

Introduction to Deep Learning (I2DL)

Lecture 0/Exercise 1: Organization

The Team

Lecturer



PhDs



Student Teaching Assistants



Julian
Balletshofer



Tathagata
Bandyopadhyay



Wei
Cao



Zeynep
Gerem



Dan
Halperin



Quoc Trung
Nguyen



Dominik
Schmauser



Anna
Weber

Today's Outline

- Organization
 - Lectures and schedule
 - Exam
 - Communication
- Exercises
 - General overview
 - Exercise 1



3. Numpy Introduction

Numpy is a high performance framework for operations on multi dimensional arrays. life as a Data Scientist in general.

```
In [12]: # Create a simple (2, 3) numpy array
# [[ 1  2  3]
#  [ 4  5  6]]
a = np.array([[1,2,3], [4,5,6]])

# Print the data type of this array
print(a.dtype)

# print the shape of this array
print(a.shape)

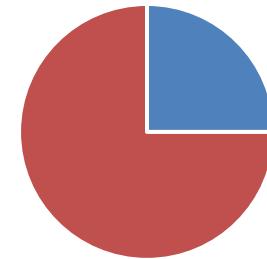
int64
(2, 3)
```

Organization

"On-Site"

- Lectures
Monday 14-16 c.t.
MI HS 1, Friedrich L. Bauer Hörsaal
- Availability
 - On-site: ~600 seats
 - Stream and VoDs: live.rbg.tum.de
(login using tum credentials)
 - Slides on our website

Are you coming to class?



■ Yes ■ No, videos only



Tentative Schedule

Introduction

0: Organization

- 1: What is Deep Learning?
- 2: Machine Learning Basics

Neural Networks

3: Introduction

4: Optimization

5: Stochastic Gradient Descent

6: Training Neural Networks

7: Training Neural Networks

8: Training Neural Networks

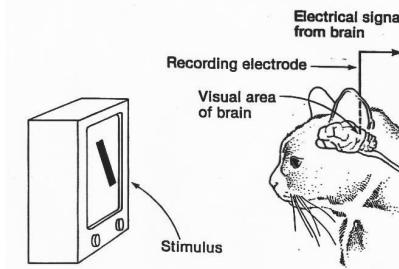
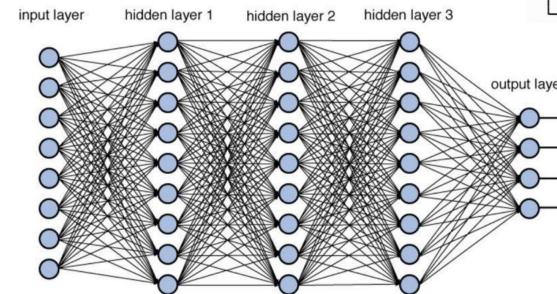
Advanced Architectures

9: Introduction to CNNs

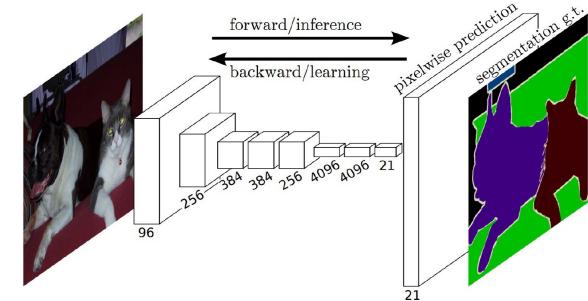
10: More on CNNs

11: Recurrent Neural Networks

12: Advanced Deep Learning Topics



Website & TUM-
online update
soon™



Exam FAQ

- Date

Who knows, probably early August

- Place

Probably on-site like the old usual exams, no remote exams or alternative available

- Retake?

