

Let $\mathtt{N}_n(X, \rho, L)$ denote multi-layer perceptrons with at most n hidden neurons. Then, there are continuous embeddings

$$A_q^\alpha(X, \mathtt{W}(X, \rho, L)) \subseteq A_q^\alpha(X, \mathtt{N}(X, \rho, L)) \subseteq A_q^{\alpha/2}(X, \mathtt{W}(X, \rho, L)).$$

If $L = 2$, then $\alpha/2$ can be replaced by α.

Proof.

- Let d be the input dimension and k the output dimension.
- Counting weights in comparison to hidden neurons reveals that

$$\mathtt{W}_n(X, \rho, L) \subseteq \mathtt{N}_n(X, \rho, L) \subseteq \mathtt{W}_{n^2+(d+k)n}(X, \rho, L).$$

- For layer $L = 2$, the term n^2 on the right-hand side can be omitted.

□

Questions to Answer for Yourself / Discuss with Friends

- Repetition: How are approximation spaces of neural networks defined?
- Check your understanding: Is the set of neural networks with fixed architecture a vector space? What about the realizations of these networks?
- Check your understanding: Is the set of neural network with fixed number of weights a vector space? What about the realizations of these networks?
- Discussion: We have been counting weights, coefficients, and hidden neurons. Can you think of other interesting measures of complexity?

MH3520 Chapter 9

Part 3

Dictionary learning

Dictionary learning

Standing assumptions.

- X is a quasi-normed space of functions $\mathbb{R}^d \supseteq \Omega \to \mathbb{R}^k$
- $\phi = (\phi_\lambda)_{\lambda \in \Lambda}$ is a dictionary in X.

Definition

For sparse approximation from a dictionary $(\phi_\lambda)_{\lambda \in \Lambda}$, one uses the sets $H_n = \Sigma_n(\phi)$ of n-term linear combinations of dictionary items:

$$\Sigma_n(\phi) = \left\{ \sum_{\lambda \in \Lambda} c_\lambda \phi_\lambda : c_\lambda \neq 0 \text{ at most } n \text{ times} \right\}.$$

Theorem

Assume for some fixed depth L and number of weights W that

$$\forall \lambda \in \Lambda : \qquad \phi_\lambda \in \overline{\mathtt{W}_W(X, \rho, L)}.$$

Then, for any $\alpha \in (0, \infty)$ and $q \in (0, \infty]$,

$$A_q^\alpha(X, \Sigma(\phi)) \subseteq A_q^\alpha(X, \mathtt{W}(X, \rho, L)).$$

Dictionary learning: discussion

$$A_q^\alpha(X, \Sigma(\phi)) \subseteq A_q^\alpha(X, \mathtt{W}(X, \rho, L)).$$

Assumptions.

- $\mathtt{W}_W(X, \rho, L)$ are networks with fixed depth and number of weights.
- The assumption that ϕ_λ can be approximated by such networks is quite strong but is satisfied for certain basis splines (and wavelets).

Conclusions.

- Any approximation rate for the dictionary yields an approximation rate for the networks.
- In this result, the hypothesis classes are n-term linear combinations of dictionary items and realizations of networks with at most n weights.

In words: 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.

Dictionary learning: proof

Proof.

- Consider $c_i \in \mathbb{R}$ and $\lambda_i \in \Lambda$. By assumption, $\phi_i \in \overline{\mathtt{W}_W(X, \rho, L)}$.
- As scalar multiplication is a continuous operation in X and does not increase the number of weights, $c_i\phi_i \in \overline{\mathtt{W}_W(X, \rho, L)}$.
- As addition is a continuous operation in X, which increases the number of weights at most n-fold, $c_1\phi_1 + \cdots + c_n\phi_n \in \overline{\mathtt{W}_{nW}(X, \rho, L)}$.
- Thus, we have shown that $\Sigma_n(\phi) \subseteq \overline{\mathtt{W}_{nW}(X, \rho, L)}$.
- Consequently, viewing n as a free (dummy) variable, one has

$$\begin{aligned} A_q^\alpha(X, \Sigma_n(\phi)) &\subseteq A_q^\alpha(X, \overline{\mathtt{W}_{nW}(X, \rho, L)}) \\ &= A_q^\alpha(X, \mathtt{W}_{nW}(X, \rho, L)) \\ &= A_q^\alpha(X, \mathtt{W}_n(X, \rho, L)). \end{aligned}$$

□

Questions to answer for yourself / discuss with friends

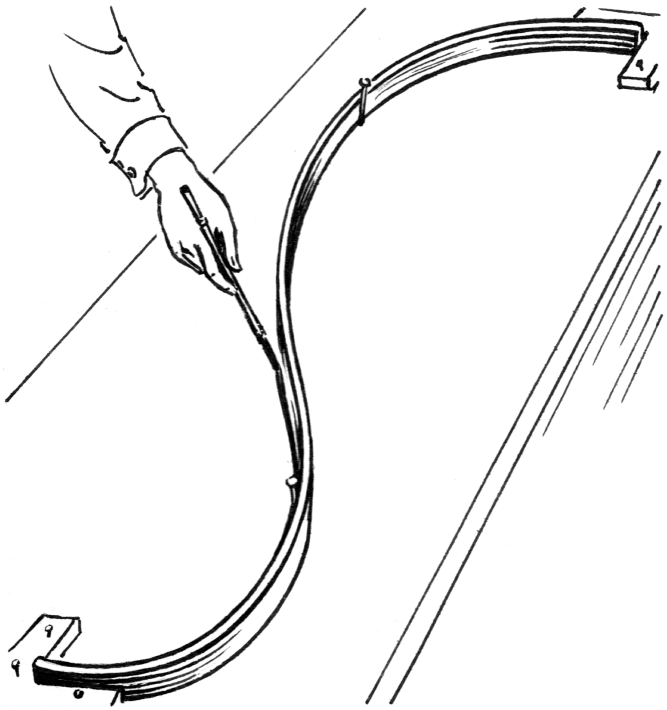
- Repetition: What are the assumptions and conclusions of the dictionary learning theorem?
- Check your understanding: Would dictionary learning work with non-sparse instead of sparse dictionary approximation?
- Check your understanding: Would dictionary learning work if $\phi_\lambda \in A^\beta(X, \mathbb{W}(X, \rho, L))$ for some β?
- Discussion: Do you think dictionary learning happens in reality when training neural networks?

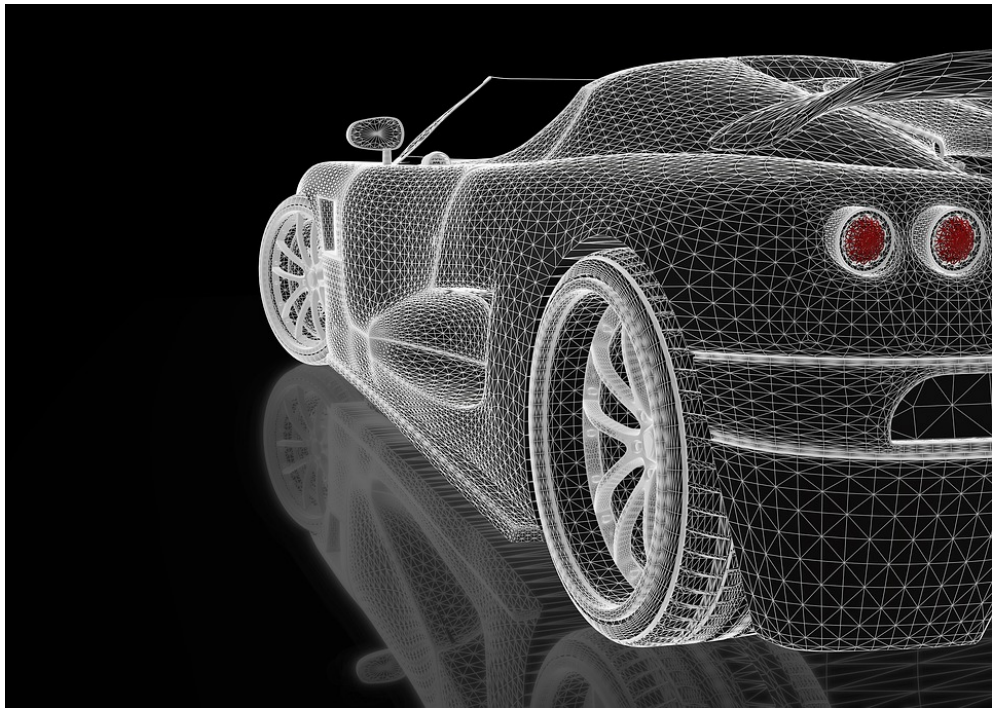
MH3520 Chapter 9

Part 4

Spline approximation

History and applications of splines





Origins in boat building, standard in computer graphics and industrial design.

Univariate basis splines

A spline is a piecewise polynomial. Univariate means in one variable.

Definition

- The cardinal basis spline of degree $s-1 \in \mathbb{N}$ is defined as

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

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

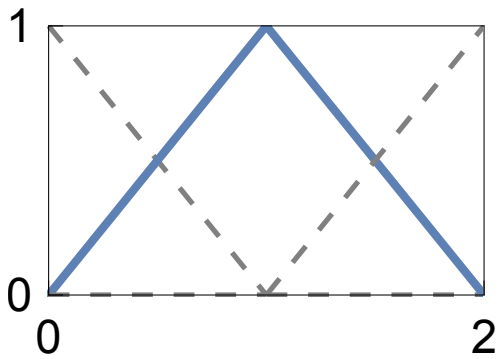
- Basis splines are defined by dilations and translations:

$$\mathcal{N}_{a,b,s}(x) := \mathcal{N}_k\big(asx - b\big), \qquad a \in \mathbb{N}, b \in \mathbb{Z}, x \in \mathbb{R}.$$

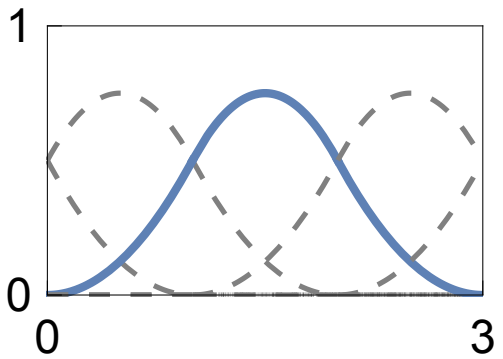
- Splines are linear combinations of basis splines.

Remark. The cardinal basis spline $\mathcal{N}_s$ is the s-fold convolution $\mathbb{1}_{[0,1)} \star \cdots \star \mathbb{1}_{[0,1)}$ of the indicator function of the unit interval.

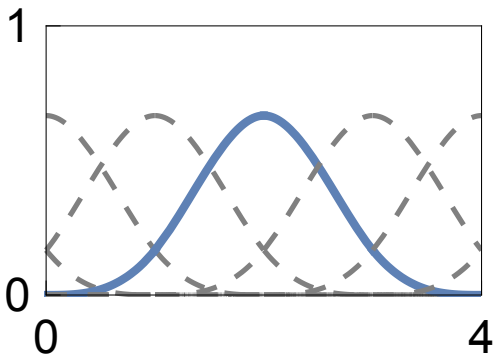
Univariate basis splines



s = 2

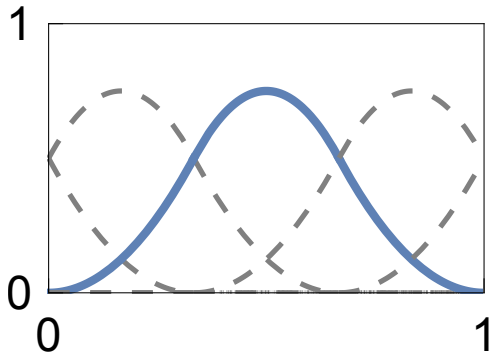


s = 3

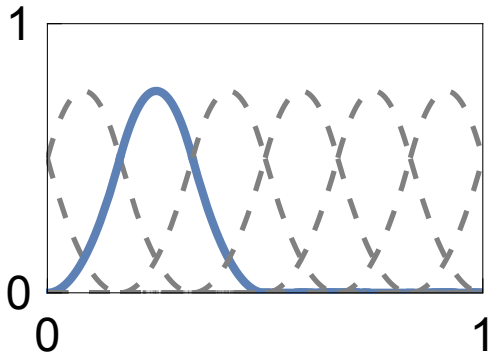


s = 4

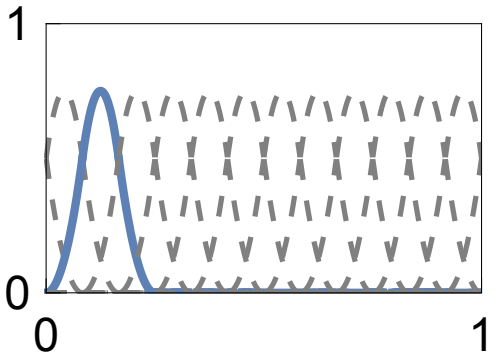
The cardinal basis spline $\mathcal{N}_s$ (blue) and some of its translates (gray).



a = 1



a = 2



a = 4

Basis splines of degree 2 for different dilations a.

Multivariate basis splines

- A multi-variate function is a function of several variables.
- Multi-variate splines are defined as products of univariate splines.

Definition

The multivariate basis splines of degree $s-1 \in \mathbb{N}$ in $d \in \mathbb{N}$ variables are

$$\mathcal{N}^d_{a,b,s}(x) = \prod_{i=1}^{d} \mathcal{N}_{a,b_i,s}(x_i), \qquad x \in \mathbb{R}^d, a \in \mathbb{N}, b \in \mathbb{Z}^d.$$

Multi-variate splines are linear combinations thereof.

Remark. To be precise, $s-1$ is the coordinate degree of the spline, i.e., the maximal power of each coordinate x_i. The total degree is higher.

Dyadic splines

To get nested hypothesis classes of splines, we use dyadic grids:

Definition

The dyadic basis splines of degree $s-1$ on $[0,1]^d$ are the functions

$$\mathcal{N}^d_{2^a,b,s} : [0,1]^d \to \mathbb{R}, \qquad a \in \mathbb{N}_0, \quad b \in \{1-s, \dots, 2^a s - 1\}^d.$$

The dictionary of dyadic basis splines is the enumeration of dyadic basis splines ordered lexicographically, i.e., first by a and then by b.

Remark. The range of indices a and b is chosen such that the dyadic basis splines, seen as functions on $[0,1]^d$, are linearly independent.

Spline approximation

Assumption.

- ϕ is the dictionary of dyadic basis splines of degree $s-1$ on $[0,1]^d$, in lexicographic order.
- $\alpha d \in (0, s)$ and $p, q \in (0, \infty]$.

Theorem

There is a continuous embedding

$$B_q^{\alpha d}(L_p([0,1]^d)) \subseteq A_q^{\alpha}(L_p([0,1]^d), \mathtt{H}(\phi))$$

which is an isomorphism if $\alpha d < s - 1 + \min\{1, p^{-1}\}$.

Take-away. Any function in the Hölder space $C^{\alpha d}$ or Sobolev space $H^{\alpha d}$ has spline best-approximation rate α or higher, if the spline degree is chosen sufficiently high.

Spline approximation: discussion

Conclusion:

- The spline degree does not influence the approximation rate, as long as it is sufficiently high.
- There is a curse of dimensionality: if f is C^α, then the approximation rate α/d decreases in d.

Intuition for $d = 1$:

- If $f \in C^\alpha([0,1])$ with $\alpha \in (0,1]$, then the first-order modulus of smoothness of f satisfies the bound $|f(y) - f(x)| \lesssim |y - x|^\alpha$.
- Thus, on a grid of step size $1/n$, f can be approximated by piecewise constant functions at rate $n^{-\alpha}$.
- For $f \in C^{k+\alpha}([0,1])$, one approximates the k-th derivative $f^{(k)}$ by piecewise constant functions and integrates k times.

Spline approximation: discussion

Conditions:

- The condition $\alpha d < s$ makes sure that the $B_q^{\alpha d}$ norm can be expressed in terms of the modulus of smoothness of order s.
- The stronger condition $\alpha d < s - 1 + \min\{1, p^{-1}\}$ is needed only for the inverse estimate, and only for dyadic grids.
- For uniform grids, the condition is not needed because problems at dyadic recurring grid points are averaged out, but then the hypothesis classes are not nested.

Implementation:

- In practice, one constructs a near-best approximation, which is optimal up to a multiplicative constant.
- One such near-best approximation is an interpolatory spline, which can be obtained by solving a linear equation, whose coefficients are the values of $f, \ldots, f^{(k)}$ at some interior points of the grid.

Questions to answer for yourself / discuss with friends

- Repetition: What are splines, basis splines, and cardinal basis splines?
- Check your understanding: At what rate can Hölder C^α or Sobolev H^α functions be approximated by splines?
- Check your understanding: What is the highest approximation rate that can be achieved via the theorem for piecewise linear splines?
- Transfer: What is spline interpolation? Does it differ from spline approximation?

MH3520 Chapter 9

Part 5

Dictionary learning with splines

Higher-order sigmoidal activation functions

Definition

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

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

Example

- Sigmoidal functions are sigmoidal of order 0.
- The ReLu function $\rho_1(x) = (x)_+$ is sigmoidal of order 1.
- The power unit $\rho_r(x) = (x)_+^r$ is sigmoidal of order $r \in \mathbb{N}$.
- Smoothing the r-th power unit near $x = 0$ results in a sigmoidal function of order r.

Dictionary learning with splines

Standing assumptions.

- $d \in \mathbb{N}$, $p, q \in (0, \infty]$, $X = L^p([0,1]^d)$, and $\alpha \in (0, \infty)$.
- ρ is sigmoidal of order $r \in \mathbb{N}$. Either $r = d = 1$ and $\alpha < 2$ or $r \geq 2$.

Theorem

There exists $L \in \mathbb{N}$ such that there is a continuous embedding

$$B_q^{\alpha d}(X) \subseteq A_q^{\alpha}(X, \mathtt{W}(X, \rho, L)).$$

Remark.

- This is our first quantitative approximation result for neural networks!
- It suffers from the same curse of dimensionality as spline approximation.
- There is no inverse estimate because perceptrons are more general than splines (at least for $d > 1$).
- One may use splines of degree $s - 1 \geq 1$ if $\alpha d < s$, in which case $L = \lceil \log_2(d) \rceil + \lceil \max\{\log_r(s-1), 1\} \rceil + 1$ will do (see proof).

Dictionary learning with splines: proof

Proof.

- Let ϕ be the dictionary of basis splines of degree $s-1$.
- If $s > \alpha d$, spline approximation satisfies

$$B_q^{\alpha d}(X) \subseteq A_q^{\alpha}(X, \mathtt{H}(\phi)) \subseteq A_q^{\alpha}(X, \mathtt{\Sigma}(\phi)).$$

- If $r = d = 1$, set $s = 2$. Then ϕ_λ are hat functions, and

$$\forall \lambda \in \Lambda: \qquad \phi_\lambda \in \mathtt{W}_3(X, \rho, 2).$$

- If $r \geq 2$, the next proposition shows for all $s \in \mathbb{N}$ and some $L, W \in \mathbb{N}$

$$\forall \lambda \in \Lambda: \qquad \phi_\lambda \in \overline{\mathtt{W}_W(X, \rho, L)}.$$

- Either way, by the dictionary learning theorem,

$$A_q^{\alpha}(X, \mathtt{\Sigma}(\phi)) \subseteq A_q^{\alpha}(X, \mathtt{W}(X, \rho, L)).$$ □

Approximating multivariate splines

Standing assumptions, in addition to the previous ones:

- $\phi = (\phi_\lambda)_{\lambda \in \Lambda}$ is the dictionary of dyadic basis splines of degree $s-1$
- $s \geq 2$, and ρ is sigmoidal of order $r \geq 2$

Proposition

There exists $L, W \in \mathbb{N}$ such that

$$\forall \lambda \in \Lambda : \qquad \phi_\lambda \in \overline{\mathtt{W}_W(C([0,1]^d), \rho, L)}.$$

Remark. This verifies the conditions of the dictionary learning theorem for $p = \infty$ and hence for all p.

Approximating multivariate splines: proof

Proof:

- We will show in the next lemma that the univariate cardinal basis spline $\mathcal{N}_s$ satisfies for some W and all $K > 0$ that

$$\mathcal{N}_s \in \overline{\mathtt{W}_W\big(C([-K,K]), \rho, \lceil \max\{\log_r(s-1), 1\}\rceil + 1\big)}.$$

- We will show in the lemma thereafter that multiplication $M_d : \mathbb{R}^d \ni (x_1, \ldots, x_d) \mapsto x_1 \cdots x_d \in \mathbb{R}$ satisfies for some W that

$$M_d \in \overline{\mathtt{W}_W\big(C([0,1]^d), \rho, \lceil \log_2(d)\rceil + 1\big)}.$$

- As basis splines ϕ_λ are translations of products of dilated and translated cardinal basis splines $\mathcal{N}_s$, it follows for all $\lambda \in \Lambda$ that

$$\phi_\lambda \in \overline{\mathtt{W}_W\big(C([0,1]^d), \rho, \lceil \log_2(d)\rceil + \lceil \max\{\log_r(s-1), 1\}\rceil + 1\big)}.$$

□

Approximating univariate splines

Lemma

There exist constants $W \in \mathbb{N}$ *and* $L := \lceil \max\{\log_r(s-1), 1\} \rceil + 1$ *such that the univariate cardinal basis spline* $\mathcal{N}_s$ *satisfies*

$$\forall K > 0 : \qquad \mathcal{N}_s \in \overline{\mathtt{W}_W(C([-K,K]), \rho, L)}.$$

Remark:

- Recall that $\mathcal{N}_s$ is a sum of translated power units $\rho_{s-1}(x) := (x)_+^{s-1}$. Thus, it suffices to consider ρ_{s-1} in place of $\mathcal{N}_s$.
- Moreover, recall that ρ is sigmoidal of order r. We will see that some dilations of ρ uniformly approximate the power unit ρ_r.
- Thus, the lemma really is about changing the power of power units from r to $s-1$.

