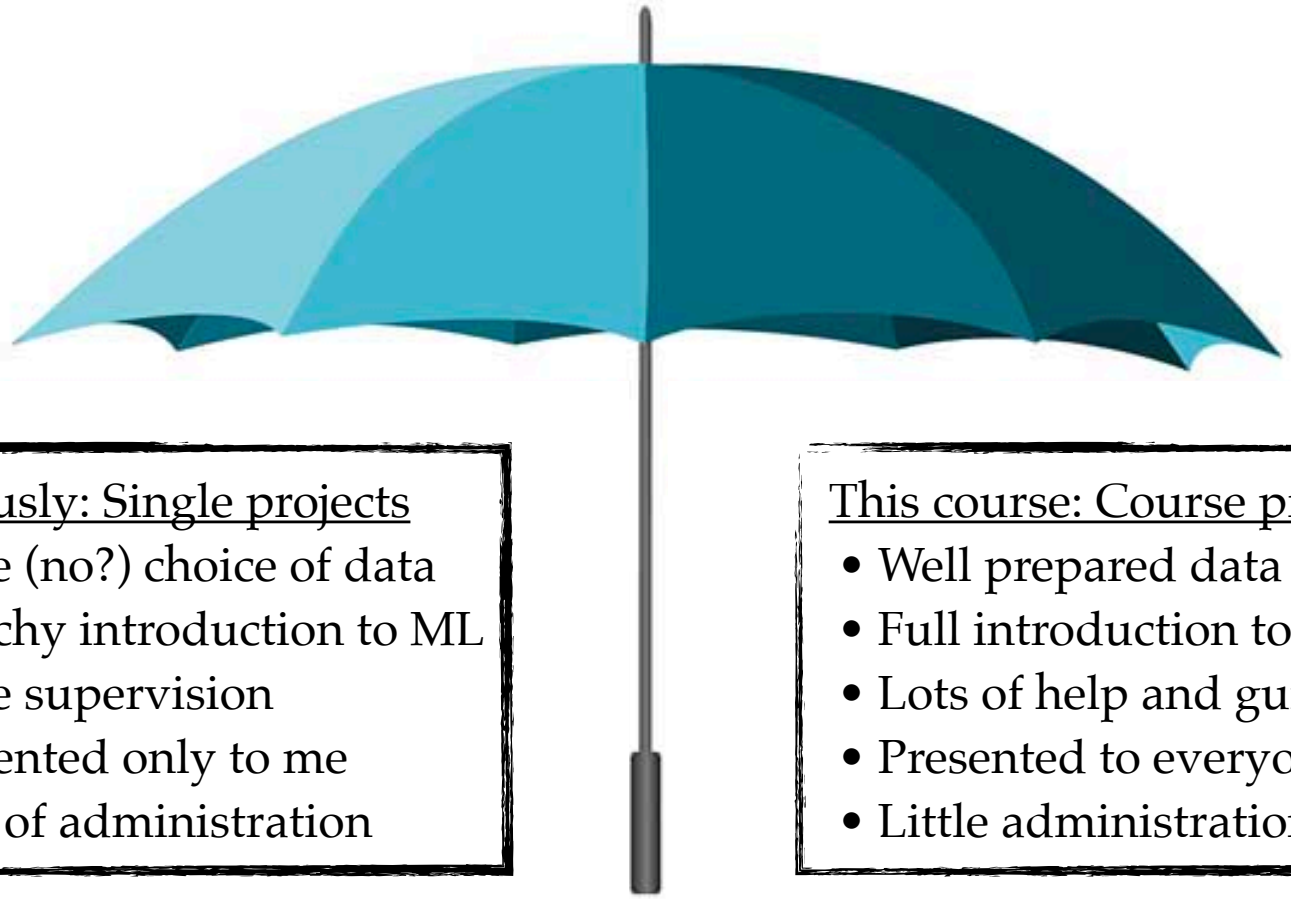
Applied Machine Learning

Course information 2023

Troels C. Petersen, Charles Steinhardt,
Azzurra d'Alessandro, Arnau Morancho, Thomas Spieksma

This course is (partially) an "umbrella course" for doing projects using ML

An "umbrella course"



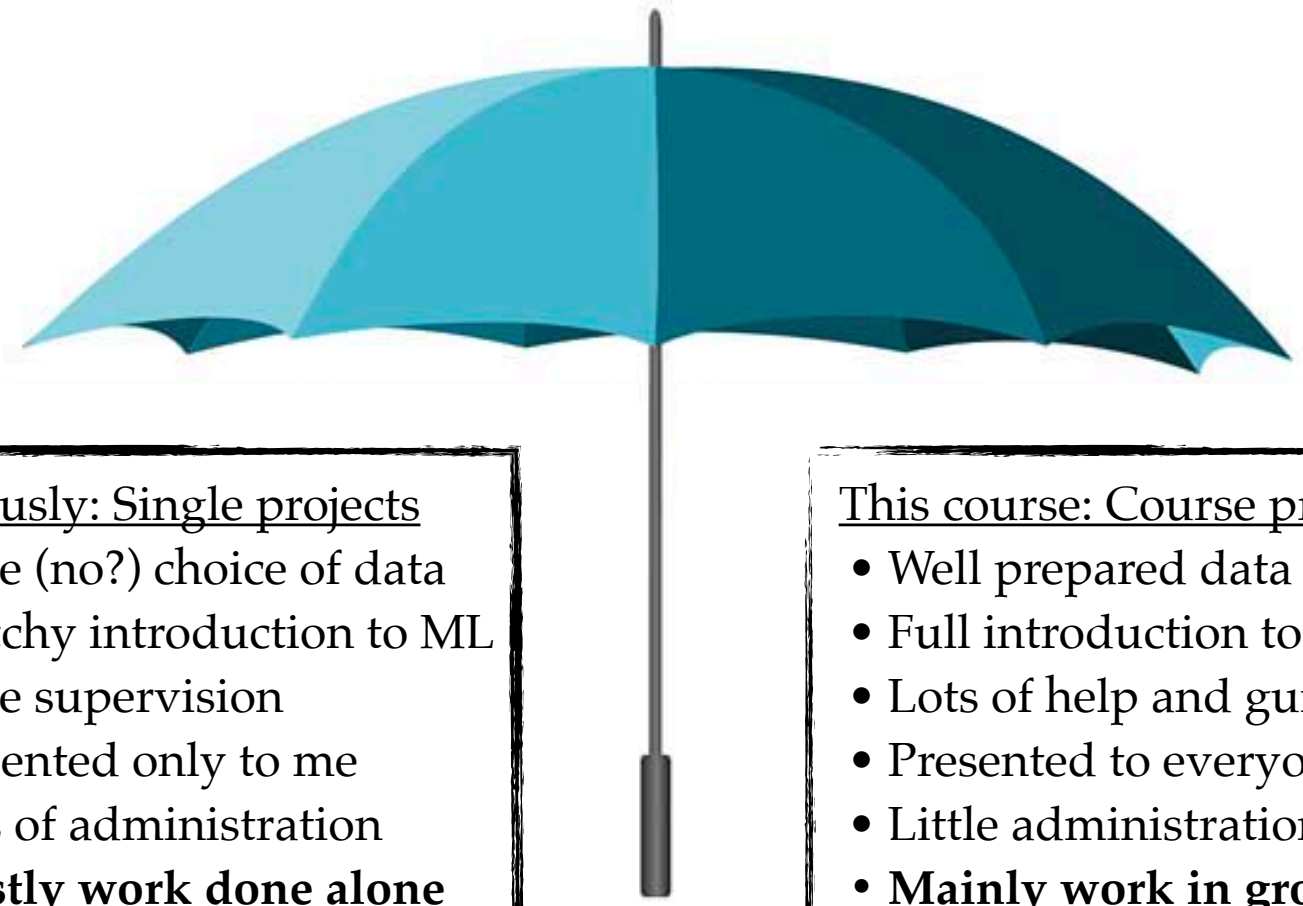
Previously: Single projects

- Little (no?) choice of data
- Sketchy introduction to ML
- Little supervision
- Presented only to me
- Lots of administration