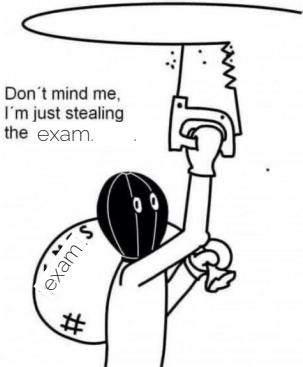
None this semester, can participate in the exam of next semester's class

Me waiting for the exam date



Exam FAQ

- Do we get access to previous exams?
Yes, including a mock exam etc. around week 10
- Do we have to code during the exam?
No, mostly short questions and some multiple choice
- What about my bonus?
Bonus from any previous semester will be transferred to this class and all future exam attempts (regardless of I2DL lecturer)
- For whom?
TUM/LMU (which will get a certificate)



Communication - Website

- <https://dvl.in.tum.de/teaching/i2dl-ss22/>
Slides, announcements, dates etc.
- External student registration
On our website soon™

External (non-TUM) Students

We want to provide access to our lecture for as many students as possible. If you are affiliated with TUM (e.g. LMU student, Ph.D. student, TUM student who cannot register for courses yet but have a TUM token, etc.), we will add you to our class manually. More information on how to register manually will be announced at the beginning of the lecture.

- Mailing List: i2dl@dvl.in.tum.de
Only for personal questions, don't contact the lecturers personally

Communication - Moodle

- Previously: Piazza
Neat features such as collaborative answers, etc.

- This semester: Moodle
 - Supervised by our TAs
 - We need your help for
 - Suggestions
 - Participation
 - Both questions and answers

⊕ [25. April - 1. May: Organization](#) 

⊕  [Moodle Enhancement Suggestions](#) 