Approximating univariate splines: proof

Proof: It suffices to show the lemma for ρ_{s-1} in place of $\mathcal{N}_s$.

- Certain dilations of ρ converge to ρ_r:

$$\forall x \in \mathbb{R}: \qquad \rho_r(x) = \lim_{\lambda \to \infty} \lambda^{-r} \rho(\lambda x).$$

- This limit is locally uniform in x, and one obtains

$$\forall K > 0: \qquad \rho_r \in \overline{\mathtt{W}_1(C([-K,K]), \rho, 2)}.$$

- Thus, for $n := \lceil \max\{\log_r(s-1), 1\} \rceil$ and $L := n+1$, one has

$$\forall K > 0: \qquad \underbrace{\rho_r \circ \cdots \circ \rho_r}_{n \text{ times}} \in \overline{\mathtt{W}_1(C([-K,K]), \rho, L)}$$

- Let $p := r^n - (s-1)$ and $W := p+1$. As the p-th derivative is the limit of p-th order difference quotients,

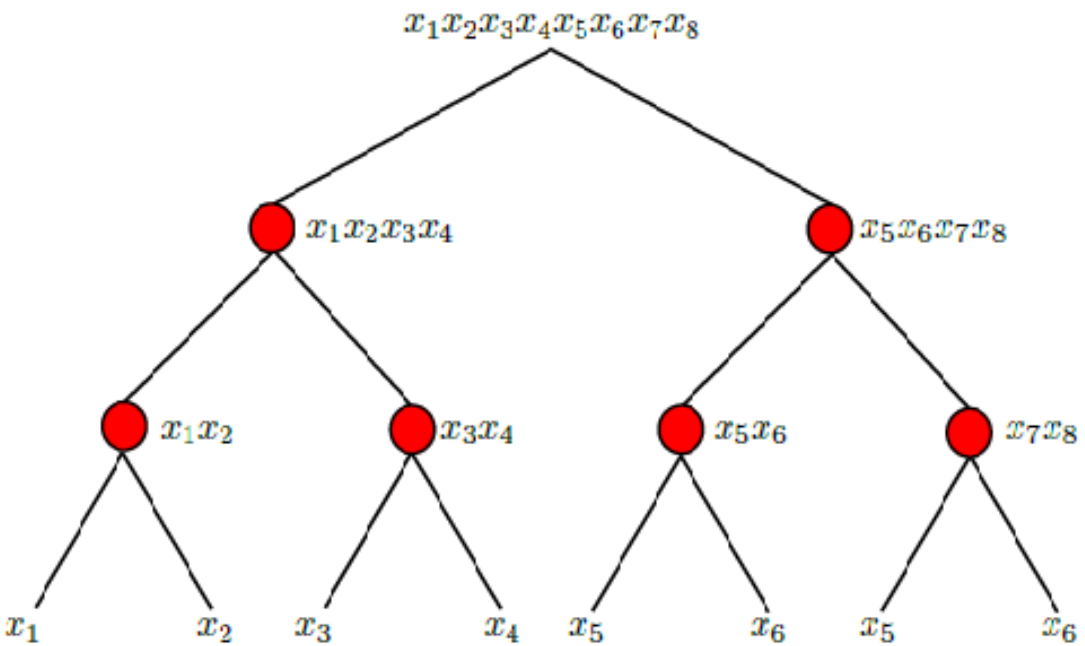
$$\forall K > 0: \qquad \rho_{s-1} = \partial_x^p(\rho_r \circ \cdots \circ \rho_r) \in \overline{\mathtt{W}_W(C([-K,K]), \rho, L)}. \qquad \square$$

Approximating products

Lemma

There exist constants $W \in \mathbb{N}$ *and* $L := \lceil \log_2(d) \rceil + 1$ *such that multiplication* $M_d : \mathbb{R}^d \ni (x_1, \ldots, x_d) \mapsto x_1 \cdots x_d \in \mathbb{R}$ *satisfies*

$$\forall K > 0 : \qquad M_d \in \overline{\mathtt{W}_W(C([-K,K]^d), \rho, L)}.$$



[Petersen]

Remark: The closure is not needed if $\rho = \rho_2$ is the power unit of order 2.

Approximating products: proof

Proof:

- Without loss of generality, $d = 2^n$ for some $n \in \mathbb{N}$.
- Multiplication of 2 variables can be represented as

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

- In terms of the power unit $\rho_2 = (x)_+^2$ this reads as

$$\exists W \in \mathbb{N}\colon \forall K > 0\colon \qquad M_2 \in \mathtt{W}_W(C(\mathbb{R}^2), \rho_2, 2).$$

- Parallelization and concatenation achieves multiplication of 2^n variables:

$$\exists W \in \mathbb{N}\colon \forall K > 0\colon \qquad M_{2^n} \in \mathtt{W}_W(C(\mathbb{R}^2), \rho_2, n+1).$$

- By the previous lemma, ρ_2 can be approximated by 2-layer networks with activation function ρ and bounded number of weights,

$$\exists W \in \mathbb{N}\colon \forall K > 0\colon \qquad M_{2^n} \in \overline{\mathtt{W}_W(C(\mathbb{R}^2), \rho, n+1)}.$$ □

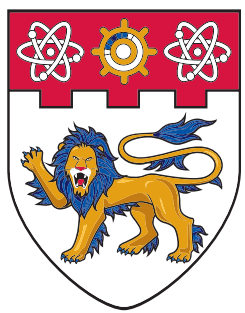
Questions to answer for yourself / discuss with friends

- Repetition: What are higher-order sigmoidal activation functions and how are they related to power units and splines?
- Repetition: At what rate can Hölder C^α or Sobolev H^α functions be approximated by perceptrons?
- Check your understanding: At what rate can multiplication $M_d : \mathbb{R}^d \ni (x_1, \ldots, x_d) \mapsto x_1 \cdots x_d \in \mathbb{R}$ be approximated by perceptrons with second-order sigmoidal activation function?
- Check your understanding: 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 $x \mapsto x^2$?

MH3520:Mathematics of Deep Learning

Chapter 10

Analyticity and approximation by deep networks



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

Last chapters:

- Lower bounds on network approximation rates via dictionaries of splines

This chapter:

- Approximation properties of deep instead of wide networks

Next week:

- Wrap-up and coaching for the final exam.

Analyticity and approximation by deep networks

Main result:

- Analytic functions can be approximated by deep ReLU networks at an exponential rate.
- Previously, with finite depth, we have seen only polynomial rates.

Idea of proof:

- ReLU networks are piecewise linear.
- The number of pieces is polynomial in the width but exponential in the depth.
- With exponentially many linear pieces, one can approximate the square function exponentially well.
- This exponential approximation rate extends to monomials, then polynomials, and then real-analytic functions.

Overview of Chapter 10

1. Approximation by deep ReLU networks
2. Operations on ReLU networks
3. ReLU representation of saw-tooth functions
4. ReLU approximation of multiplication
5. ReLU approximation of analytic functions

Sources for this chapter:

- E, Wang (2018): Exponential convergence of the deep neural approximation for analytic functions. Science China Mathematics 61, pp. 1733–1740. `arXiv:1807.00297`
- Gribonval, Kutyniok, Nielsen, Voigtlaender (2022): Approximation spaces of deep neural networks. Constructive approximation 55:1, 259–367.
- Petersen (2022): Neural Network Theory. Lecture Notes, University of Vienna. `pc-petersen.eu/Neural_Network_Theory.pdf`

MH3520 Chapter 10

Part 1

Approximation by deep ReLU networks

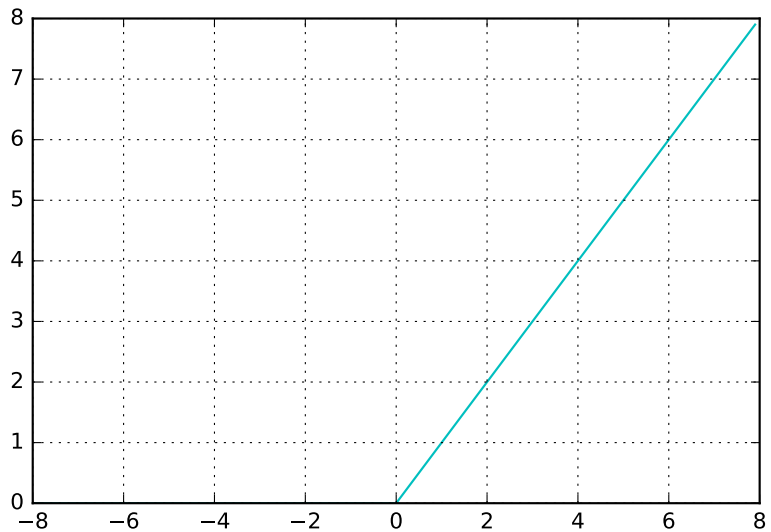
ReLU activation function

Throughout this chapter, we focus on ReLU networks.

Definition

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

$$\rho_1(x) = \max\{0, x\}, \qquad x \in \mathbb{R}.$$



Remark: The ReLU function is not sigmoidal but discriminatory.

Hypothesis classes of deep networks

Remark:

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

Definition

- $\mathtt{L}_n(X, \rho, N)$ denotes the realizations of networks with n layers, activation function ρ, and at most $N = \max\{N_1, \ldots, N_{n-1}\}$ neurons per hidden layer, seen as elements of the function space X.
- $\mathtt{L}_n^M(X, \rho, N)$ denotes the subset of realizations of neural networks with coefficients bounded in absolute value by M.

Remark.

- `L` stands for layers (similarly to `W` for weights and `N` for neurons).
- Previously, we used N for the number of hidden neurons. Now, it is the maximal number of neurons per hidden layer, aka. the width.

Real-analytic functions

Definition

A function $f\colon (-r,r)^d \to \mathbb{R}$ 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 $(a_k)_{k\in\mathbb{N}^d}$. In symbols, $f \in C^\omega((-r,r)^d)$.

Remark:

- The power series converges absolutely on $(-r,r)^d$, i.e.,

$$\forall x \in (-r,r)^d : \quad \sum_{k\in\mathbb{N}^d} |a_k||x|^{|k|} < \infty, \quad \text{where } |k| = k_1 + \cdots + k_d.$$

- On $[-r,r]^d$, the power series may or may not converge absolutely.

Examples of real-analytic functions

Example

Polynomials and trigonometric functions $\exp$, $\sin$, $\cos$ etc. are real analytic on $\mathbb{R}$. The logarithm $\log$ is real analytic on $(0, \infty)$.

Example

Sums, products, and compositions of real-analytic functions are real analytic. The reciprocal of a non-zero real-analytic function is real analytic. The inverse of a real-analytic function with invertible derivative is real analytic.

Counter-example

Bump functions (i.e., compactly supported functions) cannot be real analytic unless they are identically zero. Non-smooth functions cannot be real analytic.

Approximating real-analytic functions

The main result of this chapter is that real-analytic functions admit exponential ReLU approximation rates:

Theorem

Let $r > 1$, and let $f : (-r, r)^d \to \mathbb{R}$ be real-analytic with

$$f(x) = \sum_{k \in \mathbb{N}^d} a_k x^k, \qquad \|f\|_\omega := \sum_{k \in \mathbb{N}^d} |a_k| r^{|k|} < \infty,$$

and let $X = C([-1, 1]^d)$. Then, there exists $c > 0$ such that

$$E_n(f, X, \mathrm{L}^{8\|f\|_\omega}(X, \rho_1, 2d + 10)) \lesssim e^{-cn^{1/(d+2)}}.$$

Remark:

- This is approximation by deep networks with bounded width and bounded coefficients.
- The theorem is proven in Part 5, building upon results of Parts 2–4.

Questions to answer for yourself / discuss with friends

- Repetition: What is a real-analytic function? At what rate can it be approximated by networks, and what is the network's architecture?
- Check your understanding: For what $r > 0$ is the power series $\sum_{k=1}^{\infty}(-1)^{k+1}x^k$ convergent? The limit is a well-known function – which one?
- Check your understanding: If you control the number of layers L and the width N, do you also control the number of non-zero weights W? What's the relation?
- Background: Can you remember the proof that the ReLU function is discriminatory?
- Background: What is the difference between smooth, real-analytic, and holomorphic functions?

MH3520 Chapter 10

Part 2

Operations on ReLU networks

Representation of the identity

Lemma

For each $d, L \in \mathbb{N}$, the identity on $\mathbb{R}^d$ can be realized as a ReLU network:

$$\mathrm{Id}_d = R(\Phi_{d,L}^{\mathrm{Id}}) \in \mathtt{L}_L^1(C(\mathbb{R}^d, \mathbb{R}^d), \rho_1, 2d).$$

Proof:

- This follows from the algebraic relations

$$\rho_1(x) - \rho_1(-x) = x, \qquad \rho_1(\rho_1(x)) = \rho_1(x).$$

- For $L = 1$ we use the network $((\mathrm{Id}_d, 0))$.
- For $L \geq 2$, we use the network

$$\left(\left(\begin{pmatrix} \mathrm{Id}_d \\ -\mathrm{Id}_d \end{pmatrix}, 0 \right), (\mathrm{Id}_{2d}, 0), \ldots (\mathrm{Id}_{2d}, 0), \left((\mathrm{Id}_d, -\mathrm{Id}_d), 0 \right) \right). \qquad \square$$

Lack of sparsity in network compositions

Problem. Composition of networks involves matrix multiplication, which scales quadratically in the number of non-zero weights.

Example

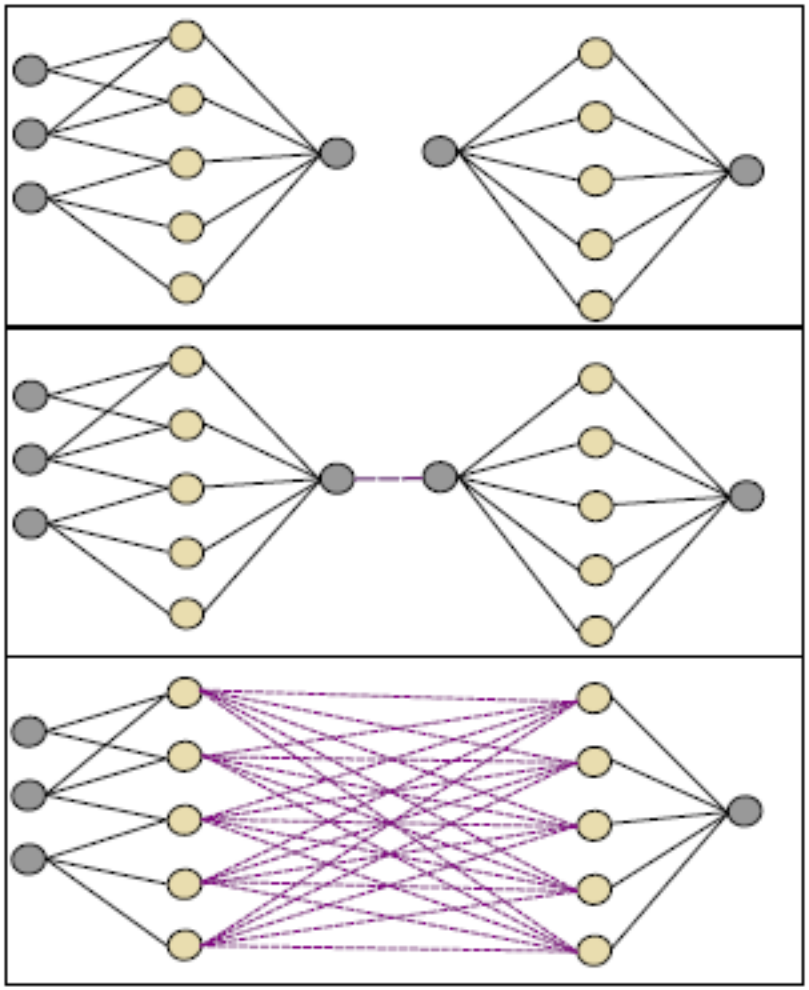
The single-layer networks

$$\Phi_2 := \left(\begin{pmatrix} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \end{pmatrix} \right), \qquad \Phi_1 := \Big(\big((1, \ldots, 1), 0 \big) \Big)$$

have n non-zero weights, but their composition has n^2 non-zero weights:

$$\Phi_2 \bullet \Phi_1 = \left(\begin{pmatrix} \begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{pmatrix}, \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \end{pmatrix} \right).$$

Solution: sparse composition



Top to bottom: two neural networks, their sparse composition, and their composition. [Petersen]

Solution: sparse composition

Definition

The sparse composition of a neural network Φ^2 with input dimension d and a neural network Φ_1 with output dimension d is defined as

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

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 compositions of increased depth.

Properties of sparse composition

Lemma

Sparse composition of networks realizes composition of functions, i.e.,

$$R(\Phi^2 \odot \Phi^1) = R(\Phi^2) \circ R(\Phi^1).$$

The number of layers L, coefficient bound M, maximal number of neurons N per hidden layer, and number of non-zero weights W satisfy

$$L = L_2 + L_1, \qquad M \leq \max\{M_2, M_1\},$$
$$N \leq \max\{N_2, 2d, N_1\}, \qquad W \leq 2(W_2 + W_1).$$

Proof. Counting exercise. □

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

Skip connections

Remark: Recall that a network $\Phi = ((A_L, b_L), \ldots, (A_1, b_1))$ can be represented as a computational graph with edges corresponding to the non-zero entries of the matrices A_i.

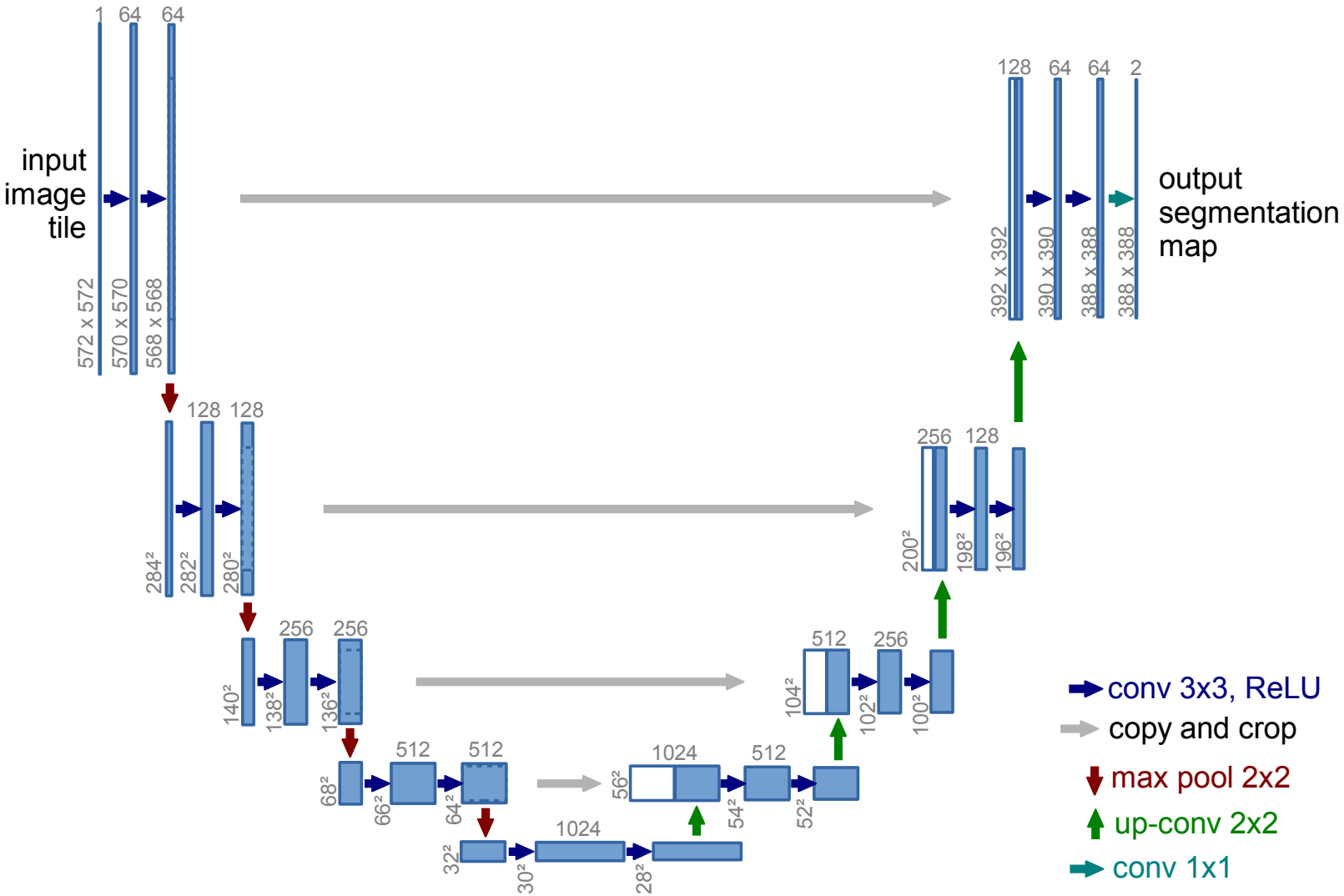
Definition

A skip connection is an edge between non-adjacent layers in the computational graph of a network.