This course: Course projects

- Well prepared data cases
- Full introduction to ML
- Lots of help and guidance
- Presented to everyone
- Little administration

An "umbrella course"



Previously: Single projects

- Little (no?) choice of data
- Sketchy introduction to ML
- Little supervision
- Presented only to me
- Lots of administration
- **Mostly work done alone**

This course: Course projects

- Well prepared data cases
- Full introduction to ML
- Lots of help and guidance
- Presented to everyone
- Little administration
- **Mainly work in groups**

General course running

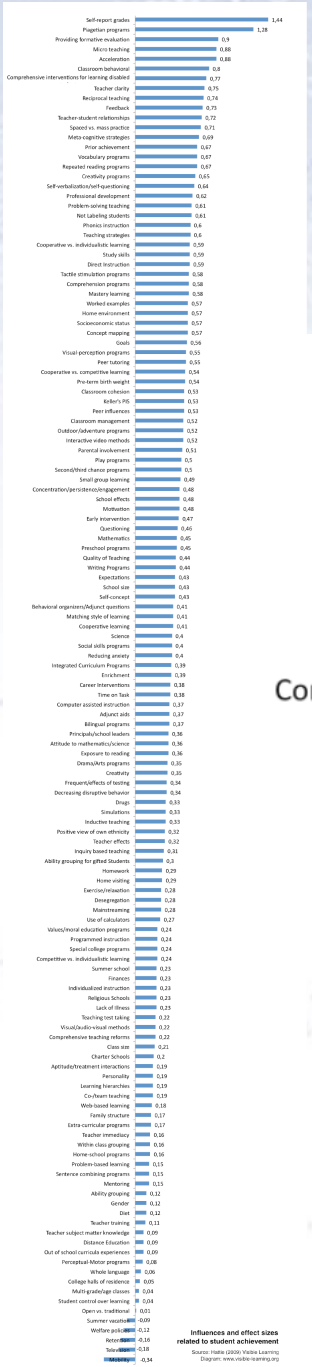
We will run the course **in person (only)**. However, most lectures will be linked to zoom videos from previous years and/or (poorly) recorded for posting.

The course require both self-disciplin and dedication.

We will of course do our best to inspire, help, debug (?), and promote collaboration, but it is of course up to you, how much you want to learn/ benefit from this course.

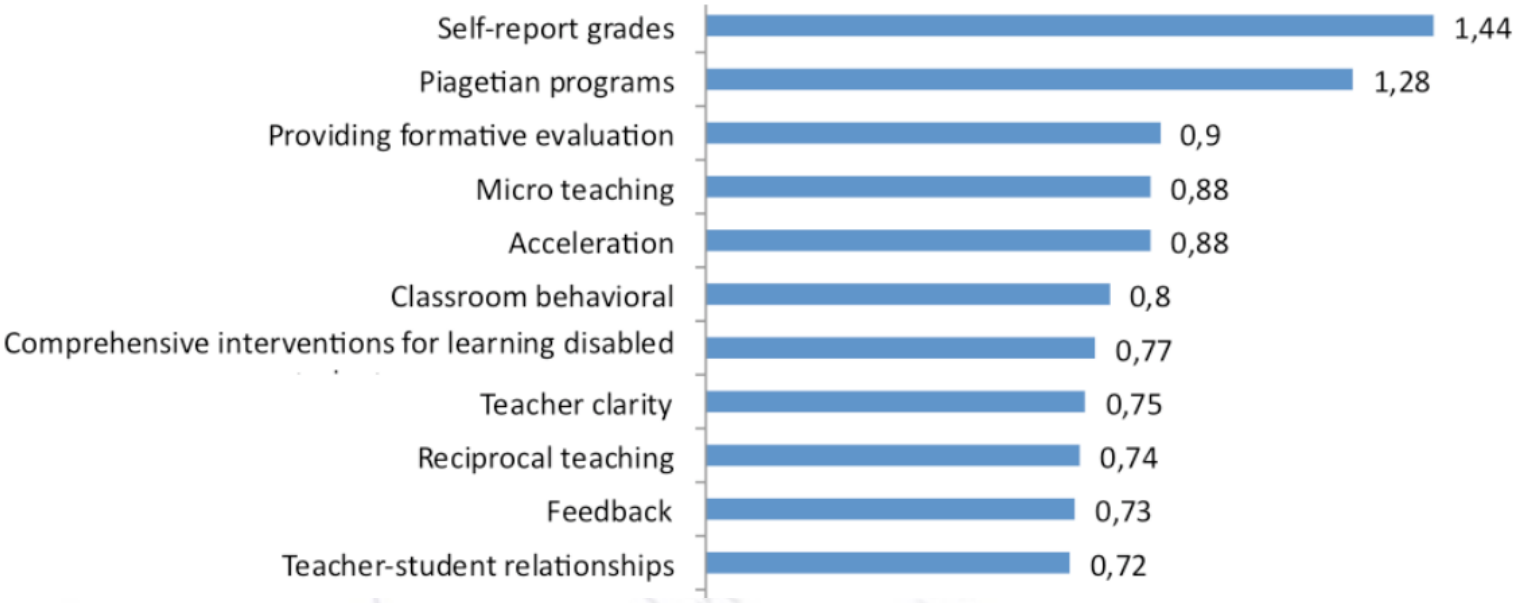
Course work can/should be done in collaboration with fellow students.

Please make small teams of peers, with whom you can discuss the many details of ML coding and the problems, challenges, and issues involved. This is you best way of discussing with peers, **learning most**, and not getting stuck.