Accurate depiction of an I2DL student that does everything alone

Communication - Office Hours

- Tentative Dates

Monday 10-11 & 17-18.30

Thursday 8-9

Friday 16-17.30



- Via zoom

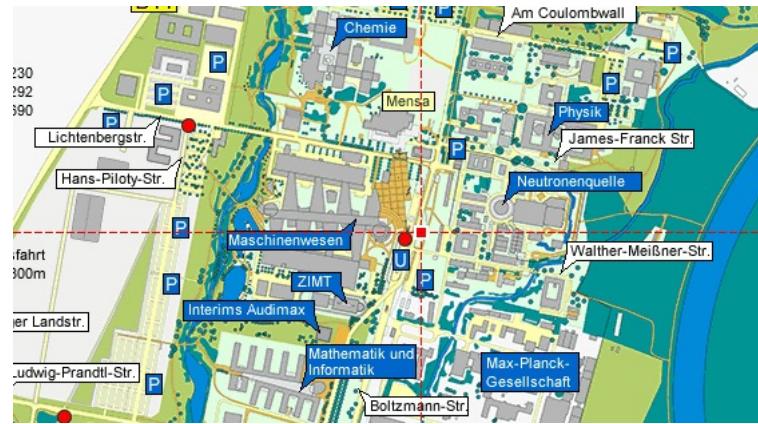
Links will be posted on moodle

- Starting from next week

M	T	W	T	F	S	S
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

Communication - Q&A Sessions

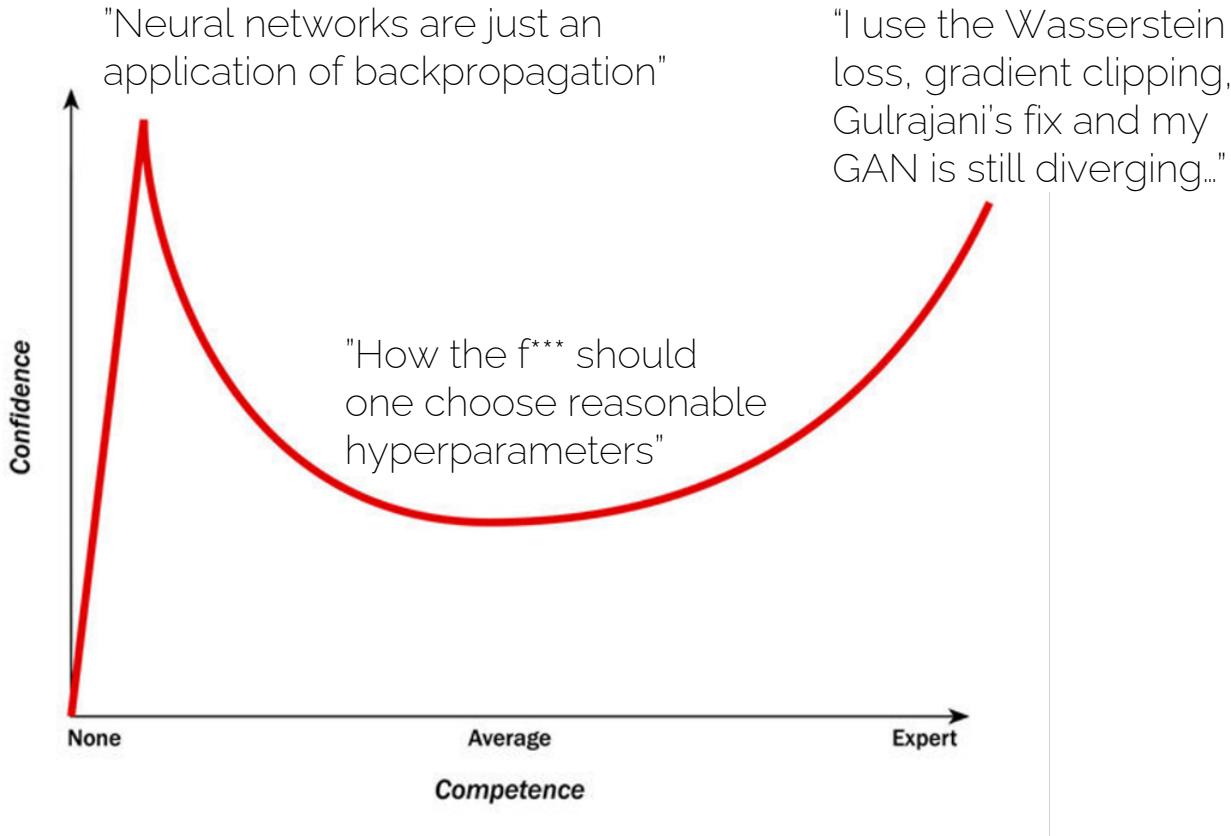
- Where?
Galileo Audimax
stream/vod as usual
- Content
Questions from audience and forums
- Tentative Dates
During exercise sessions on Tuesday
 - 21.06.22, 14-16 c.t.
 - 12.07.22, 14-16 c.t.
 - 26.07.22, 14-16 c.t.



General Exercises & Bonus

But first: questions?

From my own experience



Exercise – Tentative Schedule

Lectures

- Introduction
- 0: Organization
- 1: What is Deep Learning?
- 2: Machine Learning Basics

Neural Networks

- 3: Introduction
- 4: Optimization
- 5: Stochastic Gradient Descent
- 6: Training Neural Networks
- 7: Training Neural Networks
- 8: Training Neural Networks

Advanced Architectures

- 9: Introduction to CNNs
- 10: More on CNNs
- 11: Recurrent Neural Networks
- 12: Advanced Deep Learning Topics

Exercises

- Exercise 01: Organization
- Exercise 02: Math Recap

~~one week break~~

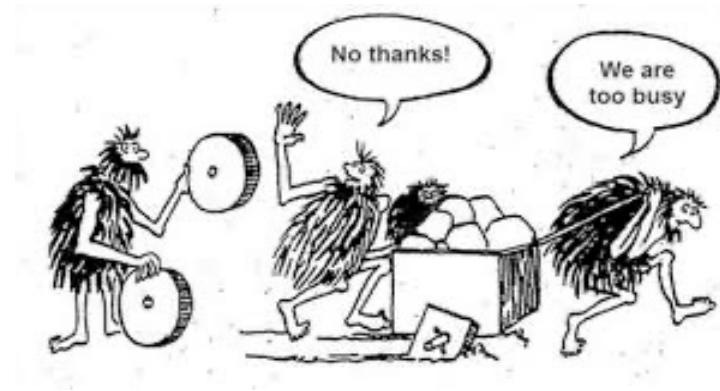
- Exercise 03: Datasets
- Exercise 04: Linear Regression
- Exercise 05: Neural Networks
- Exercise 06: Hyperparameter Tuning

- Exercise 07: Introduction to Pytorch
- Exercise 08: Autoencoder

- Exercise 09: Convolutional Networks
- Exercise 10: Semantic Segmentation
- Exercise 11: Recurrent Networks

Exercise - Goal

- Goal
Gather enough experience to start your own individual (research) project
- Focus
 - Reimplementations
 - Introduction to common libraries
 - Applications