Remark:

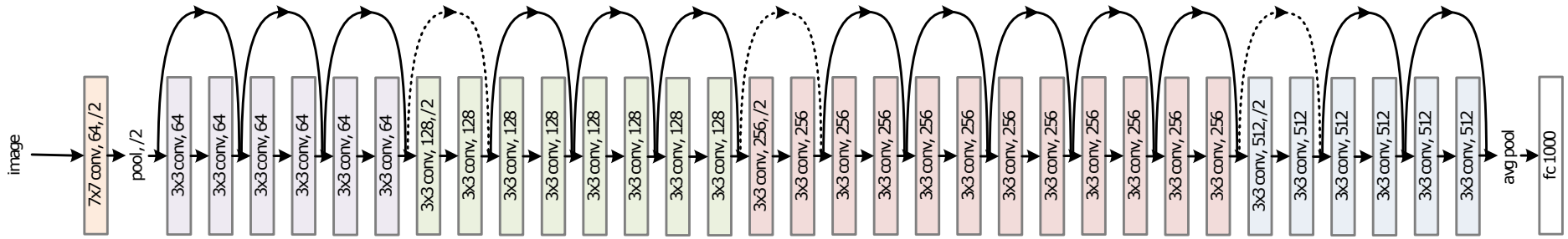
- Networks with skip connections have been highly successful in image recognition.
- The ReLU representation of the identity allows one to rewrite networks with skip connections as networks without skip connections.

Examples of networks with skip connections



U-Net: convolutional network for image segmentation. Skip connections are represented by gray arrows. [Ronneberger e.a.]

Examples of networks with skip connections



ResNet: convolutional network for image recognition. Skip connections are represented by curved arrows. [Schmidhuber e.a.]

Deep linear combinations of networks

Deep linear combinations are linear combinations of perceptrons, which increase the depth but not the width of the networks.

Lemma

For any networks $\Phi_1, \ldots, \Phi_k$ with input dimension d and output dimension n, there exists a network Φ such that

$$R(\Phi) = \sum_i R(\Phi_i),$$

and the number of layers L, coefficient bound M, and maximal number of neurons N per hidden layer satisfy

$$L = \sum_i L_i, \qquad M \leq \max_i M_i, \qquad N \leq \max_i N_i + 2d + 2n.$$

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

$$\Psi_i := \Phi^{\mathrm{sum}} \bullet \mathrm{FP}\left(\Phi^{\mathrm{Id}}_{d,L_i}, \Phi_i, \Phi^{\mathrm{Id}}_{n,L_i}\right) \bullet \Phi^{\mathrm{diag}},$$

where FP denotes full parallelization.

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. □

Remark. The idea of the proof is to express the sum by a recursion (aka. loop), which can be expressed by networks of variable depth and fixed width.

Questions to answer for yourself / discuss with friends

- 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?
- Transfer: Are approximation spaces $A^\alpha(X, \mathtt{L}(X, \rho_1, N))$ well defined?
- Discussion: To what extent are the results of this part limited to ReLU networks?

MH3520 Chapter 10

Part 3

ReLU representation of saw-tooth functions

ReLU representation of the hat function

Lemma

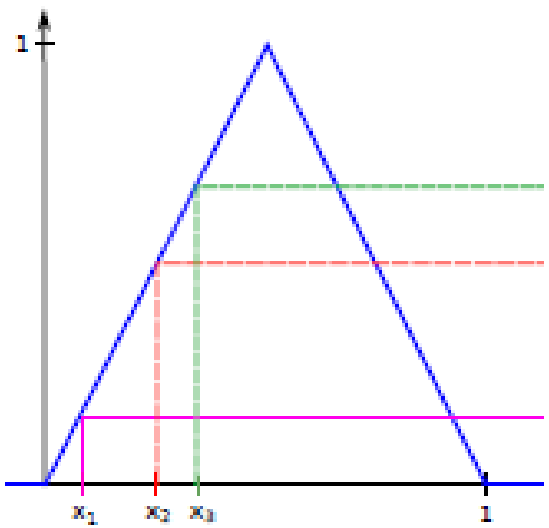
The hat function

$$F(x) := \rho_1(2x) - 2\rho_1(2x-1) + \rho_1(2x-2)$$

is the ReLU realization of the network $\Phi^{\text{hat}} := ((A_2, 0), (A_1, b_1))$,

$$A_2 := (1, -2, 1), \qquad A_1 := (2, 2, 2)^\top, \quad b_1 := (0, -1, -2)^\top$$

with depth $L = 2$, *coefficient bound* $M = 2$, *and* $N = 3$ *hidden neurons.*



Saw-tooth functions

Definition

For any $n \in \mathbb{N}$, the saw-tooth function F_n is defined as $F_n(x) = 0$ for $x \notin (0,1)$ and

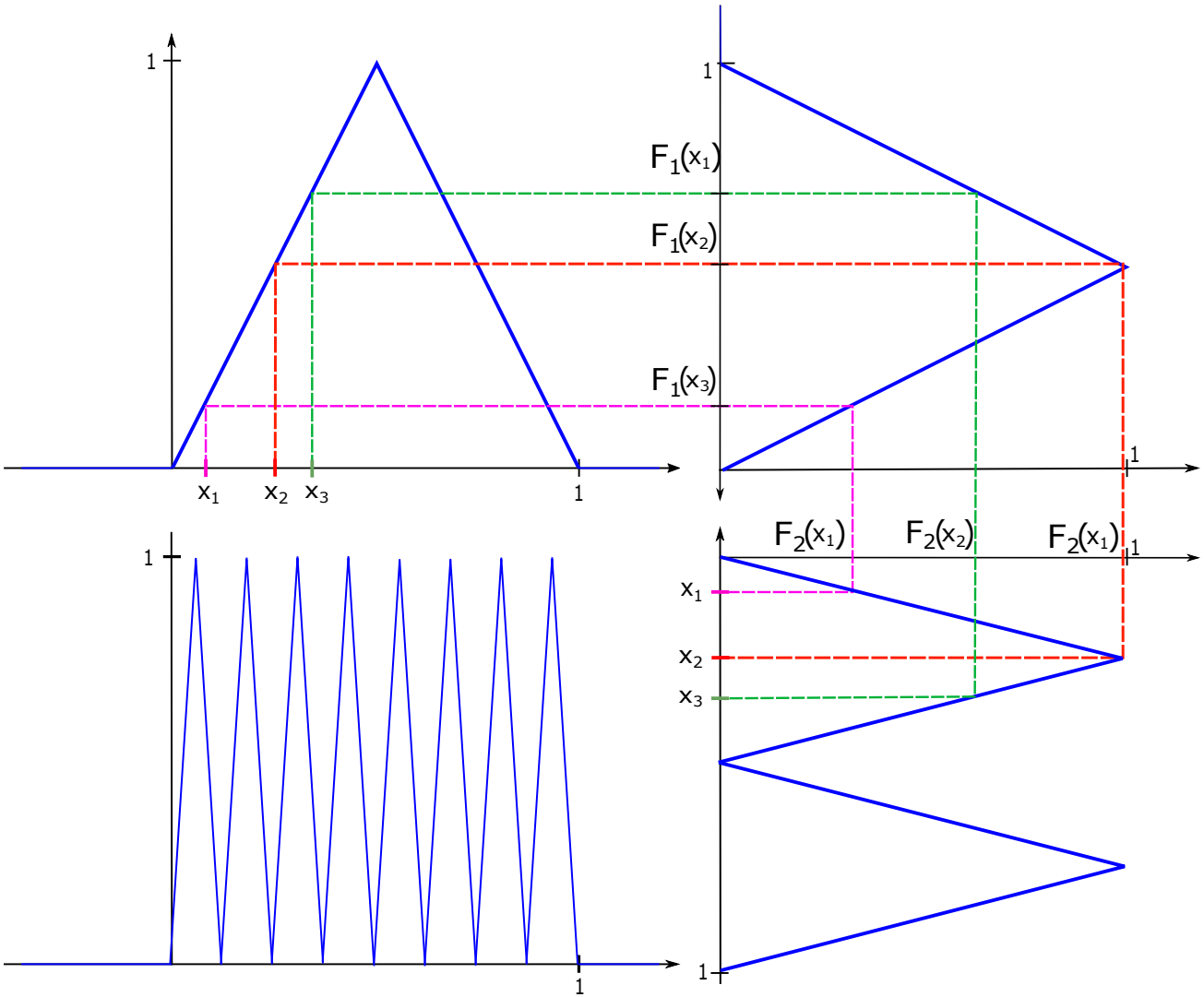
$$F_n(x) := \begin{cases} 2^n(x - i2^{-n}), & x \in [i2^{-n}, (i+1)2^{-n}],\ i \text{ even}, \\ 2^n((i+1)2^{-n} - x), & x \in [i2^{-n}, (i+1)2^{-n}],\ i \text{ odd}. \end{cases}$$

Lemma

The saw-tooth function is the n-fold composition $F_n = F \circ \cdots \circ F$ of the hat function.

Visual proof: next slide.

Saw-tooth functions



Top Left: F_1, Bottom Right: F_2, Bottom Left: F_4.

[Figure from Petersen, Ch. 3]

ReLU representation of saw-tooth functions

Corollary

The saw-tooth function F_n is the ReLU realization of the n-fold composition

$$\Phi_n := \Phi^{\text{hat}} \bullet \cdots \bullet \Phi^{\text{hat}},$$

which has depth $L = n + 1$, coefficient bound $M \leq 4$, and at most $N = 3$ neurons per hidden layer.

Proof:

- F_n is the n-fold composition of hat functions $F = R(\Phi^{\text{hat}})$.
- Φ_n is the n-fold composition of the networks Φ^{hat}. □

The role of depth

Remark: The corollary is surprising for the following reason:

- Consider ReLU realizations of networks with input dimension 1, depth L and at most N neurons per hidden layer.
- For depth $L = 2$, these are piece-wise linear with at most $2N$ pieces. (Actually, at most $N + 1$ pieces.)
- For higher L, there are at most $(2N)^{L-1}$ pieces.
- These are polynomially many in N, but exponentially many in L.
- Thus, saw-tooth functions can be represented very efficiently by deep networks, but not very efficiently by shallow networks.

Limits on inverse estimates

Saw-tooth functions serve as counter-examples for embeddings of network approximation spaces in Besov spaces:

Theorem

There is no continuous embedding

$$A_q^\alpha(X, \mathtt{L}(X, \rho_1, N)) \subseteq B_q^s(X), \qquad \textit{where} \quad X = L^p([0,1]^d),$$

for any $\alpha, s \in (0, \infty)$ *and* $p, q \in (0, \infty]$, *and* $N \geq 3$.

Remark.

- An embedding of the above form is called an inverse estimate.
- For unbounded depth, no such inverse estimate can hold.
- Thus, certain functions are well approximated by deep networks but badly approximated by e.g. splines or trigonometric polynomials.

Limits on inverse estimates

Proof for $p = q = \infty$:

- Then, $B^s_q(X) = C^s([-1,1]^d)$. Wlog. $s < 1$.
- Let $f_k(x_1, \ldots, x_d) = F_k(x_1)$, for the saw-tooth function F_k.
- As F_k has slope $\pm 2^k$ on intervals of length 2^{-k},

$$\|f_k\|_{C^s([-1,1]^d)} \geq \sup_{|x-y| \leq 2^{-n}} \frac{2^n |x-y|}{|x-y|^s} = \frac{2^n 2^{-n}}{2^{-ns}} = 2^{ns}.$$

- However, as $E_{n-1}(f_k, X, \mathtt{L}(X, \rho_1, 3)) = 0$ for all $n \geq k+2$,

$$\|f_k\|_{A^\alpha(X, \mathtt{L}_{k+1}(X, \rho_1, 3))} \leq (k+2)^\alpha \|f_k\|_X \lesssim (k+2)^\alpha.$$

- Thus, $\{k^{-\alpha} f_k : k \in \mathbb{N}\}$ is bounded in A^α but unbounded in C^s. □

Questions to answer for yourself / discuss with friends

- Repetition: How many saw-tooths can be generated by a ReLU network with L layers?
- Check your understanding: How would you prove that the saw-tooth function is a composition of hat functions?
- Check your understanding: Could one get smoother saw-tooths using ρ_2 instead of ρ_1?
- Check your understanding: What is the network approximation rate of a single saw-tooth function? What about the set of all saw-tooth functions?

MH3520 Chapter 10

Part 4

ReLU approximation of multiplication

Saw-tooth approximation of the square function

Lemma

Linear combinations of saw-tooth functions

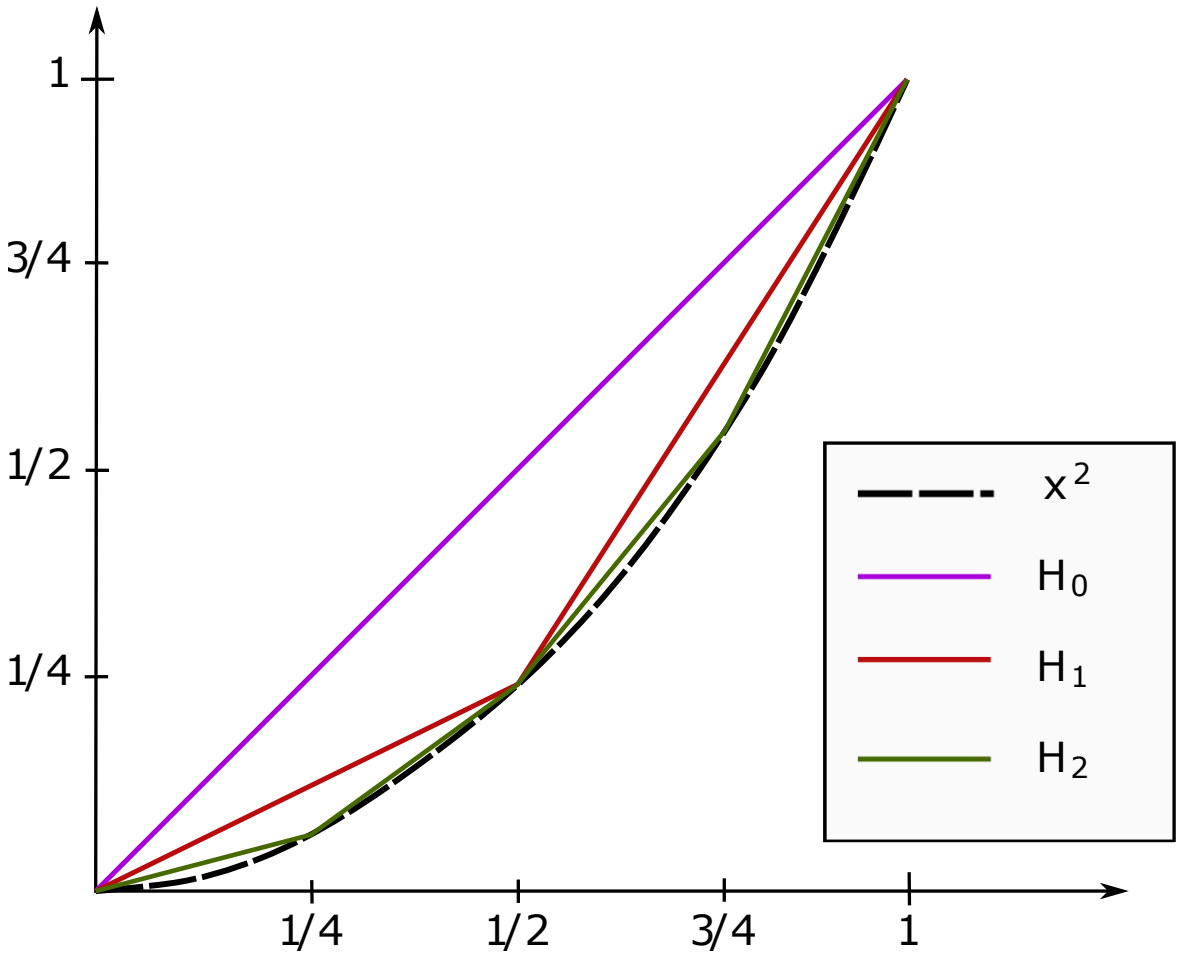
$$H_n(x) := x - \sum_{k=1}^{n} F_k(x) 2^{-2k}, \qquad n \in \mathbb{N},\ x \in \mathbb{R},$$

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.

Saw-tooth approximation of the square function



[Petersen]

Approximants $H_n(x) := x - \sum_{k=1}^{n} F_k(x) 2^{-2k}$ of the square function x^2.

Proof: Saw-tooth approximation of the square function

Proof:

- By induction, the function H_n is piecewise linear with break points $k2^{-n}$ 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

$$x^* = \frac{u+\ell}{2}, \qquad H_n(x^*) - (x^*)^2 = \left(\frac{u-\ell}{2}\right)^2 = 2^{-2(n+1)}. \qquad \square$$

ReLU approximation of the square function

Lemma

The square function $x \mapsto x^2$ can be approximated by ReLU networks at an exponential rate:

$$E_{n+1}(x^2, X, \mathtt{L}^4(X, \rho_1, 5)) \leq 2^{-2n}, \qquad \textit{where} \quad X = C([-1, 1]).$$

Remark. I will show one unsuccessful but instructive attempt of proof, and then the real proof.

Attempted proof: approximating the square function

Attempted proof:

- Approximate the square function by saw-tooth functions:

$$\sup_{x\in[0,1]} \left|x^2 - H_n(x)\right| \leq 2^{-2(n+1)}, \qquad H_n(x) = x - \sum_{k\leq n} F_k 2^{-2k}.$$

- Represent saw-tooths by networks and synchronize their depth:

$$F_k = R(\Phi^{\text{hat}} \bullet \cdots \bullet \Phi^{\text{hat}}) = R(\Phi^{\text{Id}}_{1,n-k} \odot \Phi^{\text{hat}} \bullet \cdots \bullet \Phi^{\text{hat}}).$$

- Take linear combinations to get networks of width proportional to n.
- Alternatively, take deep linear combinations to get networks of depth proportional to n^2.
- This strategy is non-recursive and sub-optimal. □

Proof: approximating the square function

Proof:

- As before, approximate the square by saw-tooth functions:

$$\sup_{x\in[0,1]} \left|x^2 - H_n(x)\right| \le 2^{-2(n+1)}, \qquad H_n(x) = x - \sum_{k\le n} F_k 2^{-2k}.$$

- Recall that F_n is the n-fold composition of the hat function

$$F(x) := 2\rho_1(x) - 4\rho_1(x - \tfrac{1}{2}) + 2\rho_1(x-1),$$

and note that $H_n(x) = H_{n-1}(x) - 2^{-2n} F_n(x)$.

- Write a recursion for (F_n, H_n):

$$\begin{cases} F_n(x) = 2\rho_1(F_{n-1}(x)) - 4\rho_1(F_{n-1}(x) - \frac{1}{2}) + 2\rho_1(F_{n-1}(x) - 1), \\ H_n(x) = \rho_1(H_{n-1}(x)) - \rho_1(-H_{n-1}(x)) - 2^{-2n} F_n(x), \end{cases}$$

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:

$$\begin{pmatrix} F_n \\ H_n \end{pmatrix} = A_2 \rho_1 \left(A_1 \begin{pmatrix} F_{n-1} \\ H_{n-1} \end{pmatrix} - b_1 \right),$$

$$A_2(x) = \begin{pmatrix} 2 & -4 & 2 & 0 & 0 \\ -2^{-2n+1} & 2^{-2n+2} & 2^{-2n+1} & 1 & -1 \end{pmatrix},$$

$$A_1(x) = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & -1 \end{pmatrix}, \qquad b_1 = \begin{pmatrix} 0 \\ 1/2 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$

- 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 of depth $n + 2$, width 5, and weights bounded by 4. □

ReLU approximation of multiplication

By polarization, approximation of the square can be turned into approximation of products:

Theorem

Multiplication $(x, y) \mapsto xy$ can be approximated by ReLU networks at an exponential rate:

$$E_{n+1}(xy, X, \mathrm{L}^8(X, \rho_1, 10)) \leq 2^{-2n-1}, \qquad \textit{where} \quad X = C([-1, 1]^2).$$

Proof. Use $xy = (\frac{x+y}{2})^2 - (\frac{x-y}{2})^2$ and the previous lemma. □

Remark: On domains $x, y \in [-K, K]$, the weight bound changes to a quadratic polynomial in K.

Questions to answer for yourself / discuss with friends

- Repetition: How can multiplication be approximated by ReLU networks at an exponential rate?
- Check your understanding: Why is a secant approximation of the square function worst half-way between the abscissas of the intersection?
- Transfer: Compare the ReLU approximation of multiplication to the sigmoidal approximation of multiplication.
- Discussion: Using coding theory, we established polynomial upper bounds on network approximation rates. Are they in contradiction to the exponential approximation rate established here?

MH3520 Chapter 10

Part 5

ReLU approximation of analytic functions

Approximating monomials

We start with an auxiliary lemma on approximation of monomials.

Lemma

Monomials can be approximated by ReLU networks at an exponential rate: for any indices $i_1, \ldots, i_p \in \{1, \ldots, d\}$,

$$E_{p(n+1)}(x_{i_1} \cdots x_{i_p}, \mathtt{L}^8(X, \rho_1, 2d + 10)) \leq p 2^{-2n-1},$$

where $x_1, \ldots, x_d$ *are the coordinates in* $[-1, 1]^d$, *and* $X = C([-1, 1]^d, \mathbb{R})$.

