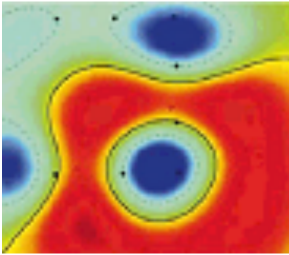
Technische Universität Berlin



Lecture 1 | Introduction

Organisational Matters

Organisational Matters

- **Lecture & homework release**
  - on Fridays 2:15pm in room A 151
- **Q&As** (the week after)
  - Q&A A on Tuesdays 2pm online (zoom)
  - Q&A B on Wednesdays 10am online (zoom)
- **Homework deadline** (the week after)
  - on Thursday 11:55pm via ISIS
- **Tutorials** (the week after)
  - Tutorial A on Fridays 10:15am in room H 2013
  - Tutorial B on Fridays 4:15pm in room A 053
  - Tutorial C on Mondays 2:15pm online (zoom)

Homework

- 10 homework assignments in total (one per week)
  - Start from Lecture 2
  - Deadline Thursday (the week after) at 23:55
  - Either theoretical or programming
  - Theoretical: Derivations, pen and paper
  - Programming: Use of ML in practice (Python/Pytorch-based)
- Submission via ISIS
  - In groups of 4 or 5 people.
  - Assignments will be graded by us
- Deadline for submitting your group: 29 October
- For general questions, please don't hesitate to use the discussion forum

Exam

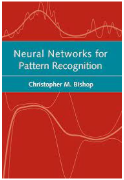
- Prerequisite for registering to Deep Learning 1 exam (6 ECTS):
  - pass ($> 50\%$) 6 out of 11 homework assignments
- Prerequisite for registering to Deep Learning 1-X exam (9 ECTS):
  - pass ($> 50\%$) 6 out of 11 homework assignments
  - complete one of the proposed electives (cf. https://web.ml.tu-berlin.de/teaching/courses/ for a list).
- Registration opens early January
- Registration either via Moses for most students or via email for exchange students
- First exam (DL1/DL1-X): 21 February 2025, room HE 101 from 14:00 to 16:00
- Second exam (DL1/DL1-X): 28 March 2025, room H 104 from 14:00 to 16:00
- Final grade is based exclusively on the course's exam

Lecture

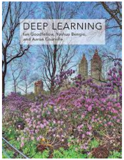
Outline

- Review of Classical ML
  - Linear & Nonlinear Models
- Deep Learning / Neural Networks
  - Motivations
  - Biological vs. Artificial Neuron
  - Biological vs. Artificial Neural Networks
  - Practical Architectures
- Applications of Deep Learning
  - DL for Autonomous Decision Making
  - DL for Data Science
  - DL for Neuroscience
- Theoretical Considerations
  - Universal Approximation Theorem
  - Compactness of Representations
  - Optimization

Book Suggestions



C. Bishop
Neural Networks for Pattern Recognition
Oxford University Press, 1995



I. Goodfellow, Y. Bengio, A. Courville
Deep Learning
MIT Press, 2016

(online version at: https://www.deeplearningbook.org/)

Part 1

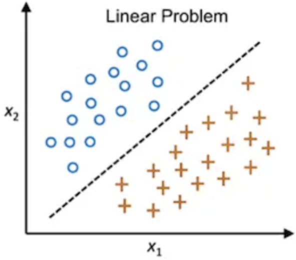
Review of Classical ML

ML Review: Linear Models

A linear classification model takes as input a data point $\boldsymbol{x} \in \mathbb{R}^d$ (a vector) and applies the linear function:

$$\begin{aligned} f(\boldsymbol{x}) &= x_1 w_1 + x_2 w_2 + \cdots + x_d w_d + b \\ &= \boldsymbol{w}^\top \boldsymbol{x} + b \end{aligned}$$

to the data point. It then classifies the data point to be of the first class if $f(\boldsymbol{x}) > 0$ and of the other class if $f(\boldsymbol{x}) < 0$.



ML Review: Learning a Linear Model

In practice, we would like to learn a model from some training set of data points and label pairs $\mathcal{D} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$. A popular formulation is given by the constrained optimization problem:

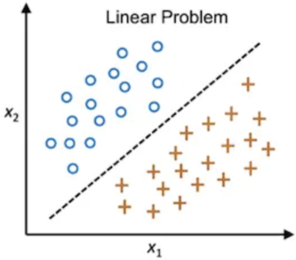
$$\min_{\boldsymbol{w}, b} \|\boldsymbol{w}\|^2$$

s.t.

$$\forall\ i \in \text{class } 1: \quad \boldsymbol{x}_i^\top \boldsymbol{w} + b \geq 1$$

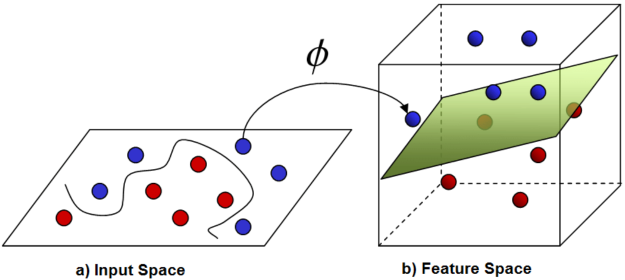
$$\forall\ i \in \text{class } 2: \quad \boldsymbol{x}_i^\top \boldsymbol{w} + b \leq -1$$

which finds the decision boundary between the two classes that has the highest margin. This is a convex optimization problem (convex objective and convex constraints), and one can easily extract the global optimum.