Exercise - Format

- Presentation: video only
Posted at live.rbg.tum.d
Slides uploaded to moodle/website
- Video length
Ex02: Full lecture with written exercise
Ex03-11: Short (~30min) video and coding exercises
- Exam
All slides, coding notebooks are exam relevant,
no coding during exam
- Main work
Interactive coding notebooks (~4h each)

Introduction to our Submission System

In this exercise you will:

- Optional: Learn how to set up Google Colab for our exercises
- Learn how to work with IPython notebooks
- Get a small numpy introduction
- Learn about our submission system

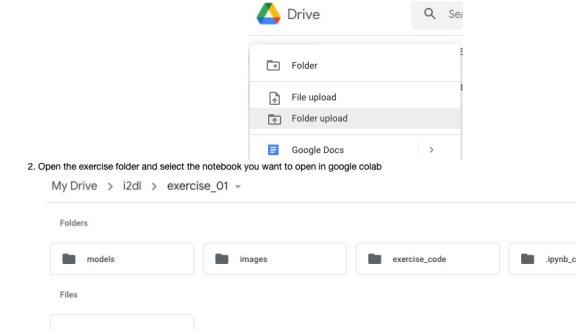
Execute a selected cell with `Shift + Enter`.

(Optional) Set up Google Colab

If you lack resources CPU or GPU resources to finish an exercise, you can also use your google account to access [google.colab](#) which sets up python environment as well as free resources for training.

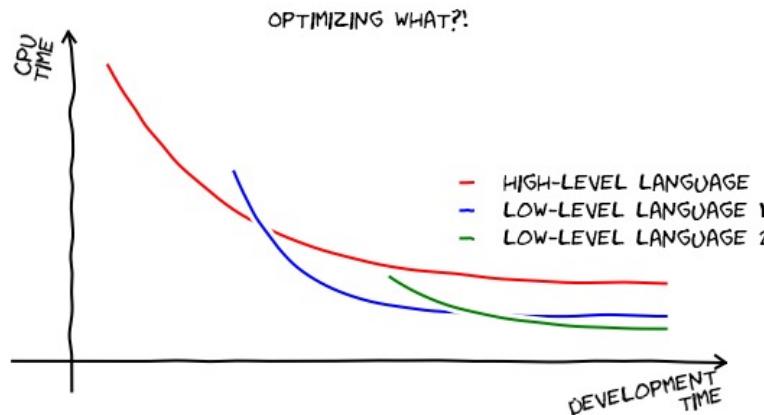
In order to use google colab follow these steps:

1. Access your [google drive](#) and upload your i2di folder that contains the respective exercises/datasets etc. in the main directory



Why Python?

- Very easy to write development code thanks to an intuitive syntax
- Biggest language used in deep learning research



Exercises – Tech Stack

- Python
Jupyter notebooks
Numpy
- Deep Learning library
Pytorch
- Hardware requirements
 - Minimum: CPU
 - Preferred: Nvidia GPU
 - Alternative: Google Colab



Extra Spice: Leaderboard (Ex6, g-11)

- Compete against your fellow students on our datasets for a variety of tasks
- Some of the best solutions will be featured in exercise discussions

Leaderboard

The leaderboard shows for each exercise the highest scoring submission from each user. Only valid submissions are displayed.

Exercise 1	Exercise 3	Exercise 4	Exercise 5	Exercise 6	Exercise 7	Exercise 8	Exercise 9	Exercise 10	Exercise 11
#	User	Score							
1	a0005	68.77							
2	a0010	64.41							

Exercises - Content

Exercise 01: Organization
Exercise 02: Math Recap

Intro

Exercise 03: Dataset and Dataloader
Exercise 04: Solver and Linear Regression
Exercise 05: Neural Networks
Exercise 06: Hyperparameter Tuning

Numpy
(Reinvent the wheel)

Exercise 07: Introduction to Pytorch
Exercise 08: Autoencoder

Pytorch/Tensorboard

Exercise 09: Convolutional Neural
Networks
Exercise 10: Semantic Segmentation
Exercise 11: Recurrent Neural Networks

Applications
(Hands-off)

Exercises – 9 Submissions

Exercise 01: Organization

Exercise 02: Math Recap

Intro

Exercise 03: Dataset and Dataloader

Exercise 04: Solver and Linear Regression

Exercise 05: Neural Networks

Exercise 06: Hyperparameter Tuning

Numpy
(Reinvent the wheel)

