Computational Intelligence Laboratory

Lecture 1

Gunnar Rätsch
(for Thomas Hofmann on Sabbatical)

ETH Zurich – `da.inf.ethz.ch/cil`

25 February 2022

Section 0

Course Philosophy

Computational Intelligence

Data Science & Machine Learning: Foundation for Sciences, Engineering & Technology

- interpreting experimental data
- making predictions about likely outcomes
- supporting data-informed decisions
- enabling machines to perform intelligent tasks

$\Rightarrow$ Computational Intelligence

AI Infrastructure

Evergrowing number of methods and models available

- conveniently packaged in software libraries (e.g., PyTorch)
- unprecedented computing machinery (e.g., GPU servers)
- active community (e.g., blogs, tutorials, best practices)
- rapid innovation, time-to-product

$\Rightarrow$ Low Barriers, High Productivity, High Success Rates

Competence and Understanding

Importance of **Understanding**

- Use of computational intelligence needs to be accompanied by appropriate **competence**
- **Mathematical analysis** of models & algorithms
- Deepen the foundations rather than following the fashion
- Models are often very complex $\Rightarrow$ difficult to analyze
- **Simplifications**: perform analysis & calculations with simple models, develop theory, extrapolate with experiments

Everything should be made as simple as possible, but no simpler.

– Albert Einstein

Calculation vs. Computation

- Easy to "run code over data"! But then what?
- Deep Learning: mainstream as **black art** – tweaking, fishing, hacking, folklore knowledge, pseudo-scientific language
- Clean calculation = insights, understanding, clarity!
- Computational intelligence: prefer **calculations** over **computations** for understanding

In any special doctrine of nature there can be only as much proper science as there is mathematics therein.

– Immanuel Kant

Metaphysical Foundations of Natural Science (1786, 4:470)

CIL $= 2\times$ Laboratory

"Hands-on" Mathematics

- "Hands-on" use of mathematical methods: linear algebra, multivariate analysis, probability theory, statistics
- Mathematical modeling (role model: theoretical physics) $=$ applied mathematics.
- Emphasis not on proving abstract theorems, but: performing sensible calculations $\Rightarrow$ Practice mathematical skills

"Hands-on" Programming

- Practical projects $\Rightarrow$ Develop genuine solutions
- But: take guidance from analysis & theory

Course Content

Dimensionality reduction

- Linear autoencoders, projections, principal component analysis, learning algorithms, non-linear autoencoders

Matrix Approximation

- Collaborative filtering, Rank 1 model, singular value decomposition, alternating least squares, projection algorithms, exact matrix reconstruction

Latent Variable Models

- Probabilistic Clustering Models, topic models, embeddings

Deep Neural Networks

- Compositional models, Backpropagation, gradient descent, convolutional neural networks

Generative Models

- Autoregressive models, normalizing flows, variational autoencoders, generative adversarial networks

Course Material

Lectures

- Thomas Hofmann is on Sabbatical this semester.
- Lectures will be given by Prof. Gunnar Rätsch (raetsch@ethz.ch)
- No major changes in the lecture material or content.

Course Material

Lectures

- Thomas Hofmann is on Sabbatical this semester.
- Lectures will be given by Prof. Gunnar Rätsch (raetsch@ethz.ch)
- No major changes in the lecture material or content.

Course Moodle

- `https://moodle-app2.let.ethz.ch/course/view.php?id=16549`
- Slides will usually be available a day in advance
- Please use Moodle to ask questions about the course content or organization.
- Explore use of "CIL Overflow", a StackOverflow-like plugin

Course Material

Lectures

- Thomas Hofmann is on Sabbatical this semester.
- Lectures will be given by Prof. Gunnar Rätsch (raetsch@ethz.ch)
- No major changes in the lecture material or content.

Course Moodle

- `https://moodle-app2.let.ethz.ch/course/view.php?id=16549`
- Slides will usually be available a day in advance
- Please use Moodle to ask questions about the course content or organization.
- Explore use of "CIL Overflow", a StackOverflow-like plugin

Lecture Notes

- Will be used unchanged (except typos/mistakes) from last year. Provided via Moodle/course website.

Course Format

Lectures Friday 10am-12pm

- By default, all lectures in presence in ML D 28.
- Zoom access: `https://ethz.zoom.us/j/69416491737?pwd=bFQvMjJCU0ZHMU8rZjhGbkd0SWZ3QT09`
- We will provide recordings of each lectures (check https://video.ethz.ch/lectures/d-infk/2022/spring.html).
- No lectures on April 15 & April 22

Course Format

Lectures Friday 10am-12pm

- By default, all lectures in presence in ML D 28.
- Zoom access: `https://ethz.zoom.us/j/69416491737?pwd=bFQvMjJCU0ZHMU8rZjhGbkd0SWZ3QT09`
- We will provide recordings of each lectures (check https://video.ethz.ch/lectures/d-infk/2022/spring.html).
- No lectures on April 15 & April 22

Exercises sessions: Friday 4-6pm, Q&A Thursday 2-3pm

- Organized via Zoom: `https://zoom.us/j/2288537317`
- Recordings of the exercises will be provided.
- No recordings of the Q&A sessions.