ML Review: From Linear to Nonlinear Models

Most problems are however not linearly separable, and we need a way to enable ML models to learn nonlinear decision boundaries. A simple approach consists of nonlinearly mapping $\boldsymbol{x}$ to some high-dimensional feature space $\phi(\boldsymbol{x})$, and classify linearly in that space. The decision boundary becomes nonlinear in input space.

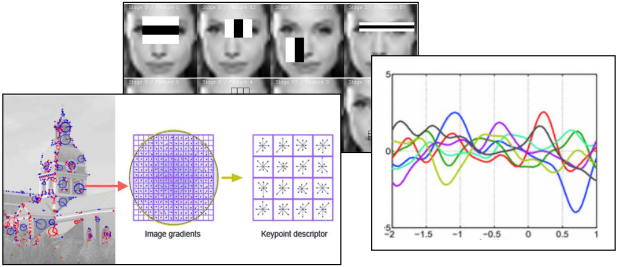


Question: How to choose the feature map ϕ?

ML Review: Features Engineering

Idea:

- Extract through some hand-designed algorithm input features that make sense for the task, and store them in some feature vector $\phi(\boldsymbol{x})$.



Limitation:

- No guarantee that the first few features the algorithm generates are good enough/sufficient to solve the task accurately. Making the problem linearly separable may require an extremely large number of features ($\rightarrow$ computationally expensive).

Part 2

Deep Learning / Neural Networks

Beyond Feature Engineering: Deep Learning

Empirical Observation:

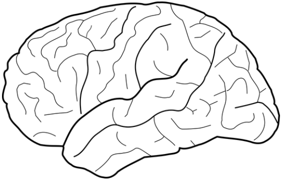
- Humans have shown capable of mastering tasks such as visual recognition, motion, speech, games, etc. All these tasks are highly nonlinear (i.e. they somehow need some nonlinear feature representation $\phi(\boldsymbol{x})$).

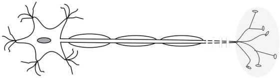
Question:

- Can machine learning models take inspiration of some mechanisms in the human's brain in order to learn the needed feature representation $\phi(\boldsymbol{x})$?

The Human Brain as a Model for Machine Learning

- The human brain is a highly complex (and so far scarcely understood) system.
- Scientific research in the past century has however provided some understanding of what might enable these systems to learn successfully:
  - Complex abstract representations result from the *interconnection* of many simple nonlinear neurons.
  - The property of these neurons to *modify* their response when exposed repeatedly to a certain stimuli enables the brain to learn.



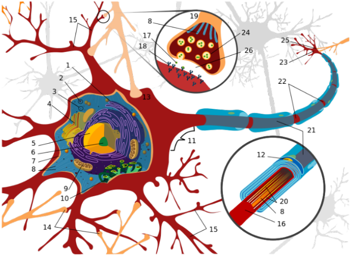


人脑作为机器学习的模型

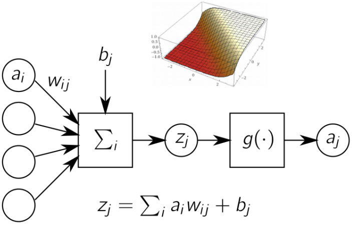
- 人脑是一个极其复杂（并且迄今仍难以理解的）系统。
- 过去一个世纪的科学研究为我们提供了一些关于这些系统如何成功学习的理解：
  - 复杂的抽象表示是由许多简单的非线性神经元**相互连接**所形成的。
  - 这些神经元具有**修改**自身反应的特性，当反复暴露于某种刺激时，它们能够调整自身，使大脑具备学习能力。

Biological vs. Artificial Neurons

Biological neuron



Artificial neuron



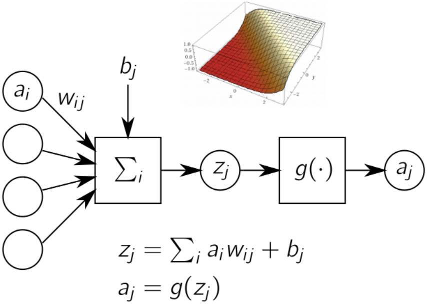
$$z_j = \textstyle\sum_i a_i w_{ij} + b_j$$
$$a_j = g(z_j)$$

- The biological neuron is a highly sophisticated physical system with complex spatio-temporal dynamics that transfers signal received by dendrites to the axon.
- Artificial neurons only retain the most essential components of the biological neuron for practical purposes: *nonlinearity* and ability to *learn*.