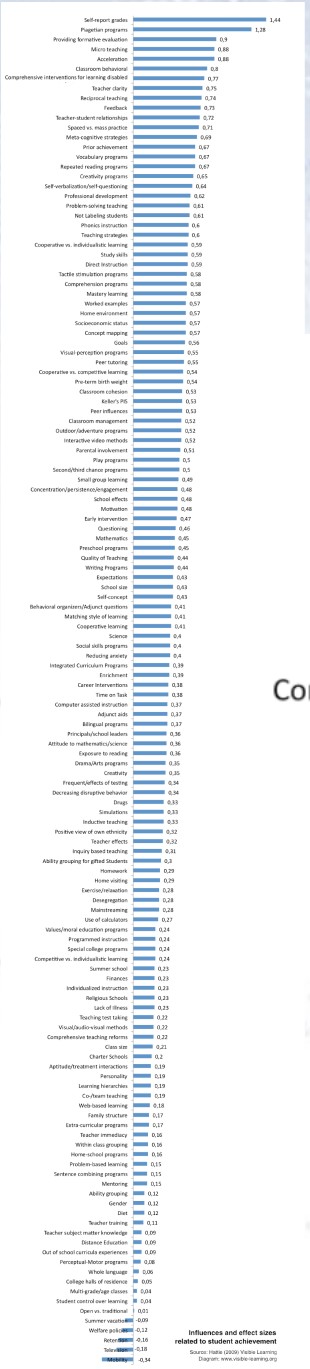
What influences learning?

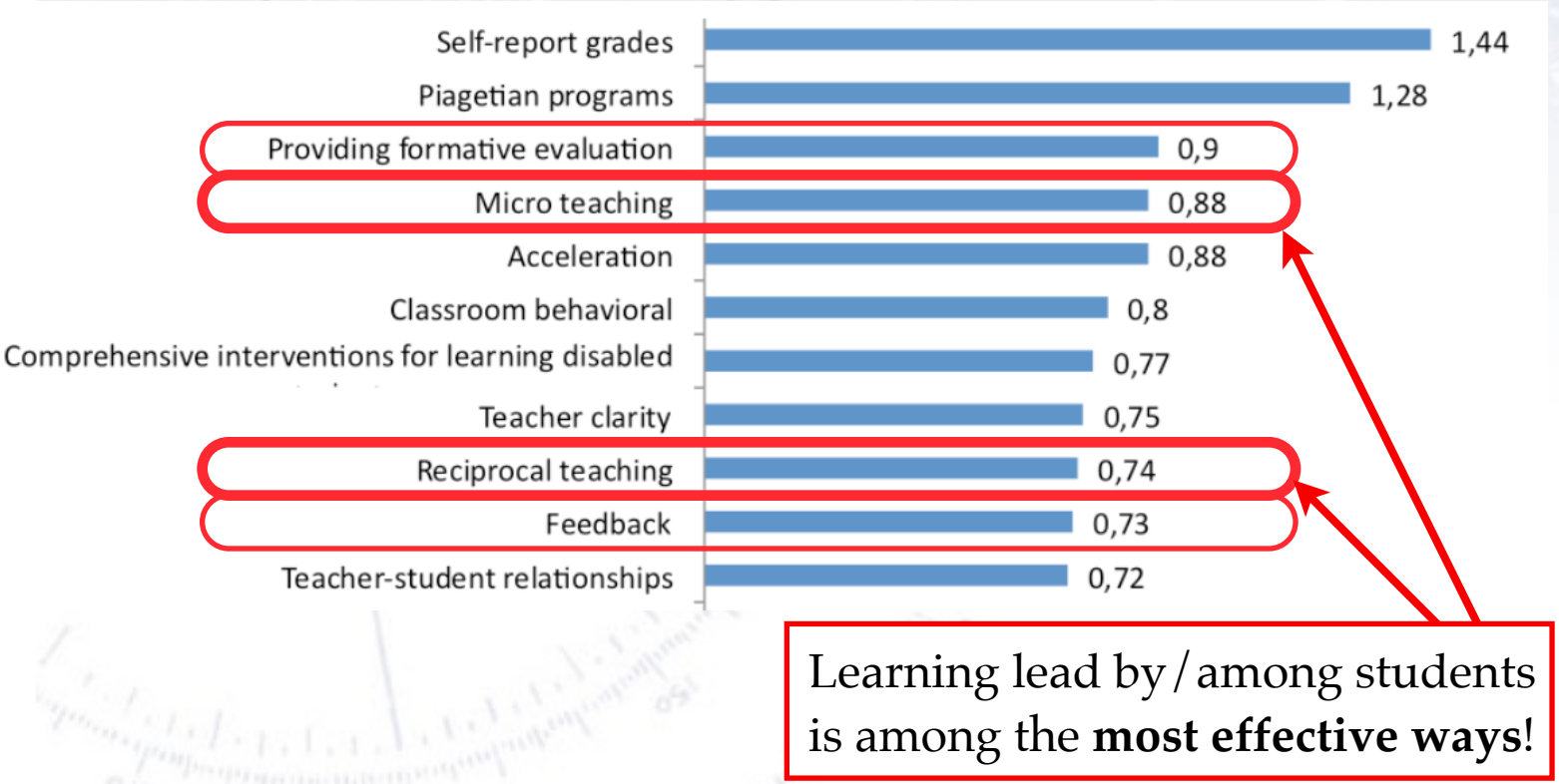
There are studies of this, one result shown below:



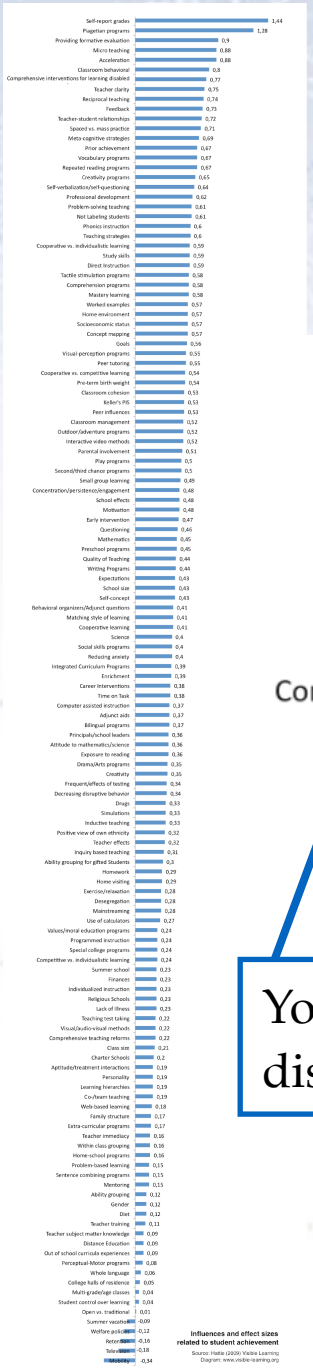
What influences learning?