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:

- Multiplication with skip connections

$$M_i(x_1, \dots, x_d, y) := (x_1, \dots, x_d, x_i y)$$

can be approximated in $Y := C([-1,1]^{d+1}, \mathbb{R}^{d+1})$ by

$$R(\Phi_i) \in \mathtt{L}^8_{n+1}(Y, \rho_1, 2d+10), \qquad \|M_i - R(\Phi_i)\|_Y \leq 2^{-2n-1}.$$

- Therefore, the desired map

$$\mathrm{pr}_{d+1} \circ M_{i_p} \circ \cdots \circ M_{i_1} \circ (\mathrm{Id}_d, 1)$$

can be approximated in $X := C([-1,1]^d, \mathbb{R})$ by the realization of

$$\Phi := \left(\begin{pmatrix} 0_d & 1 \end{pmatrix}, 0 \right) \bullet \Phi_{i_p} \odot \cdots \odot \Phi_{i_1} \bullet \left(\begin{pmatrix} \mathrm{Id}_d \\ 0_d \end{pmatrix}, \begin{pmatrix} 0_d \\ 1 \end{pmatrix} \right).$$

- As $R(\Phi_i)$ is 1-Lipschitz and bounded by 1, using dummy variables x,

$$R(\Phi) \in \mathtt{L}^8_{p(n+1)}(X, \rho_1, 2d+10), \quad \|x_{i_1} \cdots x_{i_p} - R(\Phi)\|_X \leq p 2^{-2n-1}.$$

□

Approximating real-analytic functions

We are now ready to prove the main theorem of Part 1:

Theorem

Let $r > 1$, let $f : (-r, r)^d \to \mathbb{R}$ be real-analytic with

$$f(x) = \sum_{k \in \mathbb{N}^d} a_k x^k, \qquad \|f\|_\omega := \sum_{k \in \mathbb{N}^d} |a_k| r^{|k|} < \infty,$$

where $|k|$ is the 1-norm, and let $X = C([-1, 1]^d)$. Then,

$$\exists c > 0 : \qquad E_n(f, X, \mathtt{L}^{8\|f\|_\omega}(X, \rho_1, 2d + 10)) \lesssim e^{-cn^{1/(d+2)}}.$$

Remark:

- The strategy is simple: truncate the power series and approximate the monomials by deep ReLU networks.
- Both operations admit exponential error bounds.

Approximating real-analytic functions

Proof.

- We start with an auxiliary bound: The derivatives of f are given by power series, which are absolutely converging on $[-1,1]^d$. Therefore,

$$\sum_{k\in\mathbb{N}^d} |a_k||k| \lesssim \|f\|_\omega < \infty.$$

- Approximating monomials x^k by $\psi_k \in \mathrm{L}^8_{|k|(n+1)}(X,\rho_1,2d+10)$ yields

$$\begin{aligned}\Big\|f - \sum_{|k|\le p} a_k\psi_k\Big\|_X &\le \sum_{|k|>p} |a_k| + \sum_{|k|\le p} |a_k|\|x^k - \psi_k\|_X \\ &\le r^{-p}\sum_{|k|>p} |a_k| r^{|k|} + \sum_{|k|\le p} |a_k||k| 2^{-2n-1} \\ &\lesssim \left(r^{-p} + 2^{-2n-1}\right) = e^{-p\log r} + e^{-(2n+1)\log 2}.\end{aligned}$$

- Setting $p := (2n+1)\lceil \log(2/r)\rceil$ yields an error $\lesssim e^{-cn}$ for $c := 2\log 2$.

Approximating real-analytic functions

- There are $\binom{p+d}{d}$ perceptrons ψ_k with $|k| \leq p$.
- Taking deep linear combinations, one obtains that

$$\sum_{|k|\leq p} a_k \psi_k \in \mathtt{L}^8_L(X, \rho_1, 2d+11), \quad \text{where} \quad L = p(n+1)\binom{p+d}{d}.$$

- We upper-bound the depth L using the estimate $\frac{d^d}{d!} \leq \sum_k \frac{d^k}{k!} = e^d$:

$$L = p(n+1)\frac{(p+d)\cdots(p+1)}{d!} \leq p(n+1)\left(\frac{p+d}{d/e}\right)^d \lesssim n^{d+2}.$$

- Overall, we have achieved an error $\lesssim e^{-cn}$ with depth n^{d+2}. □

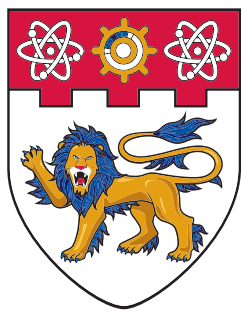
Questions to answer for yourself / discuss with friends

- Repetition: How can real-analytic functions be approximated by ReLU networks at an exponential rate?
- 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?

MH3520:Mathematics of Deep Learning

Chapter 11

Harmonic analysis and approximation by wide networks



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

In spline approximation chapter:

- Dictionary learning with splines: If a function can be approximated by splines, it can be approximated at least as well by neural networks

This chapter:

- Dictionary learning with wavelets and shearlets: If a function can be approximated by wavelets or shearlets, it can be approximated at least as well by neural networks

Next chapter:

- Networks are encoders, and there are information-theoretic limits on encoding rates

Overview of Chapter 11

1. Harmonic analysis
2. Gabor analysis
3. Wavelet analysis
4. Shearlet analysis
5. Signal classes and their approximation rates

Sources for this chapter:

- Dahlke, De Mari, Grohs, Labate (2015): Harmonic and applied analysis—from groups to signals. Birkhäuser.
- Gribonval Kutyniok Nielsen Voigtlaender (2022): Approximation spaces of deep neural networks. Constructive approximation 55:1, 259–367.

MH3520 Chapter 11

Part 1

Harmonic analysis

History of harmonic analysis

- Harmonic analysis grew out of Fourier analysis: virtually any periodic function can be represented as a weighted series of sines and cosines.
- The Fast Fourier Transform computes Fourier coefficients in logarithmic time, and people began to use it in all kind of applications – even where it is not useful!
- For instance, Fourier analysis is ill suited for describing frequency changes over time, and local coefficient errors have global effect.
- The short-time Fourier transform, also known as Gabor analysis, is an ad-hoc solution with limited success: one restricts the Fourier transform to a shifting time window.
- The break–through came with wavelets, which use wide time windows for low frequencies and narrow time windows for high frequencies.
- Abstract harmonic analysis consolidates wavelet theory using group representations and has led to far-reaching generalizations.

Groups

Definition

A group is a tuple $(G, *, e, ^{-1})$, where G is a set and

- $* : G \times G \to G$ is associative, i.e., $(a * b) * c = a * (b * c)$
- $e \in G$ is a unit element, i.e., $e * a = a * e = a$.
- $^{-1} : G \to G$ satisfies $a * a^{-1} = a^{-1} * a = e$.

Remark.

- One often calls $*$ the group multiplication and $^{-1}$ the group inversion, but these may differ from multiplication and inversion of real numbers.
- One speaks of multiplicative or additive groups if $*$ is a multiplication or addition in the ordinary sense.

Examples of groups

Example (Real numbers and non-zero real numbers)

$\mathbb{R}_+$ is an additive group, and $\mathbb{R}^\times := \mathbb{R} \setminus \{0\}$ is a multiplicative group.

Example (General linear group)

The set $GL(\mathbb{R}^d)$ of invertible $d \times d$ matrices is a multiplicative group and is called general linear group.

Example (Affine group)

The set $Aff(\mathbb{R}^d)$ of invertible affine functions $\mathbb{R}^d \to \mathbb{R}^d$ is a group with composition and inverse as group operations. In coordinates, they read as

$$(A_1, b_1) * (A_2, b_2) = A_1 A_2 x + A_1 b_2 + b_1, \quad (A, b)^{-1} = (A^{-1}, -A^{-1} b).$$

Group homomorphisms

Definition

A group homomorphism is a function $f : G \to H$ between groups G and H such that $f(a * b) = f(a) * f(b)$.

Remark.

- Note: in $f(a * b)$ the multiplication is in G, whereas in $f(a) * f(b)$, the multiplication is in H.
- A group homomorphism maps the unit element to the unit element and inverses to inverses: $f(e) = e$, $f(a^{-1}) = f(a)^{-1}$.

Example

The exponential map is a group homomorphism $\mathbb{R} \to \mathbb{R}_{>0}$ because $\exp(a + b) = \exp(a)\exp(b)$.

Group representations

Definition

A representation of a group G on a vector space X is a group homomorphism $\pi : G \to GL(X)$.

Remark.

- Think of G as acting on X via linear transformations, i.e., every group element defines a linear transformation of X.
- In harmonic analysis, X is an L^2 function space, and the linear transformations are norm-preserving.

Example

Affine transformation is a representation

$$\pi : Aff(\mathbb{R}^d) \to GL(L^2(\mathbb{R}^d)), \quad \pi(A,b)f(x) := |\det A|^{-1/2} f\big(A^{-1}(x-b)\big).$$

The harmonic analysis cookbook

Recipe

- Choose a group representation $\pi : G \to GL(L^2(\mathbb{R}^d))$
- Fix some $g_i \in G$, which are 'spread evenly' over G.
- Fix one (or sometimes more than one) function $\psi \in L^2(\mathbb{R}^d)$, which is called mother wavelet, analyzing function, or generator.
- Use the functions $\pi(g_i)\psi$ as a dictionary

Result

- Under suitable conditions, the dictionary is a Hilbert frame (i.e., a Banach frame for $L^2(\mathbb{R}^d)$ with sequence space ℓ^2).
- Other choices of ψ and g_i lead to equivalent frames.

Main uses of harmonic analysis

Analysis

- A signal $f \in L^2(\mathbb{R}^d)$ is analyzed to a sequence $Af := (\langle f, \pi(g_i)\psi \rangle)_i$.

Synthesis

- A coefficient sequence c is synthesized to a function $Sc := \sum_i c_i \phi_i$
- $\phi_i := \pi(g_i)\psi$ if these are an orthonormal basis. In general, ϕ_i are defined as linear combinations of $\pi(g_i)\psi$ such that the reconstruction relation holds.

Reconstruction

- The reconstruction relation $f = SAf$ holds.
- The atomic decomposition $f = \lim_{n\to\infty} S(\mathbb{1}_{\{1,\dots,n\}} Af)$ allows one to truncate the coefficient sequence at high n with negligible error.

Approximation spaces in harmonic analysis

Notation

- Write ϕ for the frame $\pi(g_i)\psi$ in some canonical order.
- Write e for the unit vectors in sequence space, ordered accordingly.

Non-sparse approximation

- For sequences, $A_q^\alpha(\ell^p, \mathtt{H}(e))$ is the Besov space $\mathfrak{b}_q^\alpha(\ell^p)$.
- For functions, $A_q^\alpha(L^p(\mathbb{R}^d), \mathtt{H}(\phi))$ coincides with $\{f : Af \in \mathfrak{b}_q^\alpha(\ell^p)\}$.

Sparse approximation

- For sequences, $A_q^\alpha(\ell^p, \Sigma(e))$ is the Lorentz space $\ell^{r,q}$ with $r := 1/(\alpha + 1/p)$.
- For functions, $A_q^\alpha(L^p(\mathbb{R}^d), \Sigma(\phi))$ coincides with $\{f : Af \in \ell^{r,q}\}$.

Questions to answer for yourself / discuss with friends

- Repetition: What is a group representation? How are group representations used in harmonic analysis?
- Check your understanding: Why is the affine representation defined using A^{-1} instead of A?
- Transfer: Can signal analysis or synthesis be implemented using neural networks?

MH3520 Chapter 11

Part 2

Gabor analysis

Group representations in Gabor analysis

Example

Modulation is a representation

$$\mathbb{R}^d \ni a \mapsto E_a \in GL(L^2(\mathbb{R}^d)), \qquad E_a f(x) = e^{2\pi i \langle a,x \rangle} f(x).$$

Example

Translation is a representation

$$\mathbb{R}^d \ni b \mapsto T_b \in GL(L^2(\mathbb{R}^d)), \qquad T_b f(x) = f(x-b).$$

Example

Multiplication by a complex phase is a representation

$$\mathbb{R} \ni c \mapsto M_c \in GL(L^2(\mathbb{R}^d)), \qquad M_c f(x) = e^{2\pi i c} f(x).$$

Heisenberg group and Schrödinger representation

We bundle up modulation, translation, and multiplication into a group with representation on $L^2(\mathbb{R}^d)$:

Definition

The Heisenberg group is the set $G := \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}$ with multiplication

$$(a_1, b_1, c_1) * (a_2, b_2, c_2) := \left(a_1 + a_2, b_1 + b_2, c_1 + c_2 - \tfrac{1}{2}\langle a_1, b_2\rangle - \tfrac{1}{2}\langle a_2, b_1\rangle\right).$$

Definition

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

$$\pi(a, b, c)f := M_c T_b E_a f.$$

for modulation E_a, translation T_b, and multiplication M_c.

How to come up with this definition

- One takes the Schrödinger representation as a starting point.
- Then, one computes (a, b, c) such that

$$M_{c_1} T_{b_1} E_{a_1} M_{c_2} T_{b_2} E_{a_2} f = \cdots = M_c T_b E_a f$$

  and defines multiplication as $(a_1, b_1, c_1) * (a_2, b_2, c_2) := (a, b, c)$.
- This automatically guarantees associativity of the multiplication.
- Multiplication M_c is needed because $T_{b_1} E_{a_1} T_{b_2} E_{a_2}$ can be written as $T_b E_a$ only up to a multiplicative constant.
- For signal analysis, M_c is irrelevant because $\langle f, M_c T_b E_a \psi \rangle$ equals $\langle f, T_b E_a \psi \rangle$ up to the constant $e^{-2\pi i c}$, independent of f, a, b.

Gabor frames

Theorem

Under certain conditions on $\psi \in L^2(\mathbb{R}^d)$, and for sufficiently small $\alpha, \beta > 0$, the dictionary

$$E_a T_b \psi, \qquad a \in \alpha\mathbb{Z}, \quad b \in \beta\mathbb{Z},$$

is a Hilbert frame for $L^2(\mathbb{R}^d)$.

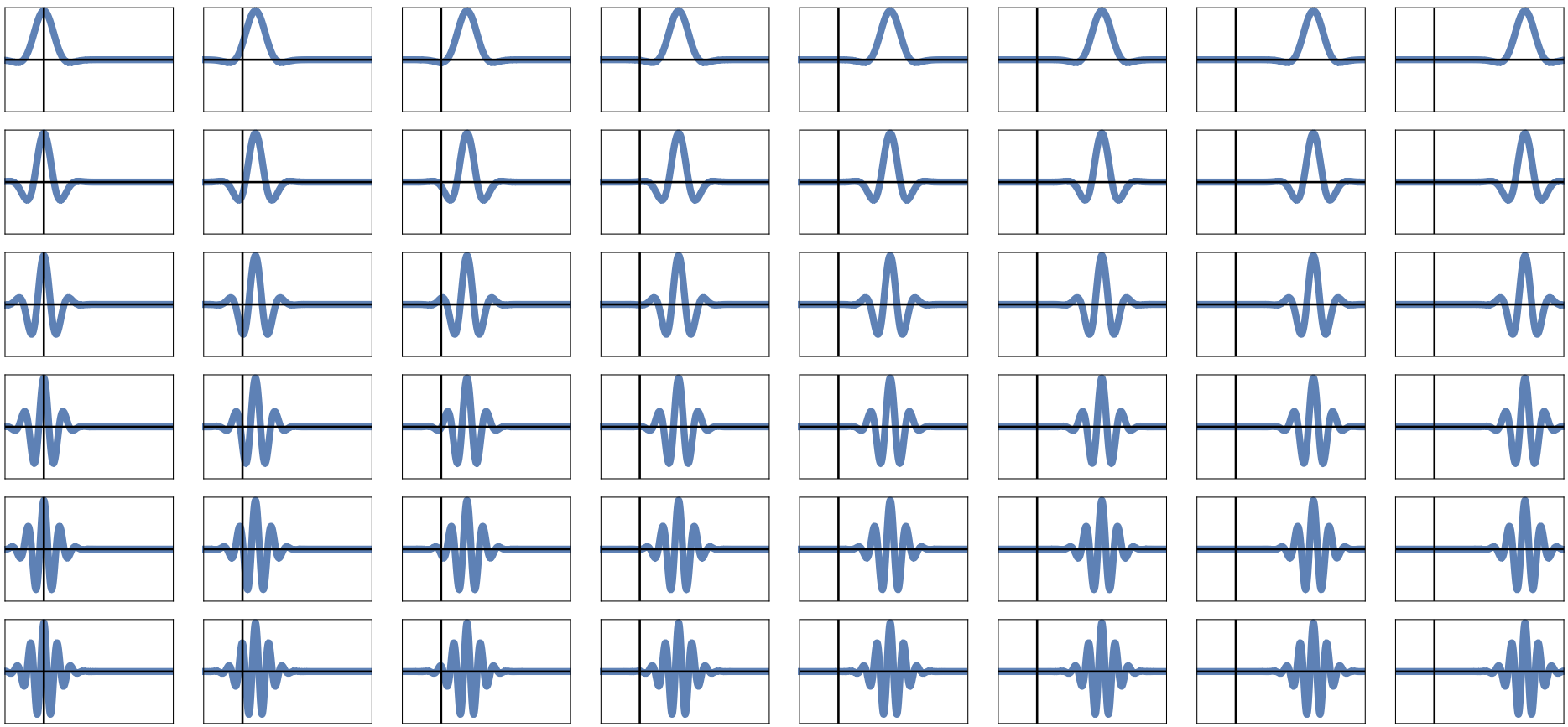
Remark.

- This coincides with $\pi(a, b, c)\psi = M_c T_b E_a \psi$ for suitable (a, b, c).
- The non-sparse approximation spaces are called modulation spaces, and the sparse approximation spaces don't have a name.

Example of Gabor frames

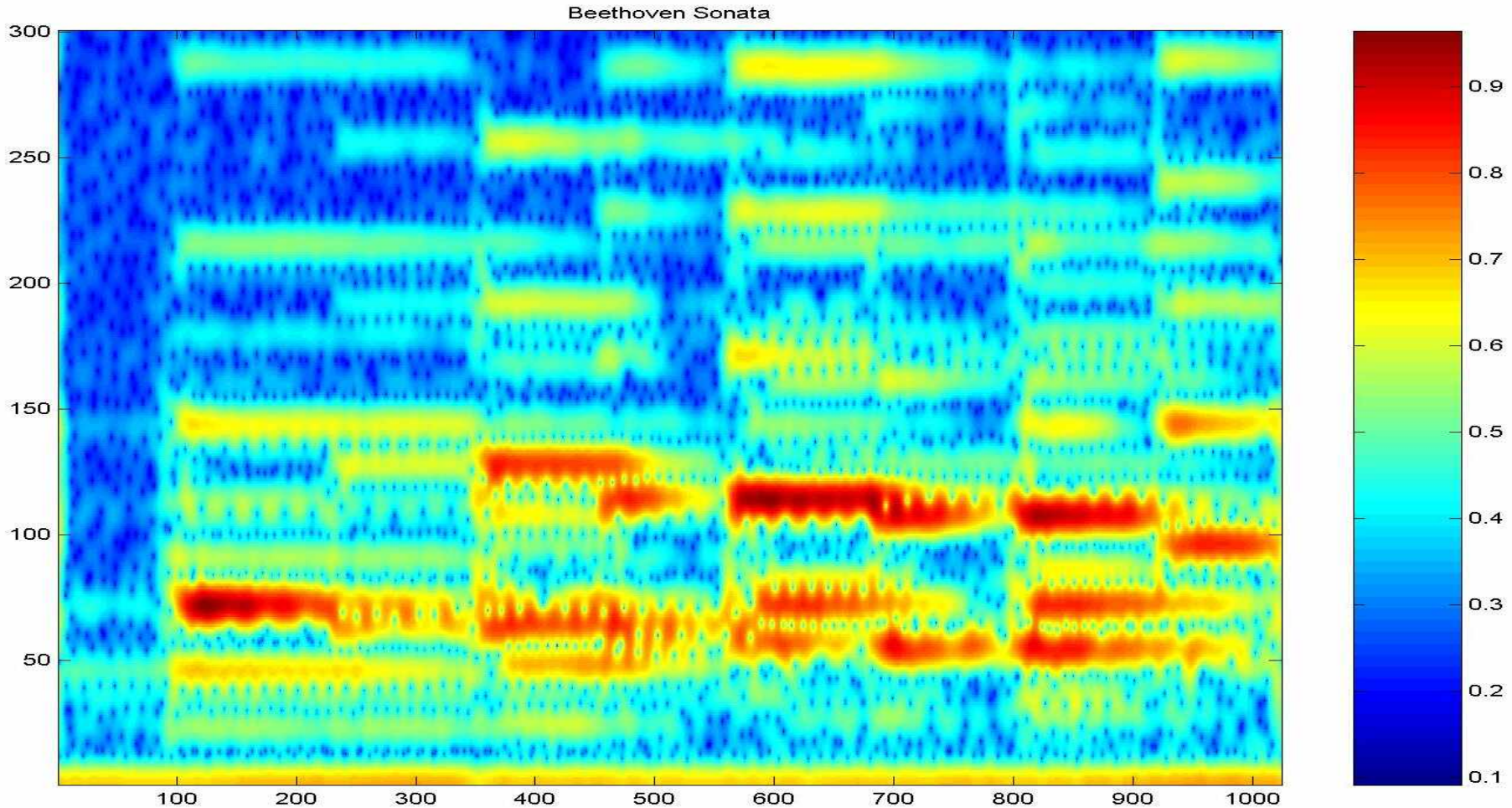
Example

The Gauss function $\psi(x) = e^{-\|x\|^2/2}$ works well. One can think of it as a smoothened time window.