Exercise 07: Introduction to Pytorch

Exercise 08: Autoencoder

Pytorch/Tensorboard

Exercise 09: Convolutional Neural
Networks

Exercise 10: Semantic Segmentation

Exercise 11: Recurrent Neural Networks

Applications
(Hands-off)

Submissions & Bonus

- Submissions
 - 8 practical exercises, labelled as submissions
 - submissions have a fixed, mostly one week due date until they have to be solved and successfully uploaded.
- Bonus
 - Passing exam grade get's a -0.3 improvement
 - Need to pass 8/9 submissions

How to pass this class

- The Efficient
Binges the lecture videos at x2 speed
Reads through all lecture notes, exercise slides and screenshots from their friend's jupyter notebook
- The 'I just want to do applications'
Visits every lecture, reads through exercise material
Works through the pytorch exercise and applications during the semester break
- The Alpha
Works through weekly lecture and exercise content, gets passing submissions in, asks questions on moodle and tries to answer other student's posts

Exercise 1

But first: questions?

Overview

- Start date/availability
Tomorrow Tuesday, 26.04.22, at 14.00
Website/moodle
- Submission deadline
Monday 16.05.22, 23.59 – no exceptions!
- Content
 - First experience with our submission system
 - Optional: python refresher

Instructions

- Download the zipped folder tomorrow
- Extract locally and at best do both
 - Install anaconda to create a new python environment as outlined in README.md
 - Set up your repository using the requirements.txt file
 - Execute `jupyter notebook` in the i2dl folder

New python users: <http://nbviewer.jupyter.org/github/irjohansson/scientific-python-lectures/blob/master/Lecture-1-Introduction-to-Python-Programming.ipynb>

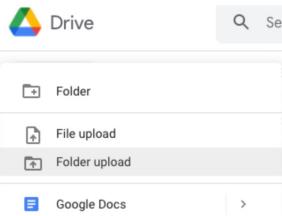
Google Colab Instructions

(Optional) Set up Google Colab

If you lack resources CPU or GPU resources to finish an exercise or have issues setting up a local python environment, you can also use your google account to access [google colab](#) which provides an already set up python environment as well as free resources for training.

In order to use google colab follow these steps:

1. Access your [google drive](#) and upload your i2dl folder that contains the respective exercises/datasets etc. in the main directory



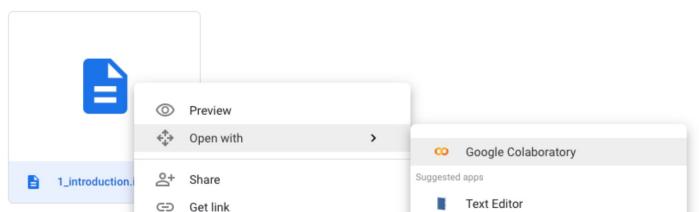
2. Open the exercise folder and select the notebook you want to open in google colab

My Drive > i2dl > exercise_01 ▾

Folders



Files



Hitchhiker's Guide: Notebooks

1. Run cells from top to bottom
2. Be careful when changing notebook cells
3. Edit external code in browser or in your favorite IDE
 - Don't code outside our boxes in exercise_code files
4. Checking other code
 - Generally optional
 - Look out for green boxes

Task: Check Code

Please read `make_dataset(directory, classes)` tasks. Additionally, it would be wise decision to get projects. As it is not beginner friendly, we removed it.

Exercise Submission

- Register at our submission webpage
 - Sign up with valid matriculation number
 - Get id and password via mail from tum-online (it will display the email address)
- Submit models with
 - Upload created zip file
 - Note: You will submit your whole code folder as well

FAQ

- I don't want to code in notebooks. Can I use my favourite IDE?
 - Yes
- Cool, so I can just change the whole code structure?
 - No
 - You can write any helper functions, but keep the skeleton classes intact (i.e., don't rename important functions or variables)
 - You will upload all files and those will be archived on our end

FAQ

- How do I know that I passed?
 - Once you submit a score that surpasses the threshold, you will receive an email that contains a message which tells you that you passed this submission
- Help, I got this message a second time?
 - You will receive this message every time you submit an exercise that exceeds the score
 - Your best performing model will count, not the latest submitted one

See you next week