There are studies of this, one result shown below:



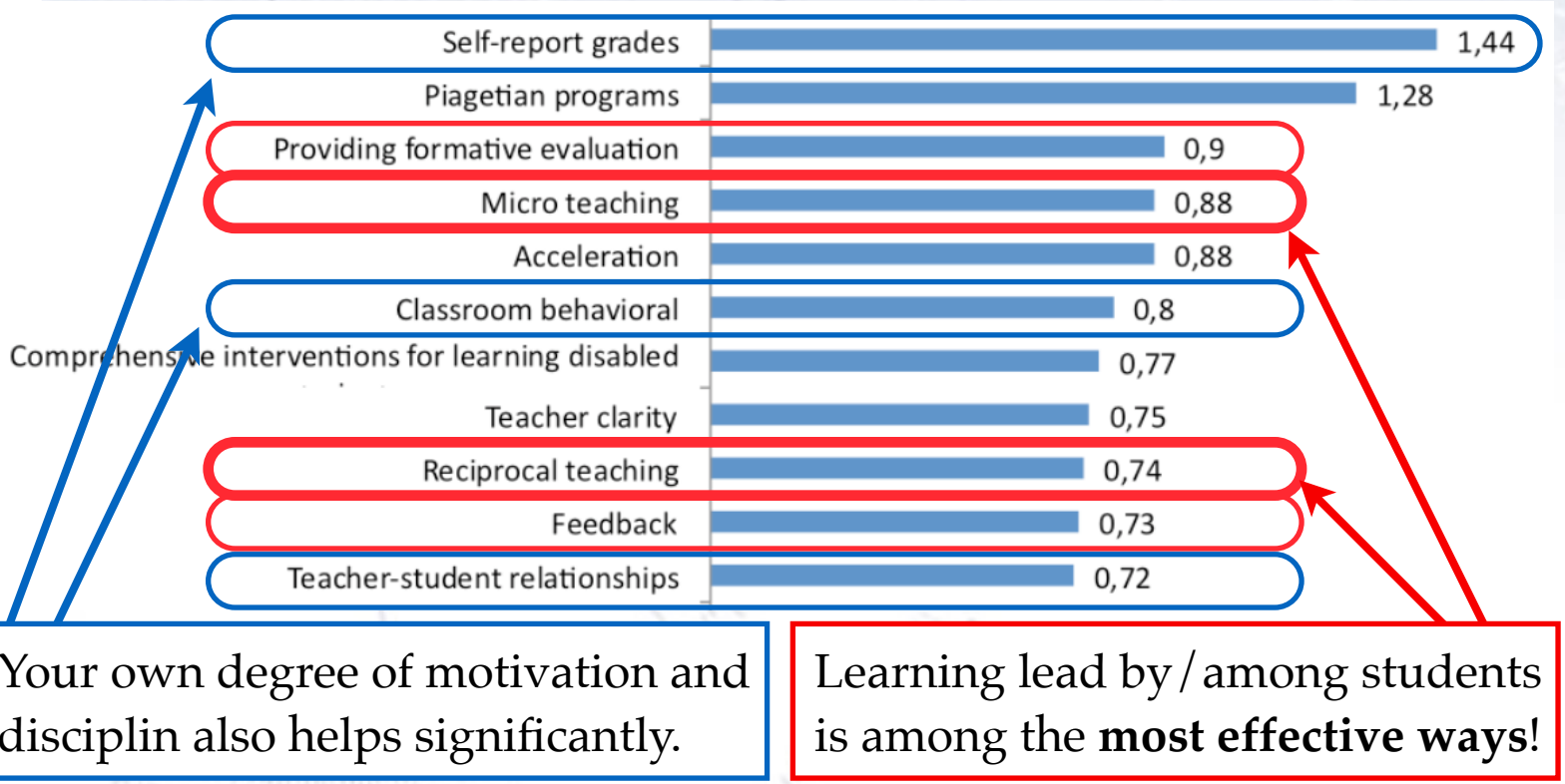


Learning lead by/among students is among the **most effective ways**!



What influences learning?

There are studies of this, one result shown below:

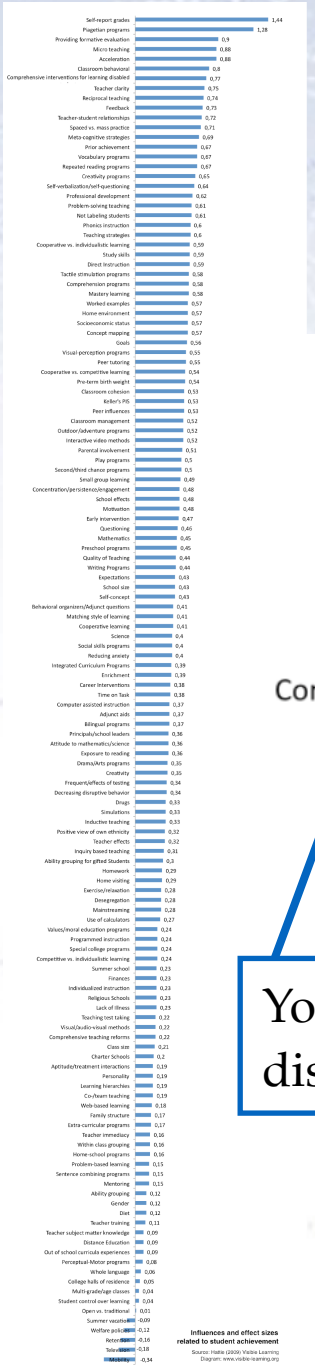


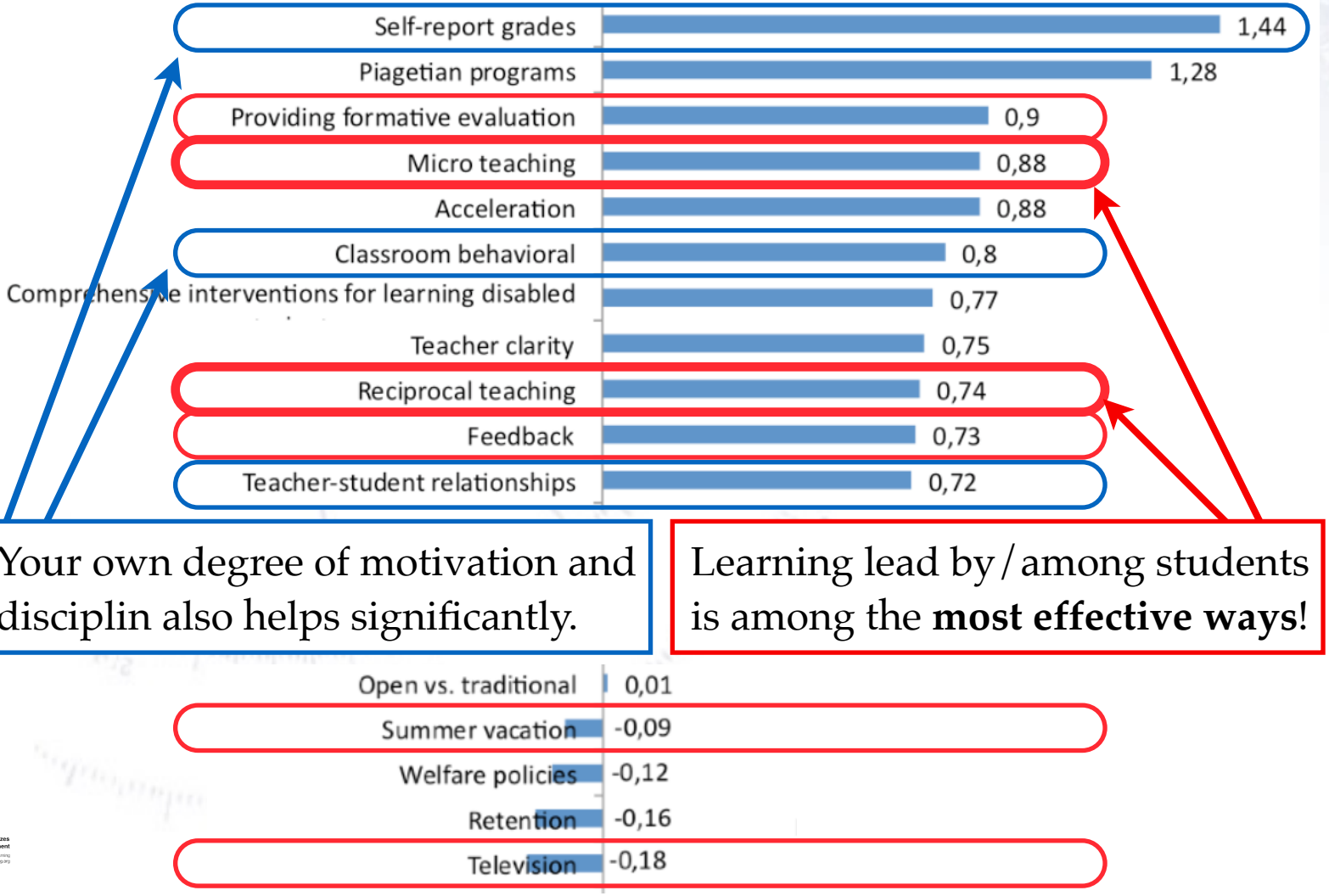
Your own degree of motivation and disciplin also helps significantly.