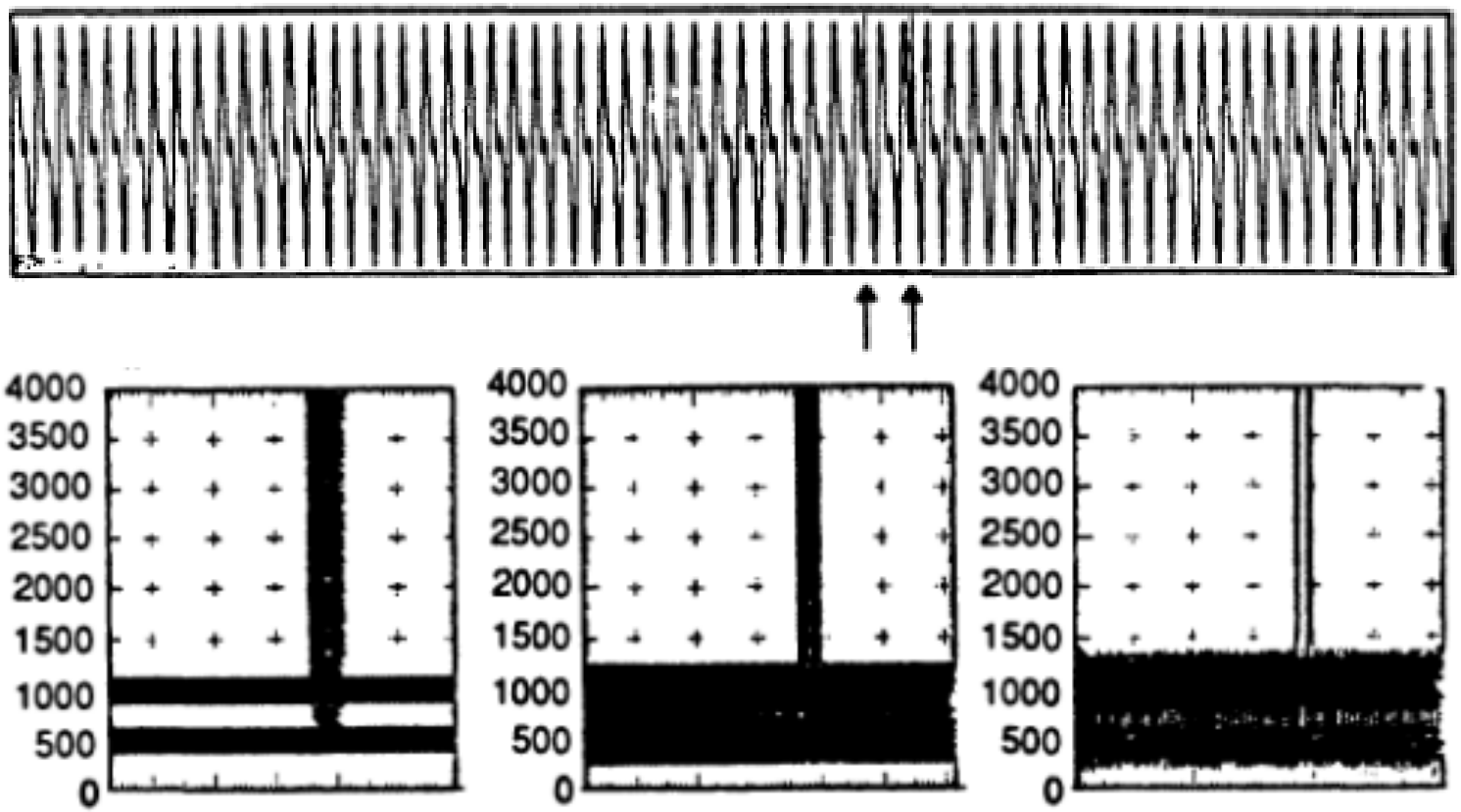
Short-time Fourier transform

The analysis operator for Gabor frames is the short-time Fourier transform.



Intensity (color-coded) of an audio signal, plotted over time (horizontal) and frequency (vertical). [Feichtinger (2015)]

Problems with the short-time Fourier transform



Top: A periodic signal consisting of 2 base frequencies with 2 pulses.
Bottom: Wide time windows (left) distinguish the base frequencies but not the pulses, and the other way round for narrow time windows (right).

Questions to answer for yourself / discuss with friends

- Repetition: Describe the Heisenberg group, the Schrödinger representation, and Gabor frames.
- Check your understanding: Gabor frames involve negative frequencies. Why? Can this be avoided?
- Discussion: Is Gabor analysis related to musical scores?
- Transfer: How would you implement Gabor analysis or synthesis using neural networks?

MH3520 Chapter 11

Part 3

Wavelet analysis

Group representations in wavelet analysis

Example

Dilation is a representation

$$\mathbb{R}^\times \ni a \mapsto D_a \in GL(L^2(\mathbb{R}^d)), \qquad D_a f(x) = |a|^{-d/2} f(a^{-1}x).$$

Example

Translation is a representation

$$\mathbb{R}^d \ni b \mapsto T_b \in GL(L^2(\mathbb{R}^d)), \qquad T_b f(x) = f(x-b).$$

Wavelet group and wavelet representation

We bundle up dilations and translations into a group with representation on $L^2(\mathbb{R}^d)$.

Definition

The wavelet group is the set $G := \mathbb{R}^\times \times \mathbb{R}^d$ equipped with the multiplication

$$(a_1, b_1) * (a_2, b_2) := (a_1 a_2, a_1 b_2 + b_1).$$

Definition

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

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

for dilations D_a and translations T_b.

Wavelet frames

Theorem

Under certain conditions on ψ, the wavelet dictionary given by

$$D_a T_b \psi, \qquad a \in \pm 2^{\mathbb{Z}}, \quad b \in \mathbb{Z}^d,$$

is a Hilbert frame for $L^2(\mathbb{R}^d)$.

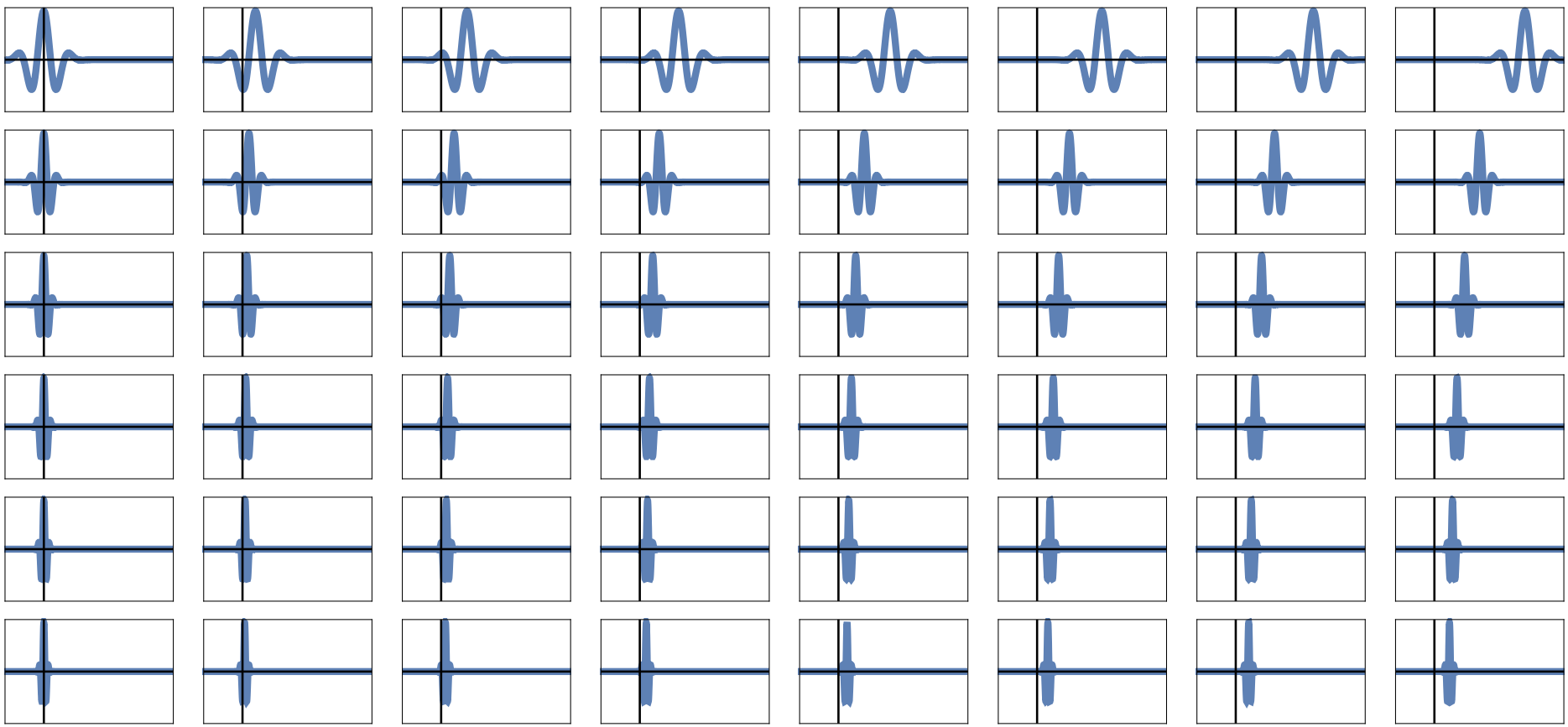
Remark.

- The dictionary coincides with $\pi(a,b)\psi = T_b T_a \psi$ for suitable a, b.
- The corresponding non-sparse approximation spaces are Besov spaces, and the sparse approximation spaces have no name.

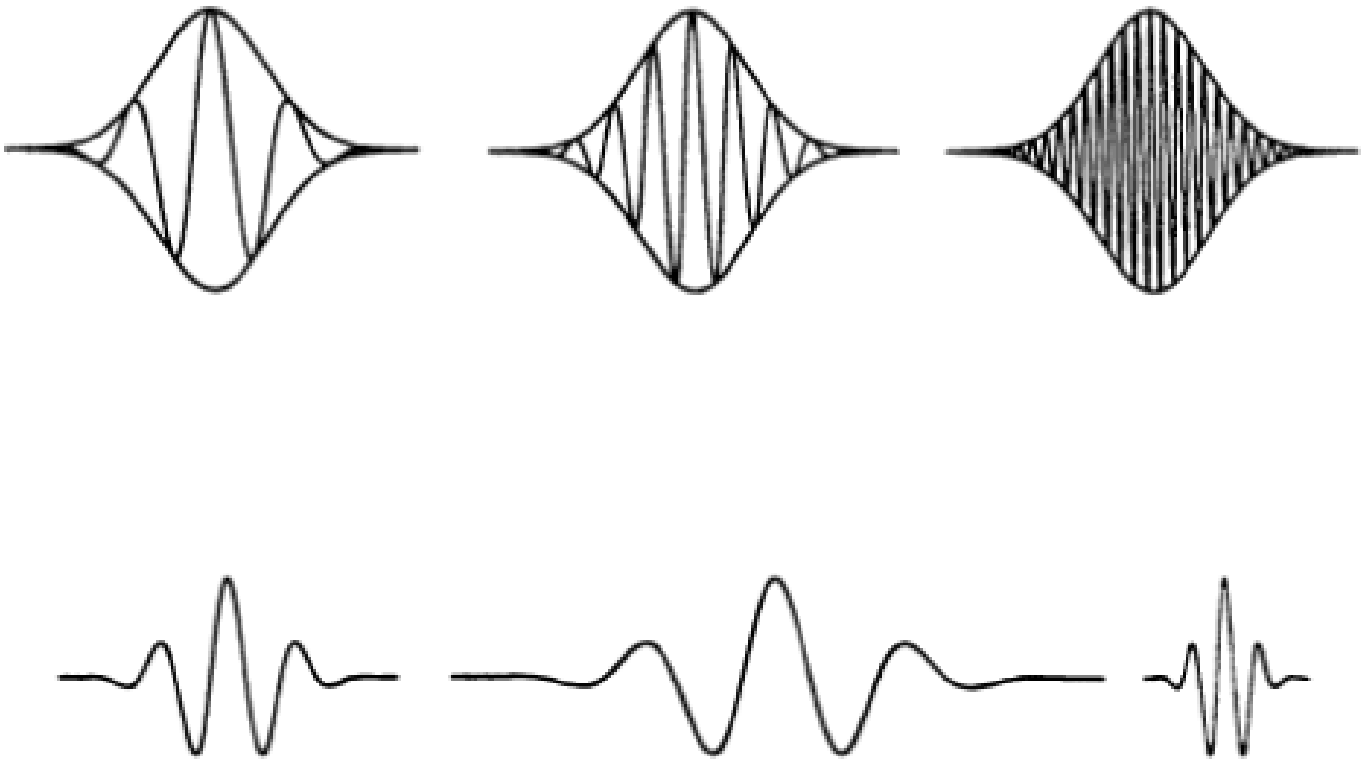
Example of wavelet frames

Example

The Morlet wavelet works well and is given by $\psi(x) = e^{2\pi i \langle c,x \rangle} e^{-\|x\|^2/2}$, for some fixed frequency $c \in \mathbb{Z}^d$. Typically, $c = (5, \ldots, 5)$ is used.



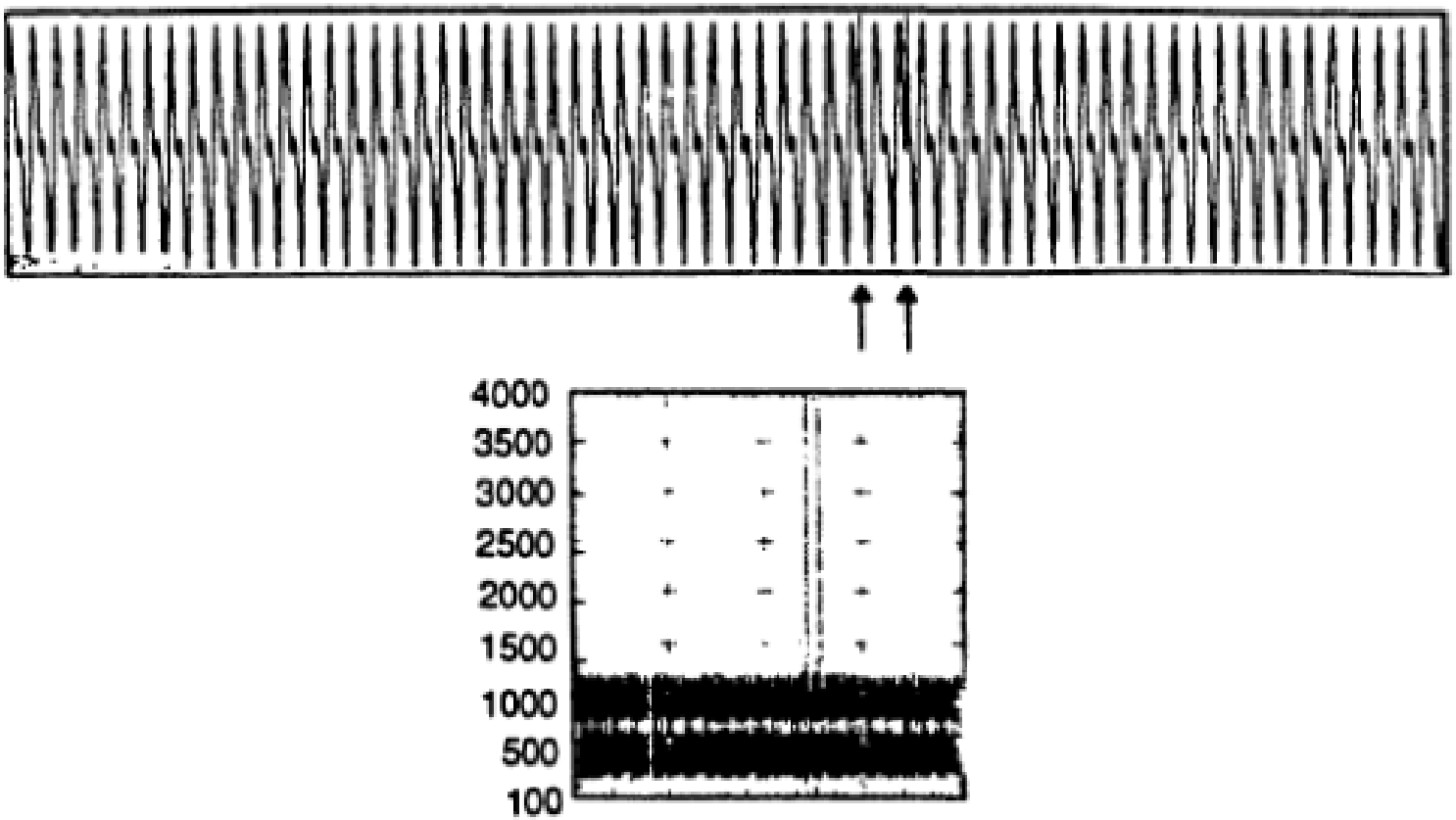
Gabor versus wavelet analysis



Top: Gabor analysis uses a fixed time window with variable number of oscillations. Bottom: Wavelet analysis uses large time windows for low frequencies and small time windows for high frequencies.

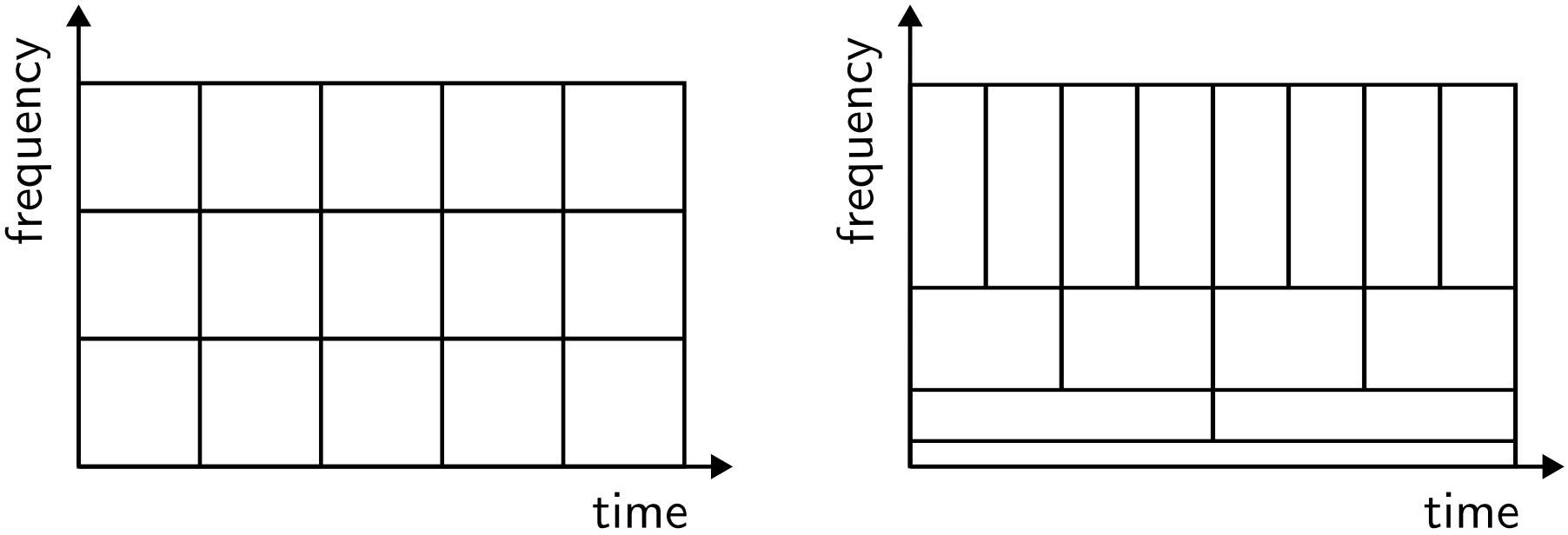
[Burke Hubbard]

Advantage of wavelet analysis



Top: A periodic signal consisting of 2 base frequencies with 2 pulses.
Bottom: Wavelets correctly distinguish the base frequencies and the pulses.

Tiling of the time–frequency domain



Left: Gabor frames define a uniform tiling.
Right: Wavelet frames define a non-uniform tiling.
Each frame element is concentrated in one of the rectangles.

Heisenberg's uncertainty principle

- There are myriads of wavelets ψ with various pros and cons.
- Ideally, one would like ψ to be localized in time and frequency, but by Heisenberg's uncertainty principle, there are limits to this:

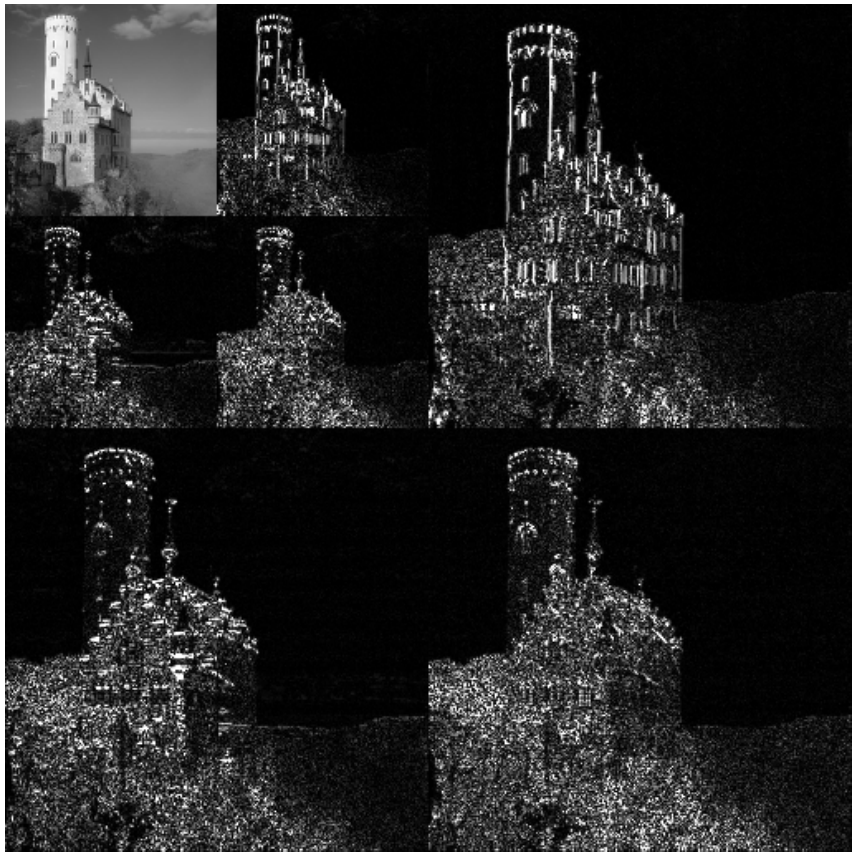
Theorem

For any $\psi \in L^2(\mathbb{R})$ with $\|\psi\|_{L^2(\mathbb{R})} = 1$,

$$\min_{x_0 \in \mathbb{R}} \|(x - x_0)\psi(x)\|^2_{L^2(\mathbb{R})} \min_{\xi_0 \in \mathbb{R}} \|(\xi - \xi_0)\mathcal{F}\psi(\xi)\|^2_{L^2(\mathbb{R})} \geq \frac{1}{16\pi^2}.$$

Remark. In quantum physics, the left-hand side represents the variance of measuring the location times the variance of measuring the momentum.