Leonard Adolphs

Antonio Orvieto

Plan for the rest of today

- Intro of exercises and projects by Head TAs
- Break
- Start of first content block (dimensionality reduction)

Section 1

Dimension Reduction
Introduction

Motivation

Finding low-dimensional data representations

- Original raw representation often high-dimensional and redundant. Examples: images, audio, time series.
- Goal (i): compress data (while preserving relevant information)
- Goal(ii): interpretable representation, different modes of variation factored out
- Historically: Pearson 1901, Principal Component Analysis

Auto-Encoder

Taking a Deep Neural Network (DNN) viewpoint: **Auto-Encoder**

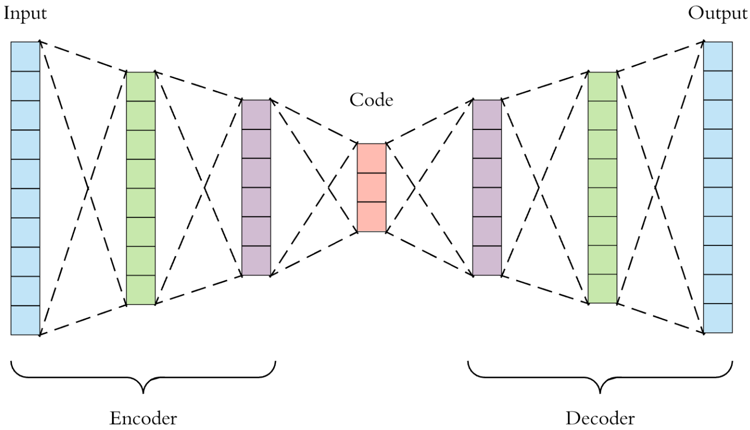

Diagram $\to$ Mathematics

Where does data come from?

- Data generating law $\mathbf{x} \sim \nu$ (probability measure, implicit)
- Sample set
$$\mathcal{S} = \{\mathbf{x}_i \overset{\text{iid}}{\sim} \nu,\ i = 1, \ldots, s\}$$
- Notation: expectation, true and empirical
$$\mathbf{E}_\nu[f(\mathbf{x})] \quad \text{and} \quad \mathbf{E}_{\mathcal{S}}[f(\mathbf{x})]$$

Diagram $\to$ Mathematics

Relevant functions and maps?

- Encoder & decoder (e.g., $m << n$)

$$F : \mathbb{R}^n \to \mathbb{R}^m, \qquad G : \mathbb{R}^m \to \mathbb{R}^n$$

- Reconstruction map

$$G \circ F : \mathbb{R}^n \to \mathbb{R}^n \qquad \text{(composition of maps)}$$

  - Ideally (but unachievable)

$$G \circ F = \mathrm{id}$$

Diagram $\to$ Mathematics

Quality criterion; Distortion Measure?

- Abstractly: **loss function**

$$\ell : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}, \quad (\mathbf{x}, \hat{\mathbf{x}}) \mapsto \ell(\mathbf{x}, \hat{\mathbf{x}})$$

    - ...in the absence of domains-specific knowledge ...

- **Quadratic loss**

$$\ell(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{2}\|\mathbf{x} - \hat{\mathbf{x}}\|^2$$

    - ... do not forget: this is a convenient choice ...

Diagram → Mathematics

Risk function?

- Risk function = average loss – what distribution?
- Empirical Risk:

$$\mathbf{E}_{\mathcal{S}}[\ell] = \underbrace{\frac{1}{2s}\sum_{t=1}^{s}}_{\text{sample average}} \| \underbrace{\mathbf{x}_t}_{\text{sample}} - \underbrace{(G \circ F)(\mathbf{x}_t)}_{\text{reconstruction}} \|^2$$

- New Data Risk: (Lebesgue integral)

$$\mathbf{E}_{\nu}[\ell] = \int \ell(\mathbf{x}, (G \circ F))(\mathbf{x})\, d\nu(\mathbf{x})$$

Diagram $\rightarrow$ Mathematics

Layers of units (vectors back and forth)

- Simplicity (1): start with single layer (for encoder and decoder)
- Simplicity (2): start with simple functions F and G
  $\Rightarrow$ Linear Auto-Encoder
- Bring back (compositional) depth in the end

Section 2

Dimension Reduction:
Linear Auto-Encoder

Linear Auto-Encoder

Identifying F, G with linear maps

linear encoder $\quad F : \mathbf{x} \mapsto \mathbf{z} = \mathbf{W}\mathbf{x}, \quad \mathbf{W} \in \mathbb{R}^{m \times n}$

linear decoder $\quad G : \mathbf{z} \mapsto \hat{\mathbf{x}} = \mathbf{V}\mathbf{z}, \quad \mathbf{V} \in \mathbb{R}^{n \times m}$