Learning lead by/among students is among the **most effective ways**!

What influences learning?

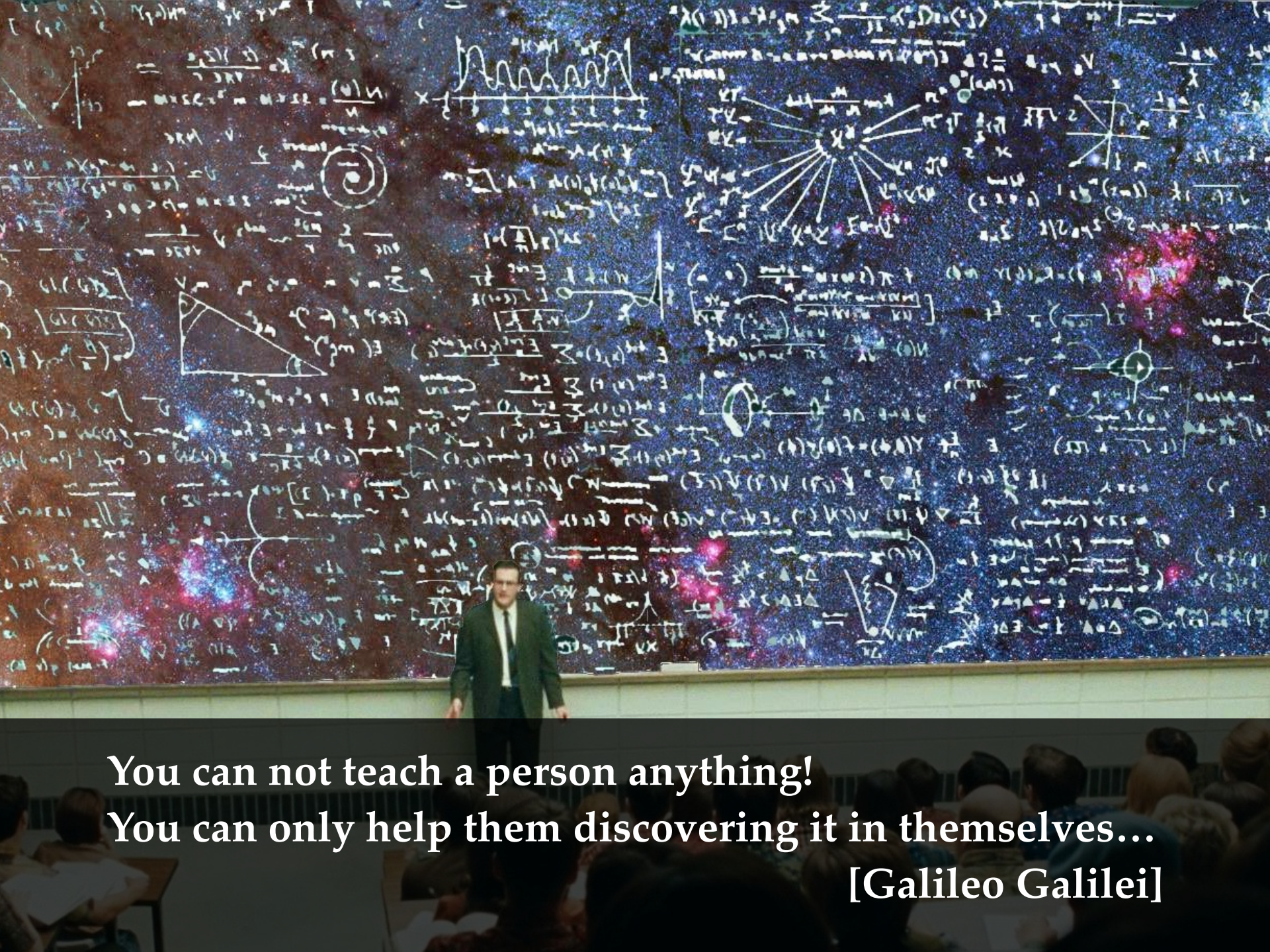
There are studies of this, one result shown below:





Your own degree of motivation and disciplin also helps significantly.

Learning lead by/among students is among the **most effective ways**!



You can not teach a person anything!
You can only help them discovering it in themselves…
[Galileo Galilei]

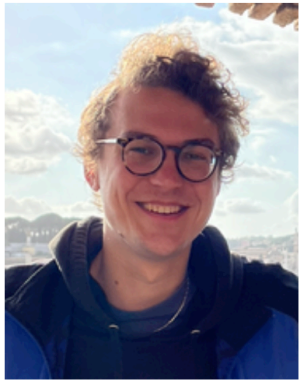
Who are we?

Troels C. Petersen
Lecturer - Associate Prof.
NBI - High Energy Physics
35 52 54 42 / 26 28 37 39
petersen@nbi.dk

Charles L. Steinhardt
Teacher - Associate Prof.
NBI - Cosmology
27 51 44 47
steinhardt@nbi.ku.dk

Azzurra d'Alessandro
Teaching assistent - Ph.D.
NBI - Astrophysics
42 45 39 54
azzurra.dalessandro@nbi.ku.dk