• 生物神经元是一个高度复杂的物理系统，具有复杂的时空动态，能够将树突接收到的信号传递到轴突。

• 人工神经元仅保留了生物神经元最基本的功能，以满足实际应用需求：**非线性和学习能力。**

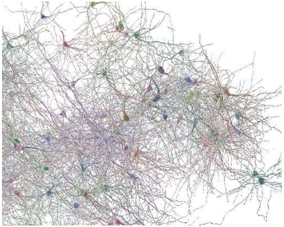
The Artificial Neuron



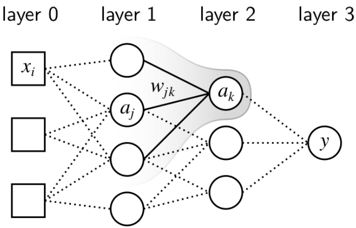
- Simple multivariate, nonlinear and differentiable function.
- Ultra-simplification of the biological neuron that retains two key properties: (1) ability to produce complex nonlinear representations when many neurons are interconnected (2) ability to learn from the data.

Interconnecting Multiple Neurons

Biological network

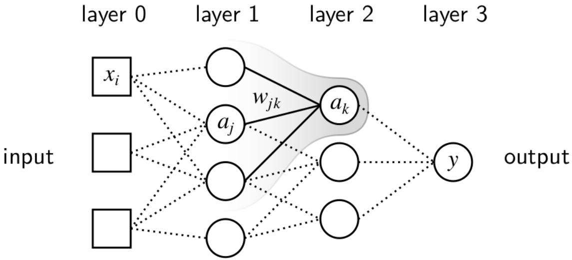


Artificial network



- The human brain is composed of a very large number of neurons (approx. 86 billions) that are interconnected (150 trillions synapses).
- An artificial neural network mimicks the way biological neurons are connected in the brain by composing many artificial neurons. For practical purposes, neurons of an artificial neural network can be organized in a layered structure.
  - 人脑由大量的神经元组成（约 860 亿个），这些神经元相互连接，形成约 150 万亿个突触。
  - 人工神经网络模拟了生物神经元在大脑中的连接方式，通过构建多个人工神经元来实现。出于实际应用目的，人工神经网络的神经元通常以**层状结构**进行组织。

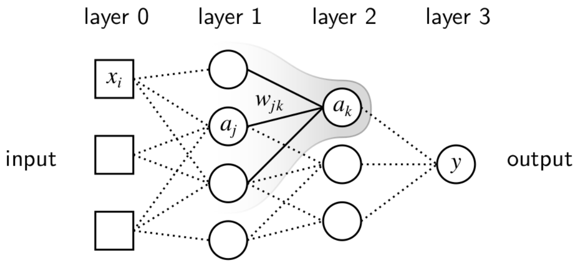
Neural Networks: Forward Pass



The forward pass mapping the input of the network to the output is given by:

$$
\begin{array}{lll}
z_j = \sum_i x_i w_{ij} + b_j & a_j = g(z_j) & \text{(layer 1)} \\
z_k = \sum_j a_j w_{jk} + b_k & a_k = g(z_k) & \text{(layer 2)} \\
y = \sum_k a_k v_k + c & & \text{(layer 3)}
\end{array}
$$

Neural Networks: Forward Pass (Matrix Formulation)



Matrix formulation:

$$z^{(1)} = W^{(1)}x + b^{(1)} \qquad a^{(1)} = g(z^{(1)}) \qquad \text{(layer 1)}$$

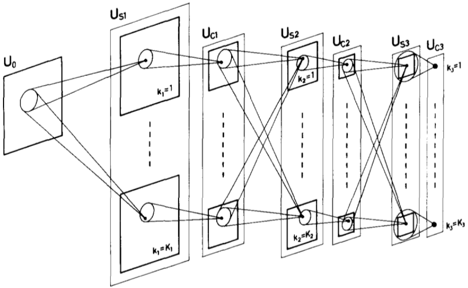
$$z^{(2)} = W^{(2)}a^{(1)} + b^{(2)} \qquad a^{(2)} = g(z^{(2)}) \qquad \text{(layer 2)}$$

$$y = v^\top a^{(2)} + c \qquad \text{(layer 3)}$$

where $[W^{(1)}]_{ji} = w_{ij}$, $[W^{(2)}]_{kj} = w_{jk}$, and where g applies element-wise. The matrix formulation makes it convenient to train neural networks with hundreds, thousands, or more neurons.

Image Recognition: The Neocognitron (1979)

The Neocognitron [2] is an early neural network for predicting images. It is designed in a way that the produced output becomes approximately invariant small local translations/distortions in the input image.