Application in image compression



[en.wikipedia.org/wiki/JPEG_2000]

JPEG2000 uses Cohen–Daubechies–Feauveau wavelets. Top left: wavelet coefficients at scale $a = 1$. Neighboring squares: differences to scale $a = 1/2$. Neighboring squares: differences to scale $a = 1/4$.

Questions to answer for yourself / discuss with friends

- Repetition: Describe the wavelet group, its representation on $L^2(\mathbb{R})$, and wavelet frames.
- Discussion: Is wavelet analysis related to musical scores?
- Transfer: How would you implement wavelet analysis or synthesis using neural networks?

MH3520 Chapter 11

Part 4

Shearlet analysis

Shearing and parabolic dilation

Shearlets are anisotropic, i.e., direction-sensitive. It would be natural to use rotations, but these are problematic numerically and mathematically, and we replace them by shear transformations.

Example

Parabolic dilation is a representation of $\mathbb{R}^\times := \mathbb{R} \setminus \{0\}$ on $L^2(\mathbb{R}^2)$:

$$\mathbb{R}^\times \ni a \mapsto P_a \in GL(L^2(\mathbb{R}^2)), \quad P_a f(x) = |a|^{-3/4} f\Big(\begin{pmatrix} a & 0 \\ 0 & \sqrt[\pm]{a} \end{pmatrix}^{-1} x\Big),$$

where $\sqrt[\pm]{a} := -\operatorname{sign}(a)\sqrt{|a|}$ is the signed square root.

Example

Shearing is a representation of $\mathbb{R}$ on $L^2(\mathbb{R}^2)$:

$$\mathbb{R} \ni b \mapsto S_b \in GL(L^2(\mathbb{R})), \qquad S_b f(x) = f\Big(\begin{pmatrix} 1 & b \\ 0 & 1 \end{pmatrix}^{-1} x\Big).$$

Shear group and its representation

We bundle up parabolic dilations, shear transformations, and translations into a group with representation on $L^2(\mathbb{R}^2)$.

Definition

The shear group is the set $G := \mathbb{R}^\times \times \mathbb{R} \times \mathbb{R}^2$ equipped with the multiplication

$$(a_1, b_1, c_1) * (a_2, b_2, c_2) := \left(a_1 a_2, b_1 + b_2\sqrt{|a_1|}, c_1 + \left(\begin{smallmatrix} 1 & b_1 \\ 0 & 1 \end{smallmatrix}\right)\left(\begin{smallmatrix} a & 0 \\ 0 & \pm\sqrt{a} \end{smallmatrix}\right) c_2\right).$$

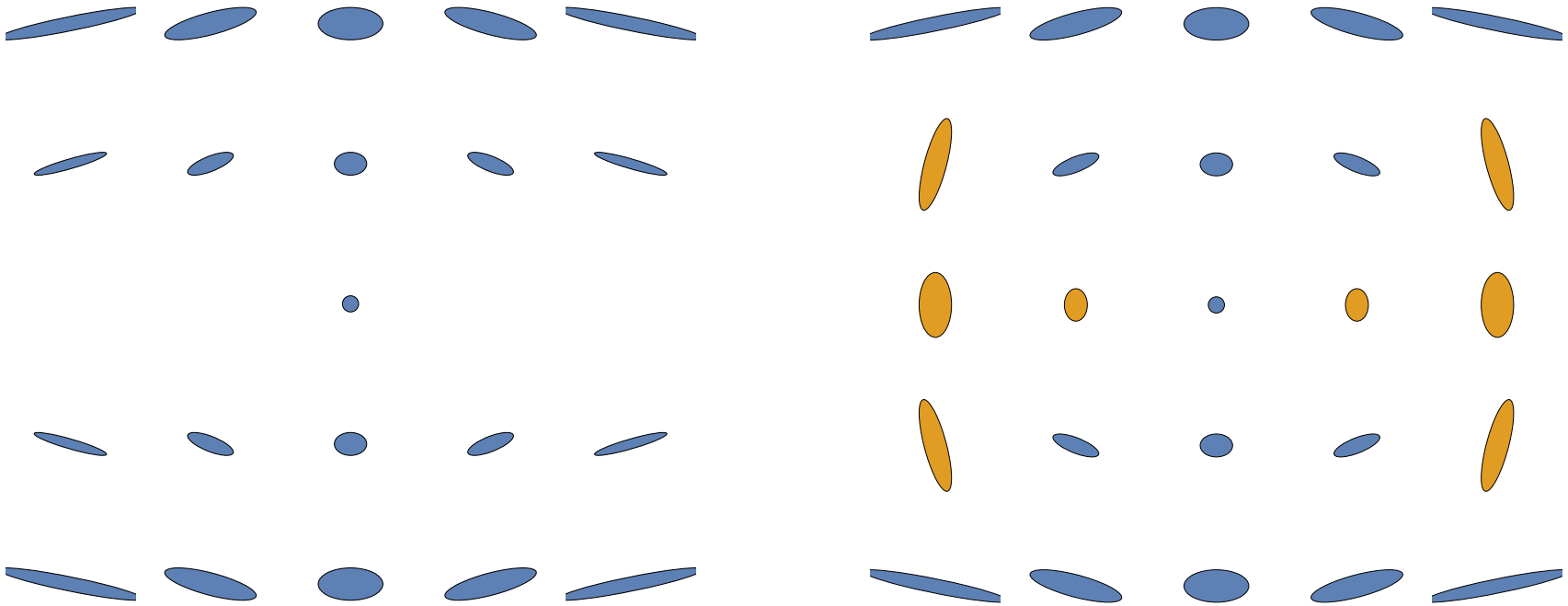
Definition

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

$$\pi(a, b, c) f = T_c S_b P_a f,$$

for parabolic dilations P_a, shear transformations S_b, and translations T_c.

Shearlet frames



Left: Parabolic dilations and shear transformations of a circle; certain angles are missing. Right: Filling in the missing angles using rotation by 90 degrees (orange).

Shearlet frames

We write $R_{\pi/2}$ for rotation by 90 degrees.

Theorem

Under certain conditions on ψ, ψ_1, ψ_2, for sufficiently small $\alpha > 1$ and $\beta, \gamma > 0$, the classical shearlet dictionary given by

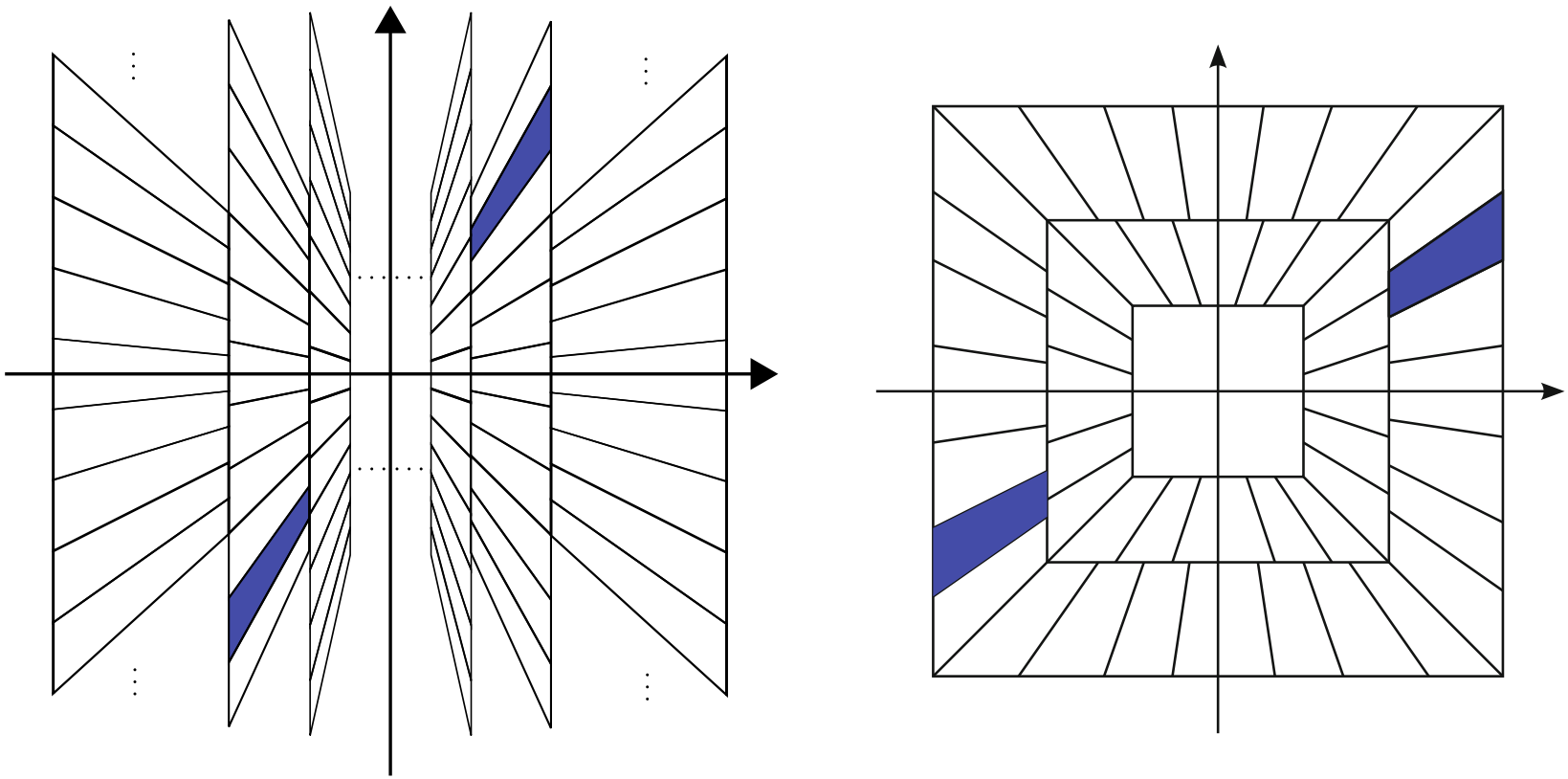
$$P_a S_b T_c \psi, \qquad a \in \pm\alpha^{\mathbb{Z}}, \quad b \in \sqrt{|a|}\beta\mathbb{Z}, \quad c \in \gamma\mathbb{Z}^d,$$

and the cone-adapted shearlet dictionary given by

$$T_c\psi_1,\ P_a S_b T_c \psi_2,\ R_{\pi/2} P_a S_b T_c \psi_2, \qquad a \in \pm\alpha^{\mathbb{N}_0},\ b \in \sqrt{|a|}\beta\mathbb{Z},\ c \in \gamma\mathbb{Z}^d,$$

are Hilbert frames for $L^2(\mathbb{R}^2)$

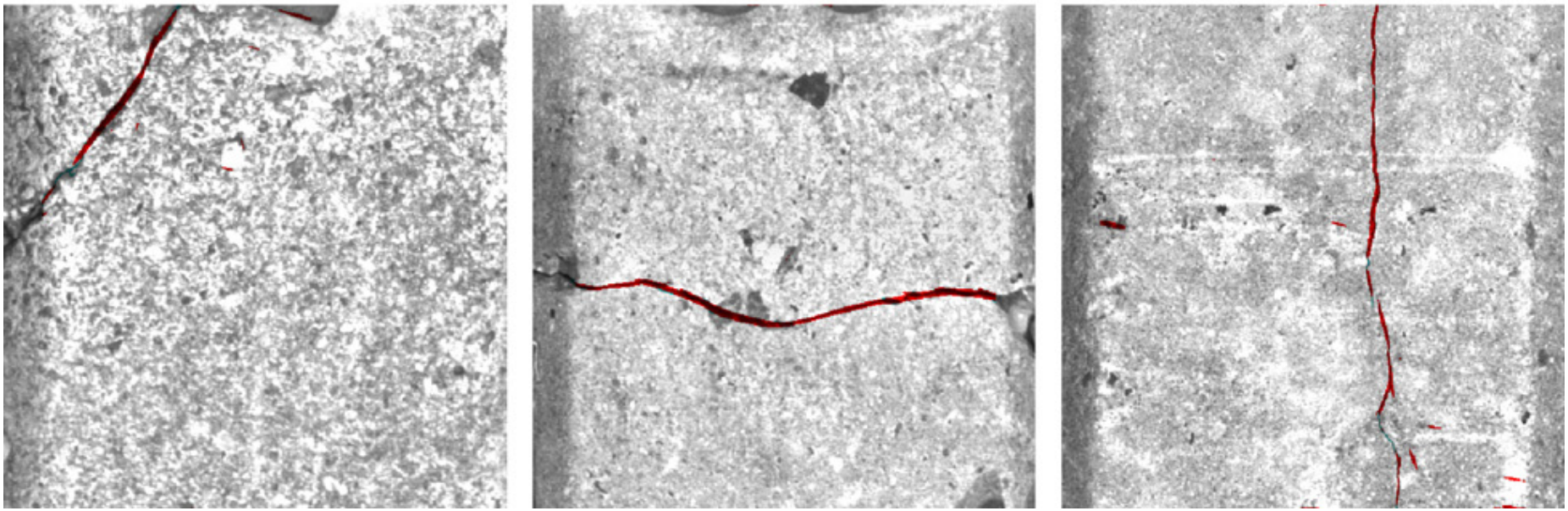
Shearlet tiling of the frequency domain



[Dahlke e.a. (2015)]

Left: classical shearlet tiling of the frequency domain.
Right: cone-adapted shearlet tiling of the frequency domain.

Application of shearlets for edge detection



[Gibert (2014)]

Shearlet coefficients concentrate along edge-like discontinuities of the signal. This can be used for edge detection.

Questions to answer for yourself / discuss with friends

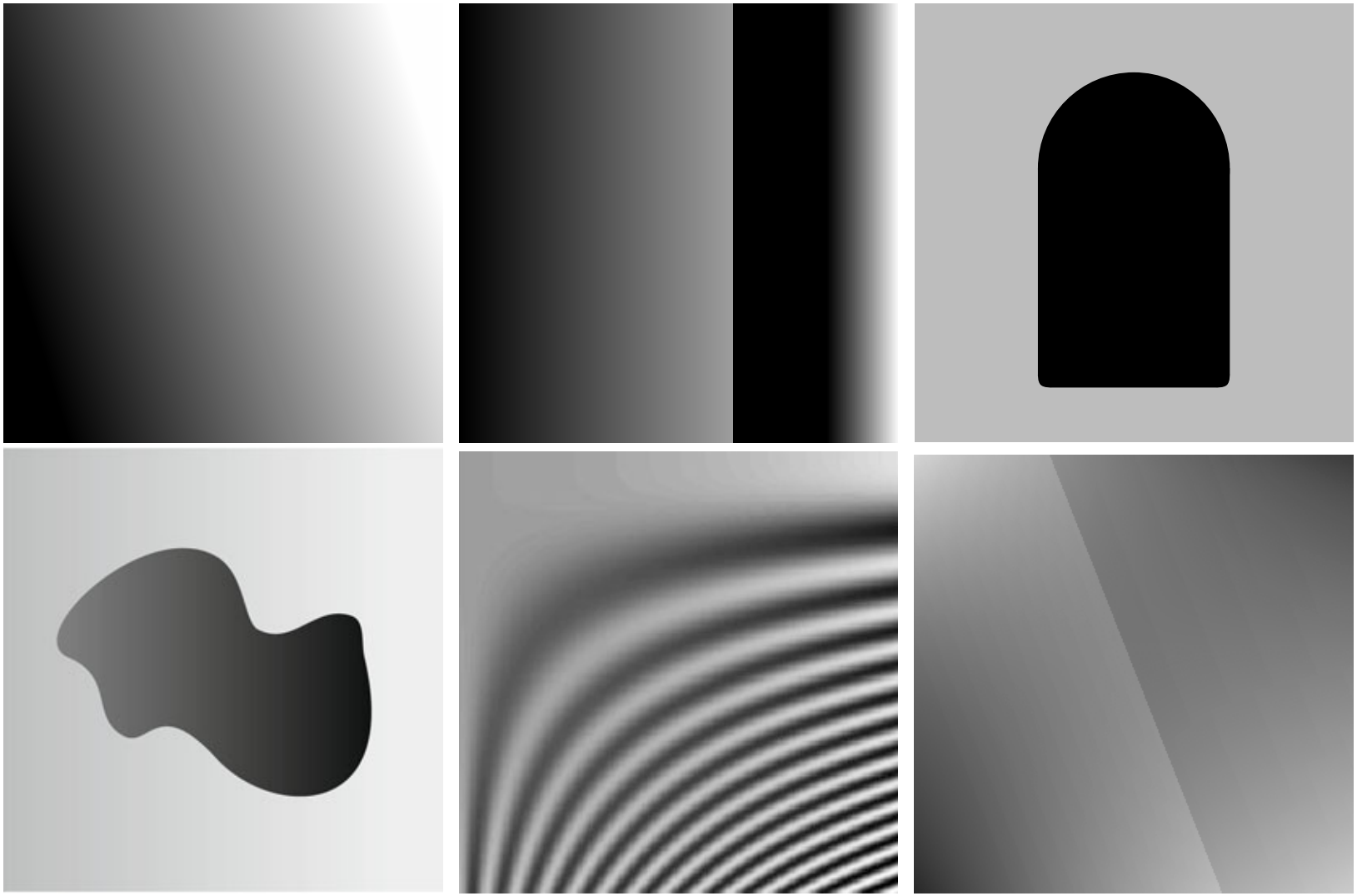
- Repetition: Describe the shearing group and its representation.
- Check your understanding: Are shearlets directional wavelets? In what sense?
- Transfer: How would you implement shearlet analysis or synthesis using neural networks?

MH3520 Chapter 11

Part 5

Signal classes and their approximation rates

Signal classes



C^k functions, piecewise C^k functions, star-shaped images, cartoon images, textures, mutilated functions, etc.

Signal classes

Generally speaking, these are relatively compact subsets of $L^2(\mathbb{R}^d)$.

Example (C^k functions)

$C_K^k([-1,1]^d)$ denotes C^k functions f on $[-1,1]^d$ with $\|f\|_{C^k(C)} \leq K$.

Example (Piecewise C^k functions)

$C_K^{k,\mathrm{pw}}([-1,1])$ denotes functions $f\mathbb{1}_{[0,c)} + g\mathbb{1}_{[c,1]}$, where $c \in (0,1)$ and $f, g \in C_K^k([-1,1])$.

Example (Star-shaped images)

$\mathrm{STAR}_K^k([-1,1]^2)$ denotes the set of indicator functions $\mathbb{1}_B$, whose boundary ∂B consists of closed regular Jordan curves of the form

$$B = \big\{(x + r\cos\phi, y + r\sin\phi) \colon r \leq f(\phi)\big\},$$

where $(x,y) \in \mathbb{R}^2$, and $f \in C_K^k([0,2\pi])$.

Signal classes

Example (Cartoon images)

$\mathrm{CART}_K^k([-1,1]^2)$ denotes functions $f\mathbb{1}_B + g$, where $f, g \in C_K^k([-1,1]^2)$ and $1_B \in \mathrm{STAR}_K^k([-1,1]^2)$.

Example (Textures)

$\mathrm{TEXT}_{K,M}^k([-1,1]^2)$ denotes functions $x \mapsto \sin(Mf(x))g(x)$, where $f, g \in C_K^k([-1,1]^2)$.

Example (Mutilated functions)

$\mathrm{MUTIL}_K^k([-1,1]^d)$ denotes functions $x \mapsto f(\langle u, x\rangle)g(x)$, where $u \in \mathbb{R}^d$ with $\|u\| = 1$, $f \in C_K^{k,\mathrm{pw}}(\mathbb{R})$ and $g \in C_K^k([-1,1]^d)$.

Approximation rates via harmonic analysis

Theorem

The approximation rate $a^*(Y, X, \Sigma(\phi)) \geq \frac{k}{d}$ *holds in* $X = L^2([-1,1]^d)$ *for the following signal classes* Y *and dictionaries* ϕ*:*

- $C_K^k([-1,1]^d)$ *via wavelets, shearlets, etc.*
- $C_K^{k,\mathrm{pw}}([-1,1]^d)$ *with* $d = 1$ *via wavelets*
- $\mathrm{STAR}_K^k([-1,1]^d)$ *with* $d = k = 2$ *via curvelets and shearlets*
- $\mathrm{CART}_K^k([-1,1]^d)$ *with* $d = k = 2$ *via curvelets and shearlets*
- $\mathrm{TEXT}_{K,M}^k([-1,1]^d)$ *with* $d = 2$ *via wave atoms*
- $\mathrm{MUTIL}_K^k([-1,1]^d)$ *via ridgelets*

Remark.