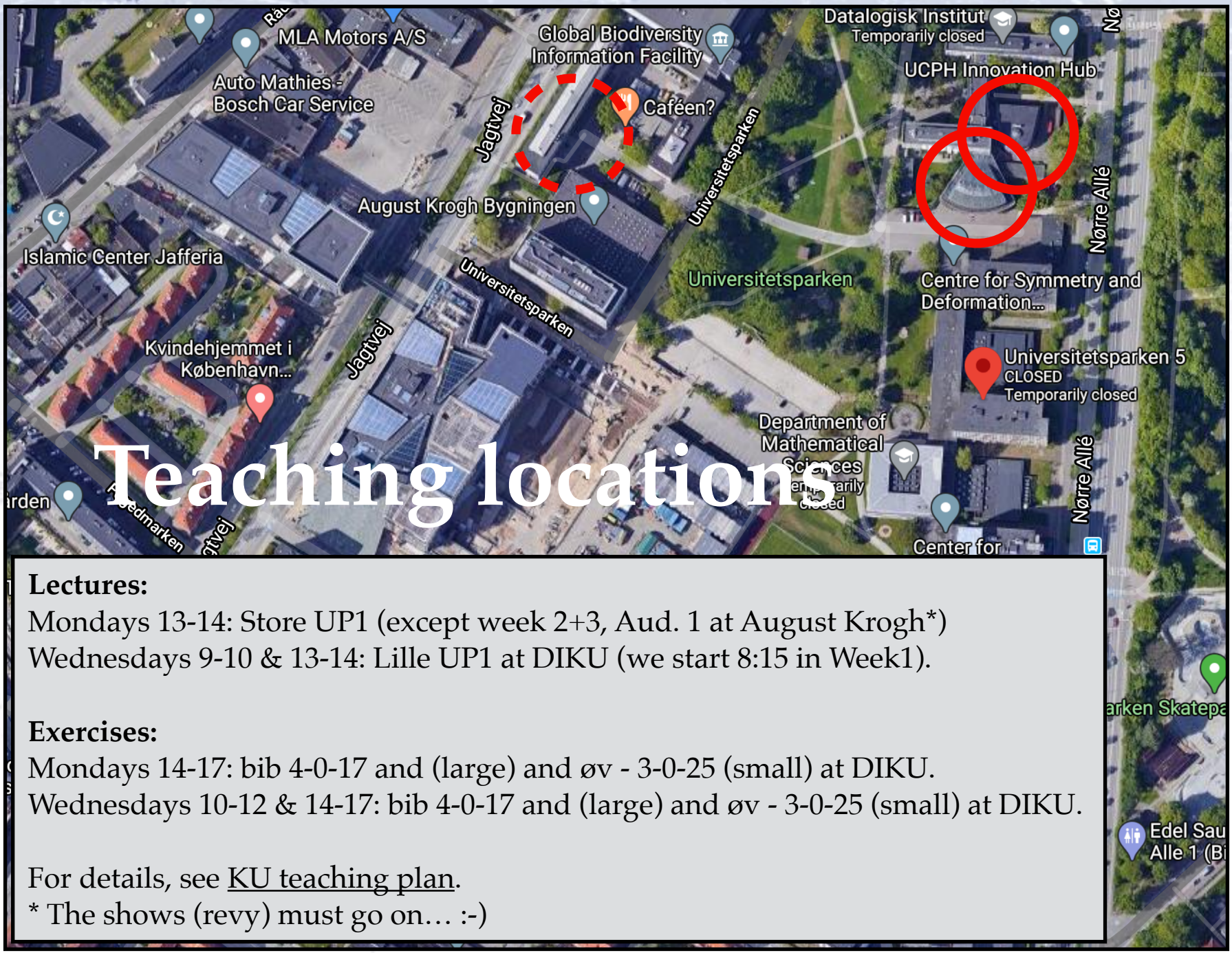
Arnau Morancho Tarda
Teaching assistent - Ph.D.
NBI - High Energy Physics
+34 629 350 151
arnau.morancho@nbi.ku.dk



Thomas F. M. Spieksma
Teaching assistent - Ph.D.
NBI - Gravitational Physics
thomas.spieksma@nbi.ku.dk

You are more than welcome to contact os anytime. Email is preferable to most of us, but in particular for coding, the Slack channel is probably best.

Our aim is for YOU to learn the most about ML and using it. If you have any wishes or feedback, don't hesitate to write us.



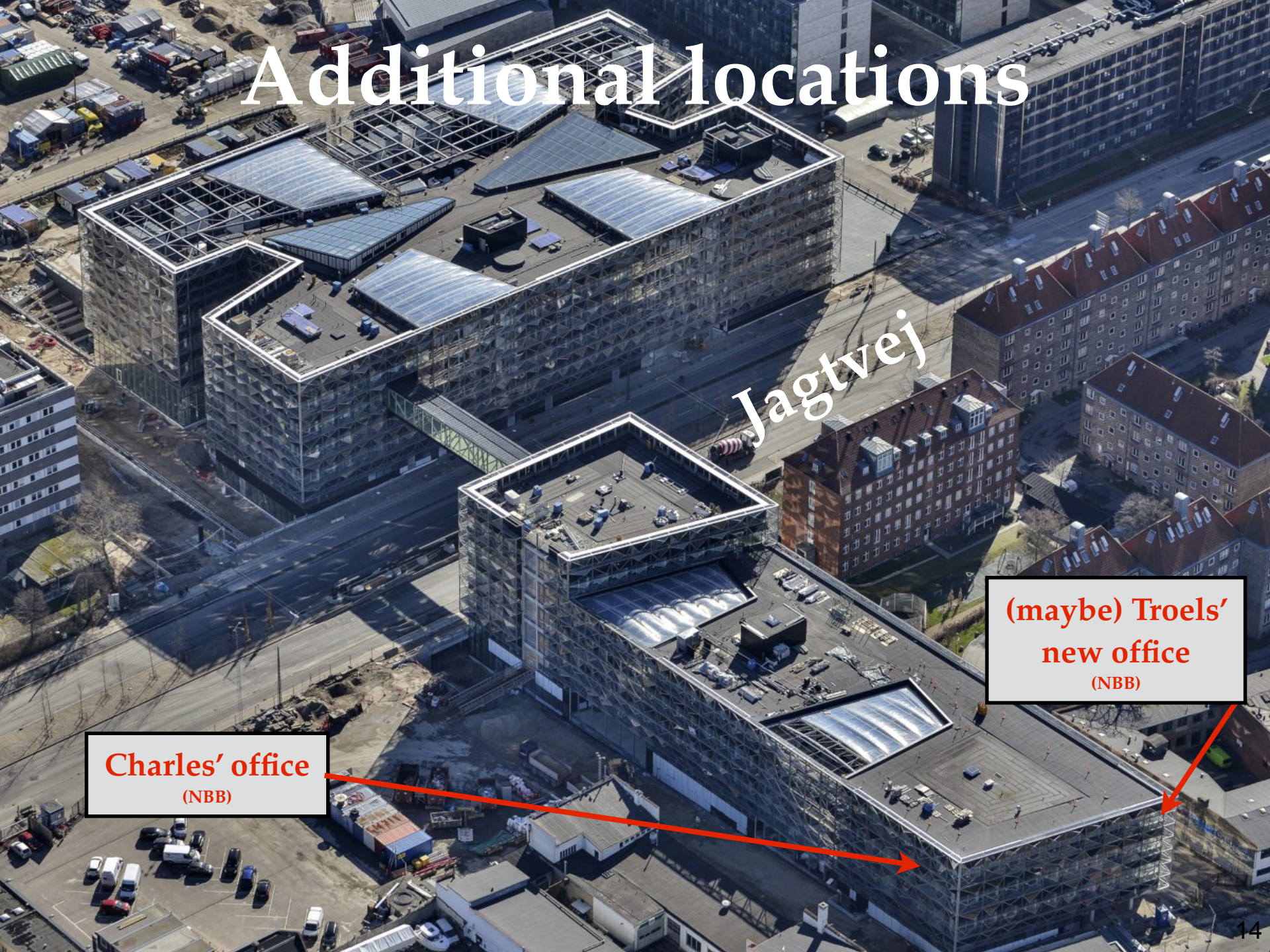
Lectures:
Mondays 13-14: Store UP1 (except week 2+3, Aud. 1 at August Krogh*)
Wednesdays 9-10 & 13-14: Lille UP1 at DIKU (we start 8:15 in Week1).