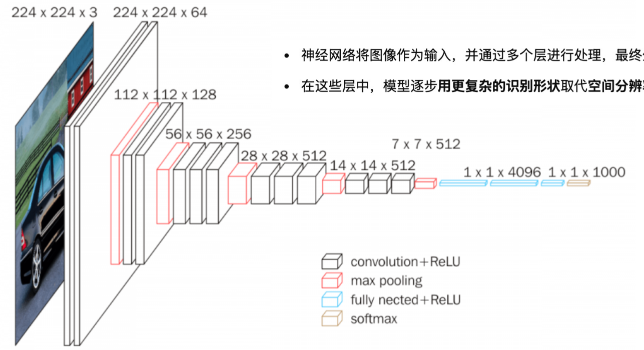


The Neocognitron consists of an alternation of 'simple cells' (convolutions) and 'complex cells' (pooling). It is a precursor of modern convolutional neural network architectures.

- Neocognitron 是一种用于图像预测的早期神经网络。其设计方式使得其输出在面对输入图像的**小尺度平移/扭曲**时，能够保持大致不变。
- Neocognitron 由"简单细胞"（卷积）与"复杂细胞"（池化）交替组成。它是现代卷积神经网络（CNN）架构的先驱。

Image Recognition: Large ConvNets (2012–. . .)

Example: The VGG-16 convolutional neural network [4]:

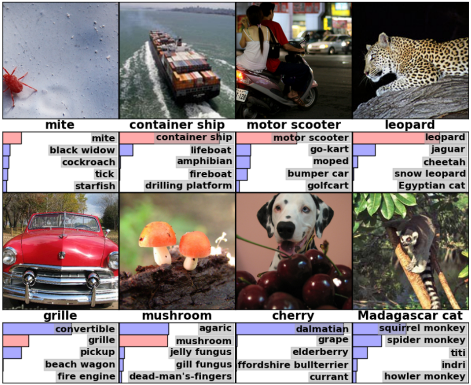


- 神经网络将图像作为输入，并通过多个层进行处理，最终生成预测结果。
- 在这些层中，模型逐步**用更复杂的识别形状**取代**空间分辨率**。

- The neural network takes the image as input and processes it by multiple layers to finally arrive at a prediction.
- Throughout the multiple layers, one progressively trades spatial resolution for more complex recognized shapes.

Image Recognition: Large ConvNets (2012–. . .)

Examples of Prediction:



Krizhevsky et al. ImageNet Classification with Deep Convolutional Neural Networks. NIPS 2012

- Can accurately predict images into a large number of classes (1000 possible classes).
- Even misclassifications of the model are somewhat reasonable (e.g. two different objects in the same image, similar classes).

Other Deep Learning Successes

- **语音识别** 手动从原始波形或声谱图中提取高质量特征十分困难。语音信号常常受到复杂噪声模式的影响（如回声、混响、多重信号源）。深度学习/神经网络已经成为**语音识别**领域的最新技术（如 DeepSpeech2）。
- **自然语言处理** 与形式语言不同，自然语言**没有简单的方法**可以进行解析。句子的复杂结构需要经过抽取（如逻辑推理、情感分析、讽刺理解）。**深度学习架构**（如 Transformer 网络）已在实践中取得了巨大成功（如 BERT、GPT、LLaMA 语言模型）。
- **游戏对弈** 深度学习与其他人工智能技术（如搜索、强化学习）相结合，使其在许多复杂和竞技类游戏中超越人类水平（如 AlphaGo、AlphaZero）。

Examples:

Speech Recognition
: Hard to manually extract good features from the raw waveform or a spectrogram. Speech entangled with complex noise patterns (e.g. echo, reverberation, multiple sources). Deep learning / neural networks have become state-of-the-art on speech recognition (e.g. DeepSpeech2).

Natural Language Processing
: Unlike formal languages, there is no simple way to parse a natural language. Yet, the complex construction of the sentence needs to be extracted (e.g. logical reasoning, sentiment, irony). Deep learning architectures such as transformer networks have been highly successful in practice (e.g. BERT/GPT/LLaMA language models).

Playing Games
: Deep learning has been combined with other AI techniques (e.g. search, RL), in order to achieve above human performance in many complex and competitive games (e.g. AlphaGo, AlphaZero).

Part 3

Applications of Deep Learning

Applications of Deep Learning

三大应用类别：

- **自主决策** 在特定环境中做出优质决策（可以替代人类决策者，或者辅助/支持人类决策）。应用领域包括机器人技术、推荐系统、医学诊断等。
- **数据科学 / 知识发现** 学习逼近复杂过程的输入-输出关系，或者分析不同变量之间的关系。然后，通过学习到的模型来理解这些过程/关系。
- **神经科学** 使用神经网络本身作为大脑的模型，以研究大脑的工作机制。例如，探究神经网络的中间层如何与大脑特定区域的神经元活动相关联。

Three main categories of applications:

Autonomous Decision Making: Take good decisions in a given environment (can be used as a substitute for a human decider, or complement/support human decisions). Application in e.g. robotics, recommender systems, medical diagnosis.