Linear Auto-Encoder objective

$$\mathcal{R}(\mathbf{W}, \mathbf{V}) = \mathcal{R}(\mathbf{P} := \mathbf{V}\mathbf{W}) = \mathbf{E}\left[\tfrac{1}{2}\|\mathbf{x} - \mathbf{P}\mathbf{x}\|^2\right]$$

Linearity $=$ natural simplification for initial analysis

Linear Compositionality

Composing of linear maps $\equiv$ matrix multiplication

$$F(\mathbf{x}) = \mathbf{W}\mathbf{x}, \quad G(\mathbf{z}) = \mathbf{V}\mathbf{z}$$

$$(G \circ F)(\mathbf{x}) = G(F(\mathbf{x})) = \mathbf{V}(\mathbf{W}\mathbf{x}) = (\mathbf{V}\mathbf{W})\mathbf{x}$$

(simply follows from associativity of matrix product)

Does it make sense to compose linear functions for F or G?

No, if the goal is to increase expressivity (or modeling power)!

Affine Maps

Q: Are affine autoencoders more powerful than linear ones?

A: For centered data and the squared loss: optimal affine reconstruction maps are linear.

Centering of data

$$\mathbf{x} \leftarrow \mathbf{x} - \mathbf{E}[\mathbf{x}]$$

(e.g., subtract sample mean, simple pre-processing)

Affine Maps

Proof.

Let $\mathbf{a} \neq \mathbf{0}$, then

$$\begin{aligned}\mathbf{E}\|\mathbf{x} - (\mathbf{P}\mathbf{x} + \mathbf{a})\|^2 &= \mathbf{E}\|\mathbf{x} - \mathbf{P}\mathbf{x}\|^2 + \|\mathbf{a}\|^2 + 2\langle \mathbf{a}, \underbrace{\mathbf{E}\mathbf{x} - \mathbf{P}\mathbf{E}\mathbf{x}}_{=\mathbf{0}} \rangle \\ &> \mathbf{E}\|\mathbf{x} - \mathbf{P}\mathbf{x}\|^2\end{aligned}$$

□

Affine Maps

A: For centered data and squared loss, optimal affine maps degenerate to linear ones.

Proof.

Note that

$$\mathbf{V}(\mathbf{W}\mathbf{x} + \mathbf{a}) + \mathbf{b} = \mathbf{V}\mathbf{W}\mathbf{x} + \mathbf{c}, \quad \text{where } \mathbf{c} = \mathbf{b} + \mathbf{V}\mathbf{a}$$

$\square$

Affine Maps

When centering data as a preprocessing step, affine maps cannot obtain better reconstruction than linear ones in autoencoders.

Identifiability & Non-Identifiability

Q: Is the representation of $\mathbf{P}$ as $\mathbf{P} = \mathbf{V}\mathbf{W}$ unique?

Two questions

- First: is the optimal linear reconstruction map unique?
- Second: is the parameterization via weight matrices $\mathbf{W}, \mathbf{V}$ unique?

Second question: **no**

$$\mathbf{V}\mathbf{W} = \mathbf{V}\mathbf{I}\mathbf{W} = \mathbf{V}(\mathbf{A}\mathbf{A}^{-1})\mathbf{W} = \underbrace{(\mathbf{V}\mathbf{A})}_{=:\mathbf{V}'}\underbrace{(\mathbf{A}^{-1}\mathbf{W})}_{=:\mathbf{W}'}$$

where $\mathbf{A}$ can be any invertible matrix.

Parameter Non-Identifiability

The weight matrices are non-identifiable and one needs to be careful not to over-interpret the found representation.

Consequence of the non-identifiability: investigate constrained classes of square matrices $\mathbf{P}$ and postpone the question of how to split it (non-uniquely) into a product of weight matrices

Rank Constraint

Q: What is the structural constraint on the reconstruction map in the autoencoder, i.e. how can we characterize $\mathbf{P} = \mathbf{V}\mathbf{W}$?

Note that we want the inner dimension $m \ll n$. It is clear that

$$\mathrm{rank}(\mathbf{P}) = \min\{\mathrm{rank}(\mathbf{V}), \mathrm{rank}(\mathbf{W})\} \leq \min\{n, m\} \stackrel{*}{=} m$$

Here the rank of a matrix (or its linear map) is defined as

$$\mathrm{rank}(\mathbf{A}) = \dim(\mathsf{im}(\mathbf{A}))$$

where the image (or range) is the linear span of the columns of $\mathbf{A}$.

Next lecture

- $\mathrm{rank}(\mathbf{P})$, linear subspace U
- Orthogonal projection to linear subspace
- Matrix Representation of Projection
- Principal Component Analysis

Homework: refresh linear algebra knowledge