- The lower bound is always k/d, as if there were no singularities!
- Wave atoms and ridgelets are similar to wavelets and shearlets but scale differently in space and frequency.

Approximation rates via harmonic analysis

Sketch of proof.

- Away from the singularities, the coefficients decay at a fast rate.
- At the singularities, they decay at a slower rate, but there are not many singular points.
- Sparse approximations are defined by thresholding, i.e., picking the n largest coefficients among the first $\pi(n)$ coefficients.

Remark. The thresholded coefficients are bounded:

Corollary

In the previous theorem, $\Sigma(\phi)$ may be replaced by $\Sigma^{\pi,M}(\phi)$, defined as

$$\Sigma_n^{\pi,M}(\phi) := \left\{ \sum_{i=1}^{\pi(n)} c_i \phi_i : c_i \neq 0 \text{ at most } n \text{ times, } |c_i| \leq M \right\},$$

for some polynomial π and coefficient bound M.

Signal synthesis via deep learning

- The above-mentioned frames are affine transformations of some generating functions ψ.
- The generating functions ψ can be approximated by networks with finite width and depth.
- By the dictionary learning theorem, the dictionary approximation rates translate into network approximation rates.

Corollary

The same rates hold for $\mathtt{W}(X, \rho, L)$ in place of $\Sigma(\phi)$ if ϕ consists of affine transformations of generating function(s) ψ which satisfy

$$\psi \in \overline{\mathtt{W}_W(X, \rho, L)}, \qquad \text{for some } L, W \in \mathbb{N}.$$

Example

For wavelets and shearlets, this holds true if ρ is smooth and coincides with the ReLU function strictly away from zero. In $d = 2$, $L = 3$ suffices.

Signal analysis via deep learning

- The analysis operator is given by a convolutional matrix, which is sparse if the mother wavelet has compact support.
- Extraction of the n most significant coefficients (aka. thresholding) can be implemented by a max-pooling layer.
- Thus, any signal that is well analyzed by wavelets, shearlets, or other affine systems is well analyzed by networks with convolutional and max-pooling layers.
- Neural networks are more flexible because they combine the analytic capabilities of all affine systems.

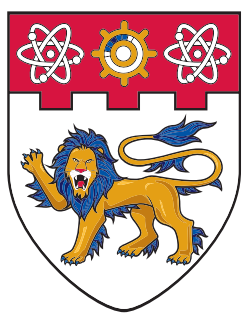
Questions to answer for yourself / discuss with friends

- Repetition: Recall some of the signal classes. How can you memorize their approximation rates?
- Check your understanding: The indicator function of the unit ball in $\mathbb{R}^2$, at what rate can it be approximated by 3-layer perceptrons?
- Discussion: In practice, to what extent do you think deep learning actually performs harmonic analysis? What about synthesis?

MH3520:Mathematics of Deep Learning

Chapter 12

Coding theory and best approximation rates



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Context

Last chapters:

- If a function can be approximated by splines or wavelets, it can be approximated at least as well by neural networks.

This chapter:

- Memory-constrained networks are function encoders, and encoders have information-theoretic limits on how well they can perform.
- For many signal classes, memory-constrained networks are rate-optimal (as are certain dictionaries from harmonic analysis).

Overview of Chapter 12

1. Rate–distortion theory
2. Upper bounds on encoding rates
3. Lower bounds on encoding rates
4. Upper bound on network approximation rates
5. Lower bound on network approximation rates

Overview of coding theory and rate-optimal approximation

Notation

- k denotes a smoothness parameter, and d the input dimension.

Main result

- Dictionary & network approximation rates = encoding rate = $\frac{k}{d}$.

Rate-optimal approximation by dictionaries

- Part 2: Many signal classes contain hypercubes, which are difficult to encode. This entails an upper bound of $\frac{k}{d}$ on the encoding rate.
- Part 3: Harmonic analysis allows one to encode the signals in a string of coefficients, resulting in a lower bound of $\frac{k}{d}$ on the encoding rate.

Rate-optimal approximation by networks

- Part 4: As memory-constrained networks are encoders, the network approximation rate has the information-theoretic upper bound $\frac{k}{d}$.
- Part 5: Dictionary approximation with bounded coefficients (as in thresholding) can be implemented by memory-constrained networks. Hence, the network approximation rate is lower-bounded by $\frac{k}{d}$.

Sources for this chapter:

- Dahlke, De Mari, Grohs, Labate (2015): Harmonic and applied analysis—from groups to signals. Birkhäuser.
- Bölcskei, Grohs, Kutyniok, and Petersen (2019): Optimal approximation with sparsely connected deep neural networks. SIAM Journal of Mathematical Data Science 1(1), pp. 8–45.

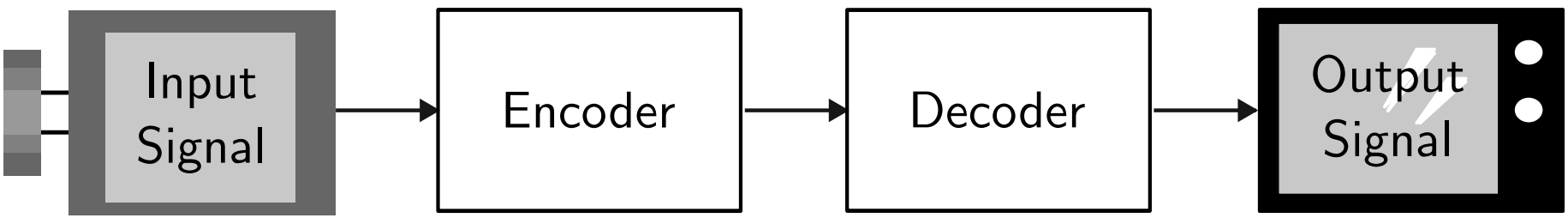
MH3520 Chapter 12

Part 1

Rate–distortion theory

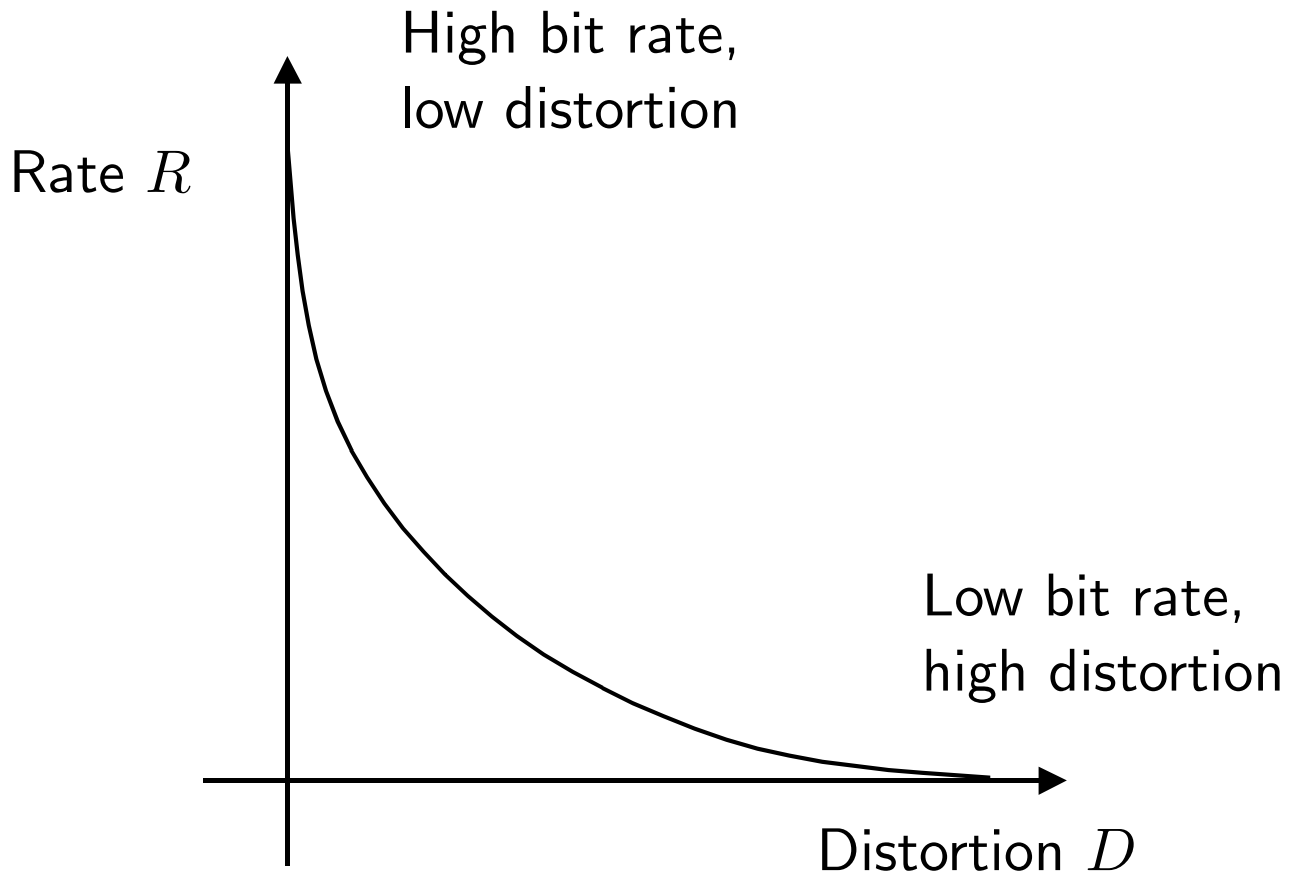
Overview of rate–distortion theory

- Rate–distortion theory is a major branch of information theory and coding theory.
- It provides mathematical foundations for lossy data compression.
- Specifically, it investigates the fundamental tradeoff between the transmission bit rate and the signal distortion.
- The results are independent of any specific coding method.



[Girod]

Rate–distortion curve



[Girod]

The rate–distortion curve consists of pairs of minimal bit rates for maximally admissible signal distortions.

Encoding, decoding, and distortion

Definition

Let X be a normed space, let $Y \subseteq X$ be a signal class, and let $n \in \mathbb{N}$.

- The set of binary encoders of Y with run-length n is defined as

$$\mathcal{E}^n := \{E : Y \to \{0,1\}^n\}.$$

- The set of binary decoders with run-length n is defined as

$$\mathcal{D}^n := \{D : \{0,1\}^n \to X\}.$$

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

$$\delta(E, D) := \sup_{f \in Y} \|f - D(E(f))\|_X .$$

Remark: More generally, X could be a quasi-normed or metric space.

Encoding rate

Definition

The encoding rate of a signal class Y in a normed space X is defined as

$$e^*(Y, X) := \sup \left\{ r > 0 \middle| \inf_{(E,D) \in \mathcal{E}^n \times \mathcal{D}^n} \delta(E, D) = O(n^{-r}) \right\}.$$

Remark:

- The encoding rate quantifies the complexity of a signal class in an information-theoretic way.
- For any $r < e^*(Y, X)$, one can compress signals $f \in Y$ using n-bit encodings with distortion proportional to n^{-r}.

Examples

Example

Any set Y of cardinality n can be encoded with $\lceil \log_2(n) \rceil$ bits and zero distortion. The encoding rate is infinite.

Remark. The above example is the most important of all: as all other rates are polynomial, a logarithmic rate is extremely slow.

Example

The interval $Y := [0, 1]$ in $X := \mathbb{R}$ can be encoded with n bits and distortion 2^{-n}. As the distortion decays exponentially in n, $e^*(Y, X) = \infty$.

Example

Any set Y with positive encoding rate is totally bounded, i.e., it can be covered by finitely many balls of any given radius. Hence, Y is relatively compact, i.e., its closure in X is compact.

No compression without distortion

Definition

The Hamming distance between two bit streams $\xi, \eta \in \{0,1\}^n$ is

$$d(\xi,\eta) := \|\xi - \eta\|_0 := \#\{i : \xi_i \neq \eta_i\}.$$

The Hamming distortion is the distortion of bit streams measured in the Hamming distance.

Lemma

For any compression rate $R < 1$, there exists $C > 0$ such that any encoder-decoder pair with sufficiently high run-lengths $n \leq Rm$,

$$E\colon \{0,1\}^m \to \{0,1\}^n, \qquad D\colon \{0,1\}^n \to \{0,1\}^m,$$

has Hamming distortion greater than or equal to Cm.

The proof is elementary; see [Dahlke e.a., Lemma 5.14].

Relation to covering numbers

Encoding rates are closely related to covering numbers, which we have already encountered in statistical learning theory.

Definition

Let Y be a relatively compact metric space.

- The covering number $\mathcal{N}(Y, \epsilon)$ is the minimal number $N \in \mathbb{N}$ such that N discs of radius ϵ cover all of Y.
- The Kolmogorov entropy is defined as $H_\epsilon(Y) := \log_2 \mathcal{N}(Y, \epsilon)$.

Relation to covering numbers

Lemma

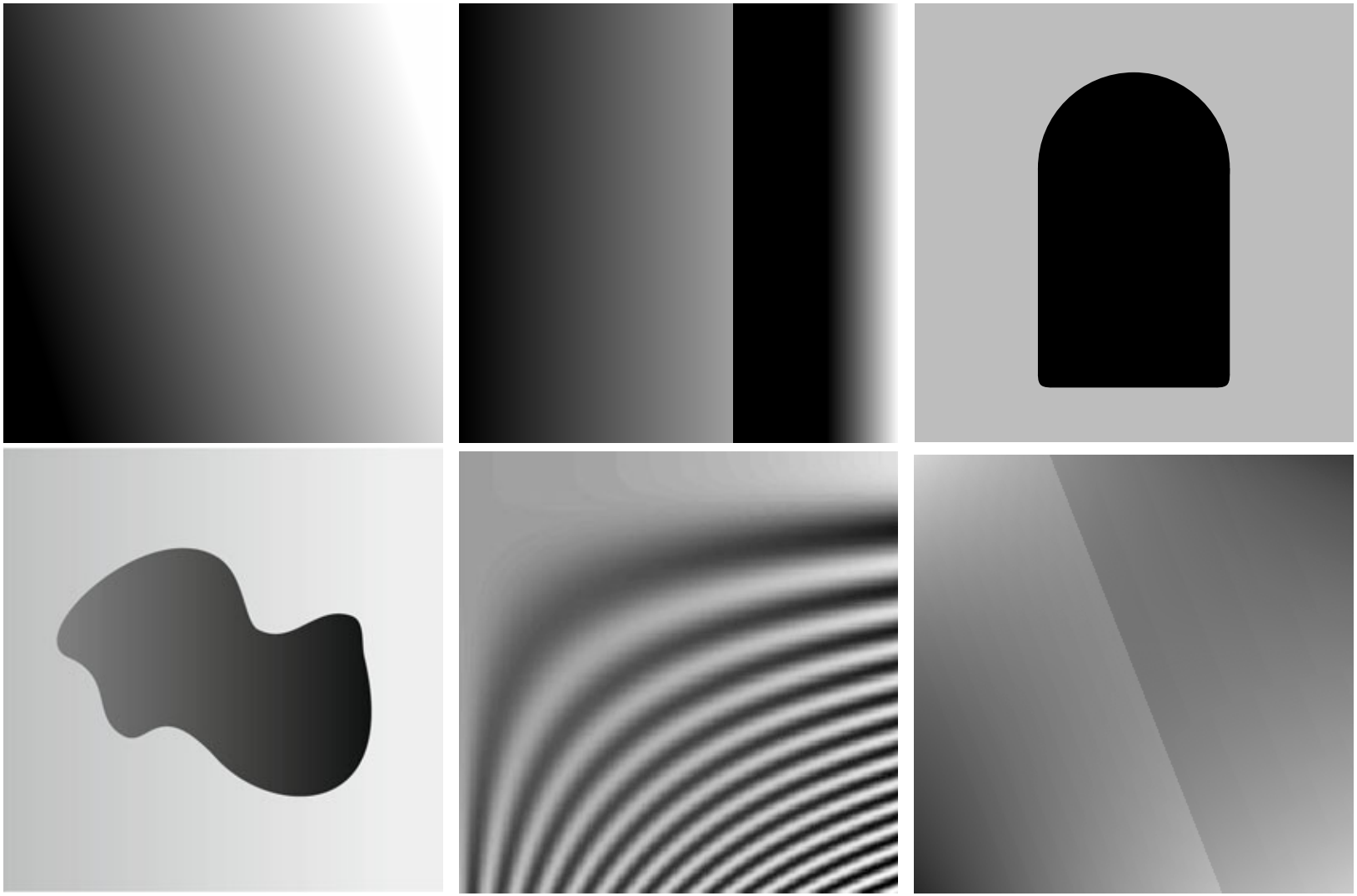
The encoding rate is related to the Kolmogorov entropy (and hence to covering numbers) by

$$e^*(Y, X) = \sup \left\{ r > 0 : H_\epsilon(Y) = O(\epsilon^{-\frac{1}{r}}) \right\}.$$

Proof:

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

Signal classes



C^k functions, piecewise C^k functions, star-shaped images, cartoon images, textures, mutilated functions, etc.

Signal classes and encoding rates

Coding theory and harmonic analysis work well together and yield sharp upper and lower bounds on encoding and approximation rates:

Theorem

The encoding rate $e^*(Y,X) = \frac{k}{d}$ *holds in* $X := L^2([-1,1]^d)$ *for the following signal classes* Y*:*

- $C_K^k([-1,1]^d)$
- $\mathrm{STAR}_K^k([-1,1]^d)$ *for* $d = k = 2$
- $\mathrm{TEXT}_{K,M}^k([-1,1]^d)$ *for* $d = 2$
- $C_K^{k,pw}([-1,1]^d)$ *for* $d = 1$
- $\mathrm{CART}_K^k([-1,1]^d)$ *for* $d = 2$
- $\mathrm{MUTIL}_K^k([-1,1]^d)$

Proof.

- $e^*(Y,X) \leq \frac{k}{d}$: see proposition in Part 2, via hypercube embeddings.
- $e^*(Y,X) \geq \frac{k}{d}$: see proposition in Part 3, via harmonic analysis. □

Memory-constrained neural networks

Memory-constrained networks can be seen as function encoders. The memory constraint is implemented by bounding the network's coefficients:

Definition

$\mathtt{W}_n^M(X, \rho, L)$ denotes the realizations of networks with coefficients bounded in absolute value by M, at most n non-zero weights, activation function ρ, and L layers, seen as elements of the function space X.

- Careful: These hypothesis classes are not closed under scalar multiplication. Hence, approximation spaces are ill defined.
- Notation: If μ is a polynomial, then $\mathtt{W}^\mu(X, \rho, L)$ denotes the sequence of hypothesis classes $\mathtt{W}_n^{\mu(n)}(X, \rho, L)$, $n \in \mathbb{N}$.

Signal classes and network approximation rates

Neural networks with polynomially bounded coefficients turn out to be rate-optimal for all of our signal classes:

Theorem

The network approximation rate $a^*(Y, X, \mathtt{W}^\mu(X, \rho, L)) = \frac{k}{d}$ *holds in* $X := L^2([-1,1]^d)$ *if* ρ *is a smoothened ReLU function,* L *is sufficiently large, and* μ *is a suitable polynomial, for the following signal classes* Y*:*

- $C_K^k([-1,1]^d)$
- $\mathrm{STAR}_K^k([-1,1]^d)$ *for* $d = k = 2$
- $\mathrm{TEXT}_{K,M}^k([-1,1]^d)$ *for* $d = 2$
- $C_K^{k,pw}([-1,1]^d)$ *for* $d = 1$
- $\mathrm{CART}_K^k([-1,1]^d)$ *for* $d = 2$
- $\mathrm{MUTIL}_K^k([-1,1]^d)$

Proof.

- $a^*(Y, X, \mathtt{W}^\mu(X, \rho, L)) \leq \frac{k}{d}$: see the proposition in Part 4, by encoding networks.
- $a^*(Y, X, \mathtt{W}^\mu(X, \rho, L)) \geq \frac{k}{d}$: see the proposition in Part 5, via harmonic analysis. □

Questions to answer for yourself / discuss with friends

- Repetition: What is an endoding-decoding pair, and how are encoding rates defined?
- Repetition: How many bits are needed to encode a natural number in $\{1, \ldots, n\}$?
- Check your understanding: How many bits are needed to encode the set $\{f\}$, where f is a Lipschitz function on $[-1, 1]^d$?
- Transfer: Some compression algorithms such as `zip` or `flac` are called lossless. What could be meant by this?

MH3520 Chapter 12

Part 2

Upper bounds on encoding rates

Hypercubes

Definition

An m-dimensional hypercube in a Hilbert space is a set

$$\sum_{i=1}^{m} \epsilon_i \psi_i, \qquad \epsilon_i \in \{0, 1\},$$

with mutually orthogonal sides ψ_i of equal length.

Remark. Hypercubes are particularly difficult to encode: whenever two vertices have the same encoding, the distortion is at least the side length.

Example

Let Y be the ℓ^p unit ball in $X := \ell^2$, for some $p \leq 2$. Then, Y contains for any $m \in \mathbb{N}$ an m-dimensional hypercube with sides of length $m^{-1/p}$:

$$\psi_i := m^{-1/p} \mathbb{1}_{A_i}, \qquad \#A_i = m, \quad A_i \cap A_j = \emptyset.$$

Hypercube embeddings

Definition

A signal class Y in a Hilbert space X has embedded ℓ^p hypercubes if it contains hypercubes of dimension m and side length $\geq Cm^{-1/p}$, for some constant $C > 0$ and some sequence of dimensions m tending to infinity.

Theorem

If Y has embedded ℓ^p hypercubes for some $p \in (0, 2]$, then

$$e^*(Y, X) \leq \frac{1}{p} - \frac{1}{2}.$$

Proof: hypercube embeddings

Proof.