Data Science / Knowledge Discovery: Learn to approximate the input-output relation of some complex process, or the relation between different variables of interest. Then, analyze the learned model in order to understand this process/relation.

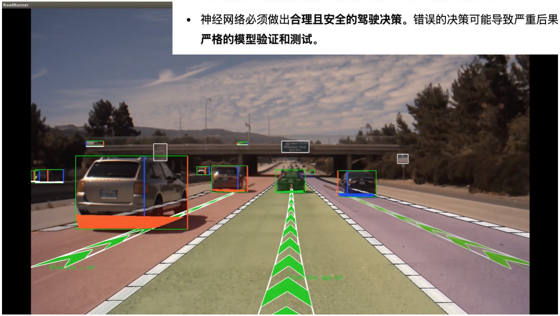
Neuroscience: Use the neural network itself as a model for the brain (in order to understand how the brain works). E.g. in which way intermediate layers correlate with neuron activations in specific areas of the brain.

Autonomous Decision Making Example

Autonomous Car Driving

自动驾驶汽车

- 深度学习可以处理传感器数据，并生成完全或部分自动化的决策，如何时左转/右转、刹车、加速等。这种自动化能够减轻（或完全取代）人类驾驶员的负担。
- 神经网络必须做出**合理且安全的驾驶决策**。错误的决策可能导致严重后果（如车祸等）。因此，**需要严格的模型验证和测试。**



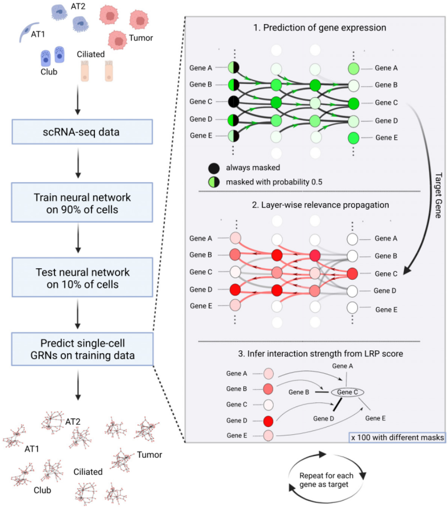
Source: https://medium.com/self-driving-cars/nvidia-drive-labs-a09627d745f9

- Deep learning can process sensor data and produce fully or partly automated decisions of when to turn left/right, brake, accelerate, etc. Such automation enables to lower the burden on (or fully replace) the human driver.
- The neural network must make meaningful and safe driving decisions. Incorrect decisions can have severe consequences (crash, etc.). $\rightarrow$ Need for stringent model validation and testing.

Data Science Example (1)

数据科学示例（1）

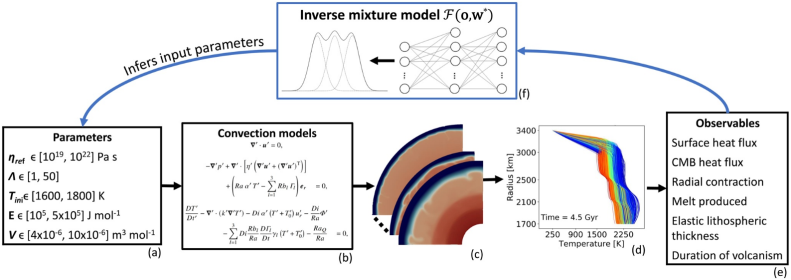
- 训练神经网络处理**基因表达数据集**，以**预测基因表达**，通过其他基因表达水平来推测目标基因的表达情况。
- 从训练好的模型中提取**基因相互作用的模型**（即基因调控网络）。



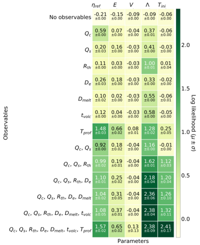
- Train a neural network on a gene expression dataset to mutually predict gene expression from other gene expressions.
- Retrieve from the trained model a model of interaction between genes (a gene regulatory network).

Keyl et al. Nucleic Acids Research, gkac1212, 2023

Data Science Example (2)



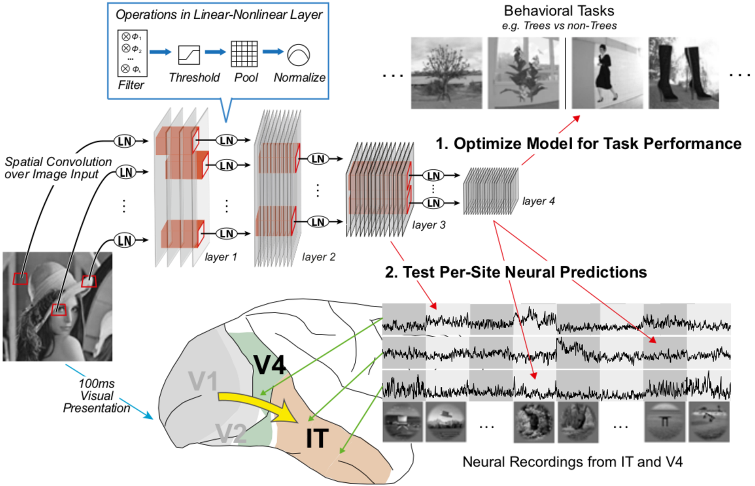
- Train several neural networks to predict from various subsets of observables internal parameters of a planet.
- This enables to infer what subsets of observables have the highest predictive power, and which ones are therefore the most worthy to measure in practice.