Exercises:
Mondays 14-17: bib 4-0-17 and (large) and øv - 3-0-25 (small) at DIKU.
Wednesdays 10-12 & 14-17: bib 4-0-17 and (large) and øv - 3-0-25 (small) at DIKU.

For details, see KU teaching plan.
* The shows (revy) must go on… :-)

Lecture locations
Mondays Week 2+3
Aud. 1
Up the stairs on the right!
Entrance
AUGUST KROGH
KROGHS KØKKEN

Additional locations



Computers and software

We will program in Python. You may **choose as you wish**, but we highly recommend Python. We will only provide data, code snippets (in Python), and occasional code/solutions for inspiration.

We suggest that you use Jupyter Notebook, and run everything on your own laptop, possibly with ERDA as a backup. For GPU access, Google Colab is (so far) the only thing we can refer to. We also recommend that you use GitHub.

Data files will typically be provided in CSV and/or HDF5 format, but others might be used.

We will be using several Python packages (see course webpage), introduced along the way, and surely you have your own favourites. Use them happily (but knowingly).

Projects / Exam

This course is to some extend an umbrella course for projects using ML.

We will be doing two projects:

- An individual "initial project" on **common data** (2 weeks - 40% of your grade).
- A group "final project" on **data of your choice** (3 weeks - 60% of your grade).

The **initial project** will be the basic applications of ML (classification, regression, and clustering) to a data set, and we will evaluate your (algorithm's) performance on a test set (not unlike Kaggle competitions).

The **final project** will be your main task, and is the application of ML on "anything that you like". You will all be presenting your results to each other, so that also others may learn from what you did (and didn't).

You can find much more information about both projects on the course webpage (so please go and read it carefully at least once!):

- Initial project (to be submitted individually).
- Final project (to be submitted in groups).

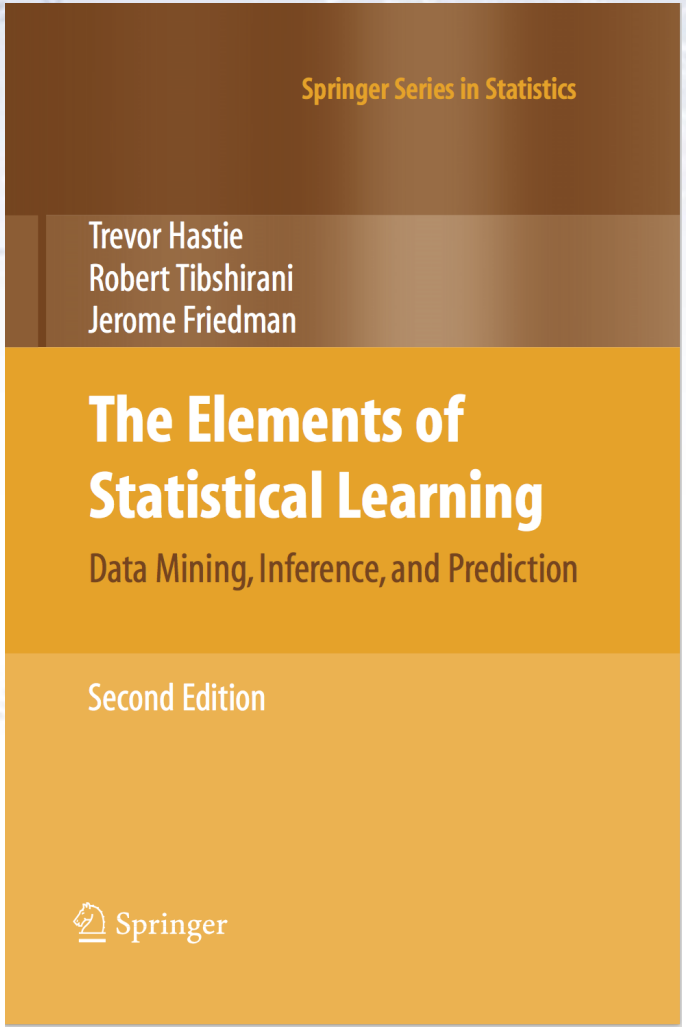
Literature

The main literature will be slides, notes, blogs, and links! However, we also wanted you to have a few more "solid" places to read comprehensively about ML.

"The Elements of Statistical Learning" (TEOSL) is a good read in PDF (though at times rather mathematical), and especially chapter 2 is a good introduction.

"Applied Machine Learning" by David Forsyth in a GREAT book in PDF, which is spot on the subject and at times even mildly entertaining.

"Deep Learning" by Ian Goodfellow et al. in HTML is also good, and Chapter 5 of Part I gives a great overview of ML and its ingredients.



Blogs as literature

In ML, blogs/articles/tutorials are a very common (and great) source of literature on ML. For this reason, we've made a list of links that we find good:

nbi.dk/~petersen/Teaching/ML2023/MLlinks.html

Applied Machine Learning 2023 - Useful ML links

The field of Machine Learning (ML) is developing at a very fast pace and by an expanding number of practitioners. For this reason, text books on ML are typically few and slightly depricated, though they of course describe the general concepts very well. Research papers have partially filled the gap, as these are more versatile and frequently updated. However, they typically only deal with a very small and specific part of the generel ML "phase space".
But whereas "classic" literature (i.e. books and papers) only covers partially, blogs and github repositories seem to fill out the rest, and typically in a much more accessible fashion with great illustrations and example code. For this reason, we have tried to gather some of the more useful links to such blogs and repositories below. It is simply our (slightly random) selection of webpages that we have come across and found illustrative and useful. So use these at will, and also build your own list of reference sites.

Books:
- Deep Learning by Ian Goodfellow et al. (2016). A short and good general introduction to ML can be found in Chapter 5 of Part 1.
- Pattern Recognition and Machine Learning by Christopher M. Bishop (2006).
- "Interpretable Machine Learning" by Christoph Molnar (2020). A Guide for Making Black Box Models Explainable.
- "Convolutional Neural Networks for Visual Recognition" by Andrej Karpathy (2017?). Used for teaching CNN in Stanford's cn231 class.
- "Deep Learning with Python" by author of Keras, Francois Chollet, now at Google AI (2017). Especially chapter 4 is a good overview of the fundamentals of Machine Learning.
- Book on neural networks and deep learning. Online book only!
- Applied Machine Learning by David Forsyth. Great book, which is at times even mildly entertaining, from one of the masters of ML vision.