- Consider an encoder-decoder pair with run-length n.
- This yields an encoder-decoder pair for any embedded hypercube (restrict the encoder's input and project the decoder's output).
- In turn, this yields an encoder-decoder pair for any bitstream of run-length m (identify bitstreams with hypercubes).
- Fix the compression rate $R := 1/3$ and set $m := n/R$.
- Then, the Hamming distortion of bitstreams is proportional to n or higher, i.e. at least that many bits are wrong.
- The squared distortion of the hypercube encoder-decoder is, by Pythagoras, the number of wrong bits times the squared side length, i.e., proportional to $n * n^{-2/p}$ or higher. Accordingly, the distortion is proportional to $n^{1/2-1/p}$ or higher.
- For the encoder-decoder without restriction and projection, the distortion is only higher.

□

Signal classes and encoding rates

Proposition

The upper bound $e^*(Y, X) \leq \frac{k}{d}$ *holds in* $X = L^2([-1,1]^d)$ *for the following signal classes* Y *via embedding of* $\ell^{1/(\frac{k}{d}+\frac{1}{2})}$ *hypercubes:*

- $C_K^k([-1,1]^d)$
- $\mathrm{STAR}_K^k([-1,1]^d)$ *for* $d = k = 2$
- $\mathrm{TEXT}_{K,M}^k([-1,1]^d)$ *for* $d = 2$
- $C_K^{k,pw}([-1,1]^d)$ *for* $d = 1$
- $\mathrm{CART}_K^k([-1,1]^d)$ *for* $d = 2$
- $\mathrm{MUTIL}_K^k([-1,1]^d)$

Proof: see next slide.

Proof: hypercube embeddings for signal classes

Proof for $C_K^k([-1,1]^d)$:

- Choose $\psi \geq 0$ with support in $[-1,1]^d$ and $\|\psi\|_{C^k} = K$.
- The m^d-dimensional hypercube with orthogonal sides

$$\psi_i(x) := m^{-k}\psi(mx - i), \qquad i \in \{0, \ldots, m-1\}^d$$

  is contained in $C_K^k([-1,1]^d)$ (why?) and has side-length

$$\|\psi_i\|_{L^2([-1,1])} = m^{-k-d/2}.$$

- Thus, $C_K^k([-1,1])$ has embedded $\ell^{1/(\frac{k}{d}+\frac{1}{2})}$ hypercubes. □

Remark. For other signal classes, one uses the following hypercubes:

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

Questions to answer for yourself / discuss with friends

- Repetition: How are upper bounds on the encoding rate obtained from hypercube embeddings?
- Check your understanding: What is the encoding rate of the unit ball Y in X?
- Check your understanding: $\partial_x^k f(ax) = \dots$ and $\sqrt{\int f(ax)^2\,dx} = \dots$?
- Discussion: Could we use other function classes (e.g. linear or Lipschitz functions) in the definition of encoders and decoders? What kind of coding theory would we get from this?

MH3520 Chapter 12

Part 3

Lower bounds on encoding rates

Dictionary approximation with bounded coefficients

Motivation.

- Dictionaries can be used as encoders and decoders.
- For encoding, one stores the coefficients.
- For decoding, one multiplies with the dictionary.
- For this to work well, the coefficients must be bounded.

Definition

Sparse approximation with polynomial-depth search and bounded coefficients uses the hypothesis classes

$$\Sigma_n^{\pi,M}(\phi) := \left\{ \sum_{i=1}^{\pi(n)} c_i \phi_i : c_i \neq 0 \text{ at most } n \text{ times, } |c_i| \leq M \right\},$$

for some polynomial π and constant M.

Careful: These hypothesis classes are not closed under scalar multiplication!

Encoding and decoding using dictionaries

As dictionaries are encoder-decoder pairs, the dictionary approximation rate can't be higher than the encoding rate:

Theorem

For any signal class Y and a bounded dictionary ϕ in X, polynomial π, and $M \in \mathbb{N}$,

$$e^*(Y, X) \geq a^*(Y, X, \Sigma^{\pi, M}(\phi)).$$

Remark.

- A dictionary ϕ is called rate-optimal for Y if equality holds above.
- The boundedness constraint on the dictionary and the dictionary coefficients can be omitted if ϕ is a Hilbert frame and Y is bounded.
- However, boundedness is needed in one way or another for encoding.

Proof: encoding using dictionaries

Proof:

- We start by constructing an encoder. For any $r < a^*(Y, X, \Sigma^{\pi,M}(\phi))$, there exists a constant $C > 0$ such that for all $n \in \mathbb{N}$ and $f \in Y$, there exist coefficients $c_i \in \mathbb{R}$ with $\|c\|_0 \leq n$ such that

$$\left\| f - \sum_{i=1}^{\pi(n)} c_i \phi_i \right\|_X \leq C n^{-r} .$$

- The set $\Lambda := \{i \in \mathbb{N} : c_i \neq 0\}$ can be encoded using $n\lceil \log_2(\pi(n)) \rceil$ bits, which is $O(n \log n)$.
- Rounding the non-zero coefficients c_i up to multiples of n^{-r-1} encodes them with a bit string of length $O(n \log n)$.
- Altogether, this gives an encoding $E : Y \to \{0,1\}^l$ with run-length $l = O(n \log n)$.

Proof: decoding using dictionaries

- Decoding is done by reversing this process: starting from a bit string ξ, reconstruct the set Λ and the rounded approximations $\hat{c}_i$ of the non-zero coefficients c_i, and define the decoder

$$D\colon \{0,1\}^l \to X, \qquad D(\xi) := \sum_{i\in\Lambda} \hat{c}_i \phi_i.$$

- It remains to control the distortion: with $K := \sup_i \|\phi_i\|_X$, one has

$$\begin{aligned}
\|f - D(E(f))\|_X &= \left\| f - \sum_{i\in\Lambda} \hat{c}_\lambda \phi_i \right\|_X \\
&\le \left\| f - \sum_{i\in\Lambda} c_i \phi_i \right\|_X + \left\| \sum_{i\in\Lambda} (\hat{c}_i - c_i)\phi_i \right\|_X \\
&\le C n^{-r} + K n^{-r-1} n = O(n^{-r}).
\end{aligned}$$

□

Signal classes and encoding rates

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

Proposition

The lower bound $e^*(Y, X) \geq \frac{k}{d}$ *holds in* $X = L^2([-1,1]^d)$ *for the following signal classes* Y*:*

- $C_K^k([-1,1]^d)$ *via wavelets, shearlets, and many more*
- $C_K^{k,pw}([-1,1]^d)$ *for* $d = 1$ *via wavelets*
- $\mathrm{STAR}_K^k([-1,1]^d)$ *for* $d = k = 2$ *via curvelets and shearlets*
- $\mathrm{CART}_K^2([-1,1]^d)$ *for* $d = k = 2$ *via curvelets and shearlets*
- $\mathrm{TEXT}_{K,M}^k([-1,1]^d)$ *for* $d = 2$ *via wave atoms*
- $\mathrm{MUTIL}_K^k([-1,1]^d)$ *via ridgelets*

Proof: $e^*(Y, X) \geq a^*(Y, X, \Sigma^{\pi,M}(\phi))$, and the indicated dictionary ϕ achieves the indicated rate on the signal class Y for suitable π, M. □

Rate-optimal approximation by dictionaries

Corollary

In each case of the previous proposition, the indicated dictionary ϕ is rate-optimal for the indicated signal class Y, i.e.,

$$e^*(Y, X) = a^*(Y, X, \Sigma^{\pi,M}(\phi)).$$

Proof: The following rates are equal,

$$\frac{k}{d} \overset{1}{=} a^*(Y, X, \Sigma^{\pi,M}(\phi)) \overset{2}{\leq} e^*(Y, X) \overset{3}{\leq} \frac{k}{d},$$

where

1. holds thanks to harmonic analysis,
2. holds because dictionaries can be used as encoders, and
3. holds thanks to hypercube embeddings. □

Limits on dictionary approximation

Here is what it means that the dictionary approximation rate cannot be higher than the encoding rate: it is an existence result for 'bad' signals.

Corollary

If $\alpha > e^*(Y, X)$ and $C > 0$, there exists a signal $f \in Y$ which cannot be approximated by $\Sigma_n^{\pi(n),M}(\phi)$ with error bounded by $Cn^{-\alpha}$, i.e.,

$$\sup_{n\in\mathbb{N}} n^{\alpha} E_{n-1}(f, X, \Sigma^{\pi,M}(\phi)) > C.$$

Remark.

- In general, we cannot find a signal f whose approximation error fails to be $O(n^{-\alpha})$, without fixed proportionality constant.
- For example, all signals in Y might have approximation errors in $O(n^{-\alpha})$ but with arbitrarily bad proportionality constants C.

Questions to answer for yourself / discuss with friends

- Repetition: How are lower bounds on encoding rates obtained from dictionary approximation rates?
- Check your understanding: Dictionaries are function encoders: what goes wrong if polynomial-depth search of boundedness or boundedness of the coefficients are omitted?
- Check your understanding: If the signal class is bounded, does this imply that the dictionary coefficients are bounded?
- Check your understanding: Under what circumstances does thresholding lead to bounded coefficients?

MH3520 Chapter 12

Part 4

Upper bound on network approximation rates

Quantization of neural networks

- For encoding, the network's coefficients, which are assumed to be bounded, must be quantized.
- The quantization error of the perceptron is proportional to the quantization error of the coefficients, for Lipschitz activation functions.
- The proportionality constant admits a polynomial bound.

Lemma

For any $L \in \mathbb{N}$, there exists a polynomial π with the following property: if $X = C([-K, K]^d)$, ρ is an activation function with Lipschitz constant C, Φ is a neural network with realization in $\mathtt{W}_W^M(X, \rho, L)$, and $\tilde{\Phi}$ is obtained by rounding the coefficients of Φ to the nearest multiple of ϵ in $[-M, M]$, then

$$\|R(\Phi) - R(\tilde{\Phi})\|_X \leq \epsilon\, \pi(C, K, M, W).$$

Remark. Alternatively, one may assume that ρ is differentiable with derivative bounded in absolute value by some polynomial.

Proof: quantization of neural networks

Proof.

- We set $\Phi_\ell = ((A_\ell, b_\ell), \ldots, (A_1, b_1))$ and $\|f\| := \sup_x \max_i |f(x)_i|$.
- For the single-layer network $R(\Phi_1)(x) = A_1 x + b_1$,

$$\|R(\Phi_1)\| \leq KMW + MW, \quad \|R(\Phi_1) - R(\tilde{\Phi}_1)\| \leq \epsilon KW + \epsilon W.$$

- For the double-layer network $R(\Phi_2)(x) = A_2 \rho(A_1 x + b_1) + b_2$,

$$\|R(\Phi_2)\| \leq CMW\|R(\Phi_1)\| + MW,$$
$$\|R(\Phi_2) - R(\tilde{\Phi}_2)\| \leq \epsilon CW\|R(\Phi_1)\| + \epsilon W + CMW\|R(\Phi_1) - R(\tilde{\Phi}_1)\|.$$

- The statement follows by induction. □

Encoding via neural networks

Neural networks with bounded coefficients are memory-constrained, hence can be used as encoders. Therefore, the network approximation rate cannot exceed the encoding rate:

Theorem

Let Y be a signal class in $X := L^p([-1,1]^d)$ for some $p \in (0,\infty]$, let μ be a polynomial, let ρ be a Lipschitz activation function, and let $L \in \mathbb{N}$. Then,

$$a^*(Y, X, \mathtt{W}^\mu(X, \rho, L)) \leq e^*(Y, X).$$

Remark:

- Neural networks are called rate-optimal for Y if equality holds above.
- The theorem implies a lower bound on the network connectivity for any given approximation rate.
- Alternatively, ρ can also be differentiable with derivative bounded in absolute value by some polynomial.

Proof: encoding via neural networks

Proof:

- We approximate signals f by perceptrons $R(\Phi)$:
  - Let $r < a^*(Y, X, \mathtt{W}^\mu(X, \rho, L))$. Then, there exists $C > 0$ such that for any signal $f \in Y$ and $n \in \mathbb{N}$, there exists a network Φ with realization in $\mathtt{W}_n^{\mu(n)}(X, \rho, L)$ and approximation error $\|R(\Phi) - f\|_X \leq Cn^{-r}$.
- If necessary, we compress the network by deleting unused neurons:
  - This ensures that $B \leq W + N_L$, where B and W are the numbers of biases and weights, respectively, and N_L is the output dimension.
  - The realization of Φ is not affected, and we use the same letter Φ to denote the compressed network.
- We encode the architecture of Φ in a bit string:
  - We encode the number $W \leq n$ of non-zero weights in a string of W 1's, followed by a single 0.
  - Then, we encode the numbers $N_0, \ldots, N_L \leq 2W$ in a string of $(L+1)\lceil \log_2 W + 1 \rceil$ bits.

Proof: encoding via neural networks

- We encode the topology (aka. connectivity) of Φ in a bit string:
  - To each neuron, we assign a unique index $i \in \{1, \ldots, N\}$. This index can be encoded in a string b_i of $\lceil \log_2 W + 1 \rceil$ bits because $N \leq 2W$.
  - 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 $2\lceil \log_2 W + 1 \rceil$ to signal the transition to neuron $i + 1$.
- We quantize the network Φ to a network $\tilde{\Phi}$:
  - Rounding the coefficients of Φ to the nearest multiple of n^{-k} in $[-\mu(n), \mu(n)]$ yields a quantized network $\tilde{\Phi}$.
  - The previous lemma allows us to choose k, independently of Φ, such that $\|R(\Phi) - R(\tilde{\Phi})\|_X \leq Cn^{-r}$.
- We encode the coefficients of $\tilde{\Phi}$ in a bit string:
  - Each coefficient requires $\lceil \log_2(2n^k \mu(n)) \rceil = O(\log_2 n)$ bits.
  - There are $O(n)$ weights and biases.
- Overall, we have used $O(n \log_2 n)$ bits. Decoding is done by reversing the process and achieves distortion $O(n^{-r})$. Thus, $r \leq e^*(Y, X)$. $\square$

Limits on network approximation

Here is what it means that the network approximation rate cannot be higher than the encoding rate: it is an existence result for 'bad' signals.

Corollary

Under the assumptions of the previous theorem, there exists for any $\alpha > e^*(Y, X)$ *and* $C > 0$ *a signal* $f \in Y$ *which cannot be approximated by perceptrons in* $\mathtt{W}_n^{\mu(n)}(X, \rho, L)$ *with error bounded by* $Cn^{-\alpha}$*, i.e.,*

$$\sup_{n \in \mathbb{N}} n^{\alpha} E_{n-1}(f, X, \mathtt{W}^{\mu}(X, \rho, L)) > C.$$

Remark.

- In general, we cannot find a signal f whose approximation error fails to be $O(n^{-\alpha})$, without fixed proportionality constant.
- For example, all signals in Y might have approximation errors in $O(n^{-\alpha})$ but with arbitrarily bad proportionality constants C.

Signal classes and network approximation rates

Proposition

The upper bound $a^*(Y, X, \mathtt{W}^\mu(X, \rho, L) \leq \frac{k}{d}$ *holds in* $X = L^2([-1,1]^d)$ *for any polynomial* μ*, Lipschitz activation function* ρ*, and* $L \in \mathbb{N}$*:*

- $C_K^k([-1,1]^d)$
- $\mathrm{STAR}_K^k([-1,1]^d)$ *for* $d = k = 2$
- $\mathrm{TEXT}_{K,M}^k([-1,1]^d)$ *for* $d = 2$
- $C_K^{k,pw}([-1,1]^d)$ *for* $d = 1$
- $\mathrm{CART}_K^k([-1,1]^d)$ *for* $d = 2$
- $\mathrm{MUTIL}_K^k([-1,1]^d)$

Questions to answer for yourself / discuss with friends

- Repetition: Why is the network approximation rate upper-bounded by the encoding rate?
- Check your understanding: Why can the logarithmic factor $\log_2 n$ in the rate computation be ignored?
- Check your understanding: In the last proof, decoding is done by reversing the encoding process—what does this mean specifically?
- Discussion: In actual deep-learning implementations, are neural networks memory-constrained?
- Discussion: What does the result say about deep learning? What are limitations of the result?

MH3520 Chapter 12

Part 5

Lower bound on network approximation rates

Dictionary learning with bounded coefficients

Motivation.

- Loosely speaking, networks with bounded coefficients can implement dictionary approximation with bounded coefficients.

Standing assumption.

- Y is a signal class in a quasi-normed function space X.
- ρ is an activation function. L, W are numbers of layers and weights.
- $\phi = (\phi_i)_{i \in \mathbb{N}}$ is a dictionary in X.

Remark.

- The hypothesis classes $\Sigma^{\pi, M}(\phi)$ and $\mathtt{W}^M(X, \rho, L)$ with coefficients bounded by M are not closed under scalar multiplication.
- Therefore, approximation spaces A^α are ill-defined, and we will formulate our dictionary learning theorem in terms of approximation rates a^*.

Dictionary learning with bounded coefficients

Theorem

Assume for some bivariate polynomial ν that

$$\forall i, n \in \mathbb{N}: \qquad nE_n(\phi_i, X, \mathtt{W}_W^{\nu(i,\cdot)}(X, \rho, L)) \leq 1.$$

Then, for any polynomial π and constant $M \in \mathbb{N}$, there exists a polynomial μ such that

$$a^*(Y, X, \Sigma^{\pi,M}(\phi)) \leq a^*(Y, X, \mathtt{W}^{\mu}(X, \rho, L)).$$

In words. If networks approximate a dictionary well, and the dictionary approximates signals well, then networks approximate the signals well.

Example

The wavelet or shearlet dictionary (ϕ_i) generated by a mother wavelet $\psi \in \mathtt{W}_W(X, \rho, L)$ satisfies the condition of the theorem. (Why?)

Proof: dictionary learning with bounded weights

Proof.

- Let $a := a^*(Y, X, \Sigma^{\pi,M}(\phi))$. We modify ν such that

$$\forall i, n \in \mathbb{N}: \qquad n^{a+1} E_n(\phi_i, X, \mathtt{W}_W^{\nu(i,\cdot)}(X, \rho, L)) \le 1.$$

- For any $f \in Y$, finite sequence c, and perceptrons ψ_i,

$$\Big\| f - \sum_i c_i \psi_i \Big\|_X \le \Big\| f - \sum_i c_i \phi_i \Big\|_X + \sum_i |c_i| \|\phi_i - \psi_i\|_X.$$

- If ψ_i have coefficients bounded by $\nu(i,n)$, then the perceptron $\sum_i \psi_i$ has coefficients bounded by $\mu(n) := M \sum_{i=1}^{\pi(n)} \nu(i,n)$.
- Taking the infimum over perceptrons ψ_i, estimating the last sum by a supremum, and taking the infimum over sequences c yields

$$E_n(f, X, \mathtt{W}^{\mu}(X, \rho, L)) \le E_n(f, X, \Sigma^{\pi,M}(\phi)) + M \sup_i n E_n(\phi_i, X, \mathtt{W}_W^{\nu(i,\cdot)}(X, \rho, L)).$$

- For any $\alpha < a$, we multiply by n^α, take the supremum over n, and note that the right-hand side is finite. □

Lower bounds on network approximation rates

The following lower bounds are special cases of the previous theorem:

Proposition

The lower bound $a^*(Y, X, \mathtt{W}^\mu(X, \rho, L)) \geq \frac{k}{d}$ *holds in* $X = L^2([-1,1]^d)$ *for smoothened ReLU activation functions* ρ*, sufficiently many layers* L*, suitable polynomials* μ*, and the following signal classes* Y*:*

- $C_K^k([-1,1]^d)$ *via wavelets, shearlets, and many more*
- $C_K^{k,pw}([-1,1]^d)$ *for* $d = 1$ *via wavelets*
- $\mathrm{STAR}_K^2([-1,1]^d)$ *for* $d = k = 2$ *via curvelets and shearlets*
- $\mathrm{CART}_K^2([-1,1]^d)$ *for* $d = k = 2$ *via curvelets and shearlets*
- $\mathrm{TEXT}_{K,M}^k([-1,1]^d)$ *for* $d = 2$ *via wave atoms*
- $\mathrm{MUTIL}_K^k([-1,1]^d)$ *via ridgelets*

Proof: $e^*(Y, X) \geq a^*(Y, X, \Sigma^{\pi,M}(\phi))$, and the indicated dictionary ϕ consists of affine transforms of perceptron(s) $\psi \in W_W(X, \rho, L)$. □

Rate-optimal approximation by neural networks

Corollary

In the previous theorem, the networks are rate-optimal for the listed signal classes Y.

Proof: The following rates are equal,

$$\frac{k}{d} \overset{1}{=} a^*(Y, X, \Sigma^{\pi,M}(\phi)) \overset{2}{\leq} a^*(Y, X, \mathtt{W}^{\mu}(X, \rho, L)) \overset{3}{\leq} e^*(Y, X) \overset{1}{=} \frac{k}{d},$$

because

1. the dictionary ϕ is rate-optimal by Parts 2 and 3,
2. networks with bounded coefficients can implement dictionary approximation with bounded coefficients by the present Part 5, and
3. networks with bounded coefficients are function encoders by Part 4.

□

Questions to answer for yourself / discuss with friends

- Repetition: Why is the network approximation rate lower-bounded by the dictionary approximation rate?
- Check your understanding: How wide and deep are the approximating networks?
- Check your understanding: The polynomial bound on the networks' coefficients has no impact on the approximation rate—why?
- Transfer: How does the present dictionary learning theorem differ from the one of Chapter 9?
- Discussion: What does the result say about deep learning? What are limitations of the result?