Agarwal et al. Earth and Space Science 8 (4), e2020EA001484, 2021

- 训练多个神经网络，通过不同可观测变量的子集来预测**行星的内部参数。**
- 这使得我们能够推断出**哪些可观测变量子集具有最强的预测能力**，以及哪些变量在实际测量中最值得关注。

Neuroscience Example



Cadieu et al. PNAS 111 (23), 8619-8624, 2014

Part 4

Theoretical Considerations

Theoretical Considerations about Neural Networks

- **Universality:** Can they approximate any functions (assuming we have enough neurons)?
- **Compactness:** Can functions (and the learning of these functions) be represented in a compact form (i.e. using finitely many neurons)?
- **Optimization:** Are neural network easy/hard to optimize (e.g. can the optimization procedure get stuck in local optima)?

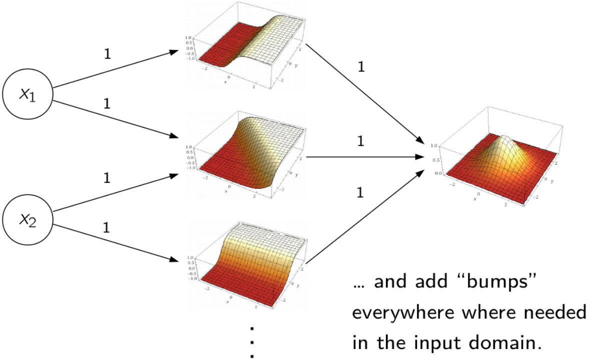
关于神经网络的理论考量

- **普适性**：神经网络是否可以逼近**任何函数**（假设我们有足够的神经元）？
- **紧凑性**：函数（以及对这些函数的学习）是否可以用**有限数量的神经元**以紧凑的形式表示?
- **优化**：神经网络是否易于/难以优化（例如，优化过程是否容易陷入**局部最优**）？

Universal Approximation Theorem (1)

Neural networks with sufficiently many neurons can approximate any function f of its input variables $x_1, x_2, \dots, x_d$.

"Proof" by construction:



Universal Approximation Theorem (2)

Theorem (simplified): With sufficiently many neurons, neural networks can approximate any nonlinear functions.

Sketch proof taken from the book Bishop'95 Neural Network for Pattern Recognition, p. 130–131, (after Jones'90 and Blum&Li'91):

- Consider the special class of functions $y : \mathbb{R}^2 \to \mathbb{R}$ where input variables are called x_1, x_2.
- We will show that any two-layer network with threshold functions as nonlinearity can approximate $y(x_1, x_2)$ up to arbitrary accuracy.
- We first observe that any function of x_2 (with x_1 fixed) can be approximated as an infinite Fourier series.

$$y(x_1, x_2) \simeq \sum_s A_s(x_1) \cos(s x_2)$$

Universal Approximation Theorem (3)

- We first observe that any function of x_2 (with x_1 fixed) can be approximated as an infinite Fourier series.

$$y(x_1, x_2) \simeq \sum_s A_s(x_1) \cos(s x_2)$$

- Similarly, the coefficients themselves can be expressed as an infinite Fourier series:

$$y(x_1, x_2) \simeq \sum_s \sum_l A_{sl} \cos(l x_1) \cos(s x_2)$$

- We now make use of a trigonometric identity to write the function above as a sum of cosines:

$$\cos(\alpha)\cos(\beta) = \frac{1}{2}\cos(\alpha + \beta) + \frac{1}{2}\cos(\alpha - \beta)$$

- Thus, the function to approximate can be written as a sum of cosines, where each of them receives a linear combination of the input variables:

$$y(x_1, x_2) \simeq \sum_{j=1}^{\infty} v_j \cos(x_1 w_{1j} + x_2 v_{2j})$$

Universal Approximation Theorem (4)

- Thus, the function to approximate can be written as a sum of cosines, where each of them receives a linear combination of the input variables:

$$y(x_1, x_2) \simeq \sum_{j=1}^{\infty} v_j \cos(x_1 w_{1j} + x_2 v_{2j})$$

- This is a two-layer neural network, except for the cosine nonlinearity. The latter can however be approximated by a superposition of a large number of step functions.