Papers:
- XGBoost paper (2016). Highly readable paper showing the innovations of the XGBoost algorithm.
- LightGBM paper (2017). Explaining the great speedup and showing examples of execution times.
- More general paper (100 pages) on Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges (2021).
- ROCKET: Time series classification using random convolutional kernels
-

Blogs/Links/Tutorials - supervised learning:
- **Introduction to tree based learning**. Very good introduction to the basics of tree based learning.
- **Introduction to neural net based learning**. Very good introduction to the basics of Neural Net (NN) based learning.
- SciKit Learn tutorial. Gives a quick introduction to ML in general and has code examples for SciKit Learn.
- Simple introduction code to a simple Neural Network in PyTorch.
- XGBoost vs. LightGBM. Discussion of differences, with code examples.
- XGBoost, LightGBM, and CatBoost. Discussion of differences and hyperparameters.
- Introduction to NGBoost, which is a tree based algorithm, which makes a probabilistic predictions (i.e. uncertainties).
- PyTorch Geometric Graph Neural Network Tutorial (2019). Reasonably good guide with code examples.
- Survey of the performance of BDTs and NNs on tabulated data, essentially showing that tree based methods simply are very good at this!
-

Blogs/Links/Tutorials - unsupervised learning:
- SciKit-Learn manual for and discussion of (unsupervised) clustering.

Expectations alignment

Expectations

We want (read: insist) this course to be useful to all of you! Therefore, please give us feedback (the earlier the better), if you have anything to add/suggest/criticise/alter.

However, it is also a VERY independent course in the sense that it is up to YOU, how much you get out of it. Consider it as much a project as a course!

The aim is to make you knowledgable about the basics of Machine Learning, and being able to apply it to (suitable) data.

Problems?

If you experience problems in relation to the course, whatever their origin and nature, **then write me**!

We may not be able to do anything about it, but if I don't know about your problems, then I most certainly can not do anything about them.

We consider ourselves fairly large, as long as I feel that this largeness is met by sincerity and will.

But… you need to write in the first place! That is your responsibility.

Checklist

The following should be done before starting the course:

- Fill in the questionnaire
- Ensure that Python runs with the relevant packages
- Ensure that you got GitHub running

The following should hopefully be in place by the end of the first week:

- **Ensure, that you have fellow students to work with!**

Working on exercises

The exercises are meant to make you work through all the parts that make up "understanding and being able to use" Machine Learning.

If you don't understand the purpose or aim of the exercise with your peers, read through the exercise one more time (~5 minutes) and then ask us.

A. Work on exercise by yourself/in a group, preferably during exercise hours.

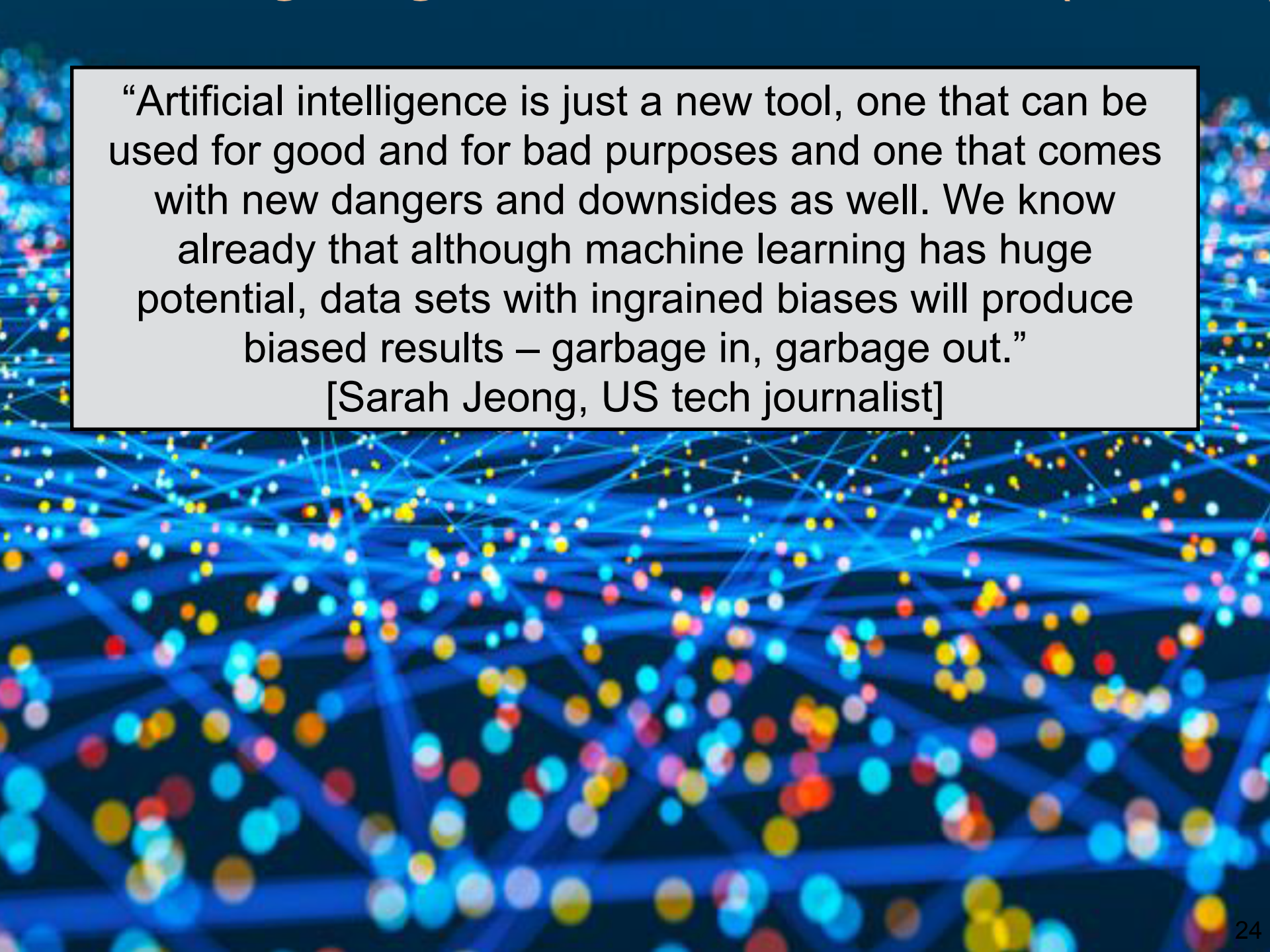
B. If you get stuck, discuss with your group/peers (for 5-10 minutes).
Of course also check "Dr. Google" and "Miss ChatGPT" for solutions!

C. If this doesn't solve it, call/write out in the chat and/or ask a TA.

D. If this doesn't solve it, we'll take it up in plenum.

E. If this doesn't solve it, we'll work out a solution for next class.

F. If this doesn't solve it, we'll write an ML paper on this interesting problem :-)



"Artificial intelligence is just a new tool, one that can be used for good and for bad purposes and one that comes with new dangers and downsides as well. We know already that although machine learning has huge potential, data sets with ingrained biases will produce biased results – garbage in, garbage out."
[Sarah Jeong, US tech journalist]