$$\cos(z) = \lim_{\tau \to 0} \sum_i \underbrace{[\cos(\tau \cdot (i+1)) - \cos(\tau \cdot i)]}_{\text{constant}} \cdot \underbrace{1_{z > \tau \cdot (i+1)}}_{\text{step function}} + \text{const.}$$

- 因此，要逼近的函数可以表示为**余弦函数的和**，其中每个余弦函数都接收输入变量的线性组合：

$$y(x_1, x_2) \simeq \sum_{j=1}^{\infty} v_j \cos(x_1 w_{1j} + x_2 v_{2j})$$

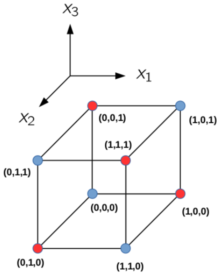
- 这个结构是一个**两层神经网络**，不同之处在于其**非线性部分使用了余弦函数**。然而，余弦函数可以通过**大量阶跃函数的叠加**来逼近：

$$\cos(z) = \lim_{\tau \to 0} \sum_i \underbrace{[\cos(\tau \cdot (i+1)) - \cos(\tau \cdot i)]}_{\text{常数项}} \cdot \underbrace{1_{z > \tau \cdot (i+1)}}_{\text{阶跃函数}} + \text{常数}$$

Neural Networks: Compactness (1)

Neural networks can express a broad range of 'useful' functions in compact manner (e.g. without having to use exponentially many neurons).

Function to approximate: $f(\boldsymbol{x}) = \text{parity}(x_1, x_2, \dots, x_d)$



Naive approach:
Use one neuron to handle each corner of the hypercube. (overall 2^d neurons)

Better approach:
Use a deep composition of elementary modules (4 x d neurons).

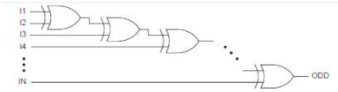
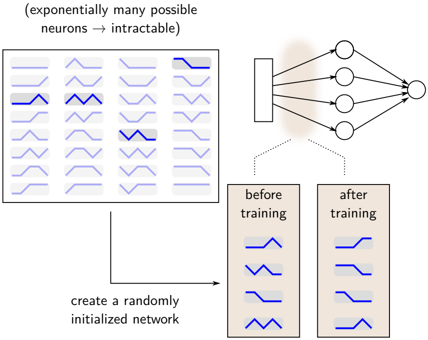


Image source: http://vlsi-design-engineers.blogspot.com/2015/10/exclusive-or-gates-parity-circuits-and.html

Neural Networks: Compactness (2)

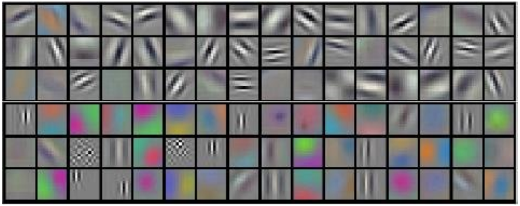
- 神经网络从**有限且通常较小的随机初始化神经元集合**开始（即所有可能神经元的一个子集）。
- 在训练过程中，**紧凑的问题表示**逐步被提取，这一过程受**优化（最小化误差）和有限数量的神经元**的共同作用影响。
- 学到的表示几乎与**包含所有神经元的完整模型**一样具有预测能力，但更加紧凑。



- The neural network starts with a finite and typically small set of randomly initialized neurons (i.e. a subset of all possible neurons).
- The compact problem representation is progressively extracted during training under the simultaneous effect of optimization (minimizing the error) and the finite number of neurons in the model.
- The learned representation is almost as predictive as an exhaustive set of neurons, but much more compact.

Neural Networks: Compactness (3)

Example of the set of first-layer filters learned by a neural network trained on image classification (AlexNet):

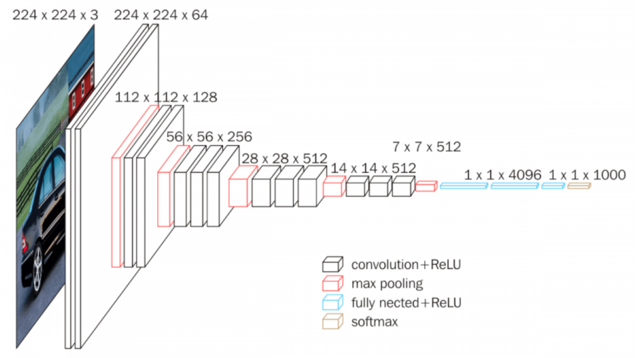


These 96 filters capture most of the important low-level signal for image classification, and are much more compact than the exhaustive set of all possible filters (potentially thousands or millions of possible filters).

示例：神经网络在图像分类任务（AlexNet）中学习到的第一层滤波器

- 这些**96 个滤波器**捕捉了图像分类任务中**大部分重要的低级信号**，相比于包含所有可能滤波器（可能达到**成千上万个甚至百万个**）的完整集合，它们更加紧凑。

Neural Networks: Compactness (4)



- Progressive tradeoff between spatial resolution and semantic resolution ensures that the representation remains compact at every step.

Neural Networks: Optimization

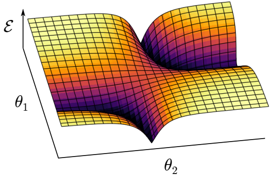
Neural networks also have downsides:

- Non-convex objective (e.g. even the simplest two-layer network $\phi(x;\theta) = \theta_1\theta_2 x$ is already non-convex with θ). Many hyperparameters (e.g. initialization, learning rate, etc.) can affect the result of learning.
- Multiple layers can cause pathological curvature, i.e. the gradient vanishes along certain directions of the parameter space. The optimizer may get stuck on large plateaus.

With heuristics on the neural network design (e.g. choice of layers and nonlinearities) and optimization (e.g. momentum, batch-normalization), it is however still possible to train them efficiently.

神经网络：优化

- 神经网络也存在一些缺点：
  - **非凸目标函数**（例如，即使是最简单的两层网络 $\phi(x;\theta) = \theta_1\theta_2 x$ 也已经是关于 θ **非凸**的）。许多**超参数**（如初始化方式、学习率等）都会影响学习结果。
  - **多层结构可能导致病态曲率**，即在参数空间的某些方向上梯度会消失，导致优化器可能陷入**大范围的平台**。
- 通过在**神经网络设计**（如层数选择、非线性函数）和**优化策略**（如动量、批归一化）上的启发式方法，仍然可以**高效训练**神经网络。



Neural Networks vs. Other Feature Extraction

	Universal	*Compact*	*Convex/Easy*
Feature Engineering			
(few features)	✗	✓	✓
(many features)	✓	✗	✓
Neural Networks	✓	✓	✗

- Compare to feature engineering approaches, neural networks are able to achieve at the same to solve a broad range of problems (universal) and in a way that keeps the model reasonably small (compact).
- However, this comes at the cost of a more complex optimization procedure. Heuristics will be presented in Lectures 3 and 4 on how to nevertheless optimize neural networks efficiently.

- 与**特征工程**方法相比，神经网络能够**解决更广泛的问题（通用性）**，同时保持模型的**相对紧凑性。**
- 但是，这也导致了**优化过程更复杂**。在**第 3 讲和第 4 讲**中将介绍一些优化神经网络的策略。

Summary

Summary

- **Deep learning** is a learning paradigm where both the classifier and the features supporting the classifier are learned from the data.
- Deep learning relies on **neural networks**, specifically, their ability to represent and learn complex nonlinear functions through the interconnection of many simple computational units (neurons).
- Deep learning provides a solution for **difficult tasks** where many classical ML techniques do not work well (e.g. image recognition, speech recognition, natural language processing), and has become state-of-the-art on many such tasks.
- Deep learning is often used in practice for its ability to produce accurate decisions **autonomously**, however, there are also a broad range of possible applications of deep learning in **data science** as well as in **neuroscience**.
- Deep learning can learn models that are both **compact** and highly **adaptable** to the task. At the same time, the optimization problem is **non-convex** and generally harder, which makes them more difficult to handle.

References

- **深度学习（Deep learning）** 是一种学习范式，其中分类器及其支持的特征**都是从数据中学习**而来的。
- 深度学习依赖于**神经网络（neural networks）**，特别是其**通过许多简单计算单元（神经元）之间的相互连接**来表示和学习复杂的非线性函数的能力。
- 深度学习**为许多复杂任务（difficult tasks）提供了解决方案**，这些任务使用传统的机器学习技术效果不佳（如图像识别、语音识别、自然语言处理），并且已成为许多此类任务的**当前最优技术。**
- 在实际应用中，深度学习**常用于自主（autonomously）决策**，但它的应用范围也很广，包括**数据科学（data science）**和**神经科学（neuroscience）**等领域。
- 深度学习能够学习出**紧凑（compact）且高度适应（adaptable）任务**的模型。但与此同时，它的优化问题是**非凸（non-convex）**的，通常较难处理，这使得神经网络更加难以训练。

S. Agarwal, N. Tosi, P. Kessel, S. P
Toward constraining mars' thermal
Earth and Space Science, 8(4):e2020EA001484, 2021.

K. Fukushima.
Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position.
Biological Cybernetics, 36(4):193–202, 1980.

P. Keyl, P. Bischoff, G. Dernbach, M. Bockmayr, R. Fritz, D. Horst, N. Blüthgen, G. Montavon, K.-R. Müller, and F. Klauschen.
Single-cell gene regulatory network prediction by explainable AI.
Nucleic Acids Research, Jan. 2023.

K. Simonyan and A. Zisserman.
Very deep convolutional networks for large-scale image recognition.
In *ICLR*, 2015.

D. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo.
Performance-optimized hierarchical models predict neural responses in higher visual cortex.
Proceedings of the National Academy of Sciences, 111:8619–8624, 2014.