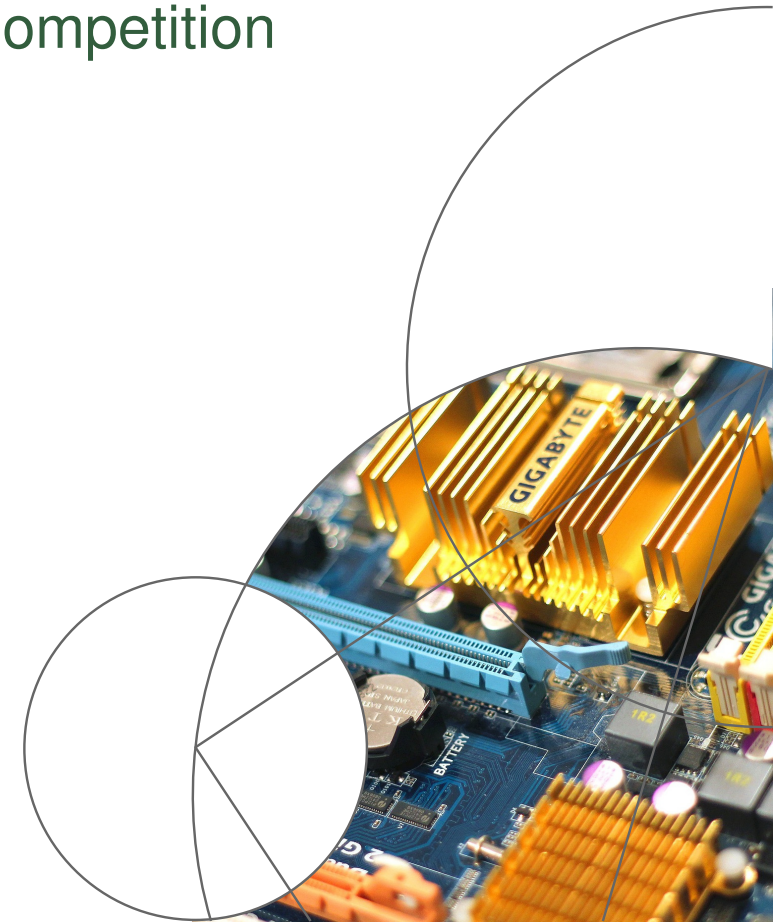
L1 – Introduction, Organization, and Competition

Large-Scale Data Analysis

Fabian Gieseke

Image Group
Department of Computer Science
University of Copenhagen

Universitetsparken 1, Room 1-1-N110
fabian.gieseke@di.ku.dk



23 April 2019

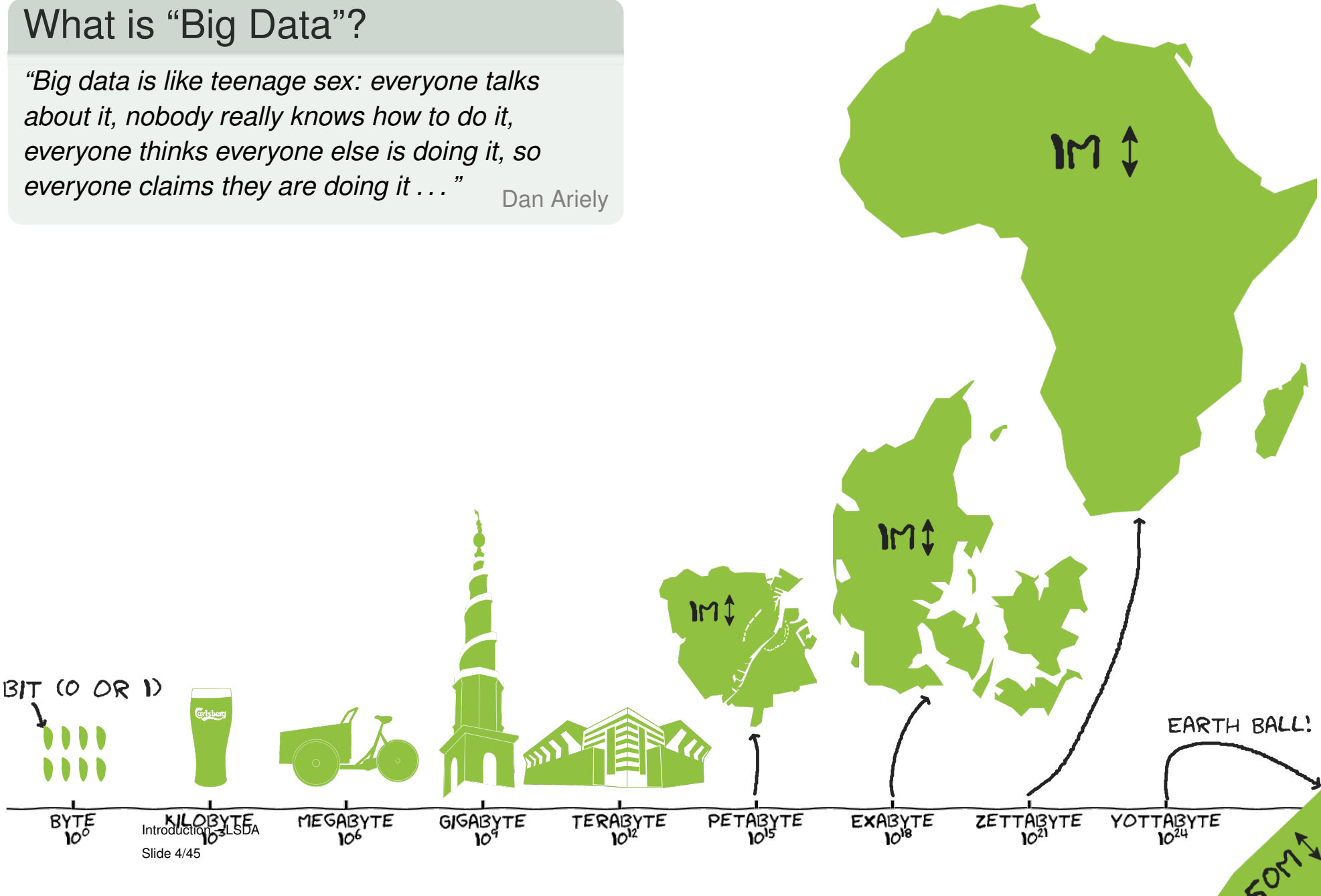
Outline

Outline

What is “Big Data”?

“Big data is like teenage sex: everyone talks about it, nobody really knows how to do it, everyone thinks everyone else is doing it, so everyone claims they are doing it . . . ”

Dan Ariely



What is “Big Data”?

“Big data is like teenage sex: everyone talks about it, nobody really knows how to do it, everyone thinks everyone else is doing i everyone claims they are doing it . . . ”

1x IMAGE

1x E-MAIL

BIT (0 OR 1)

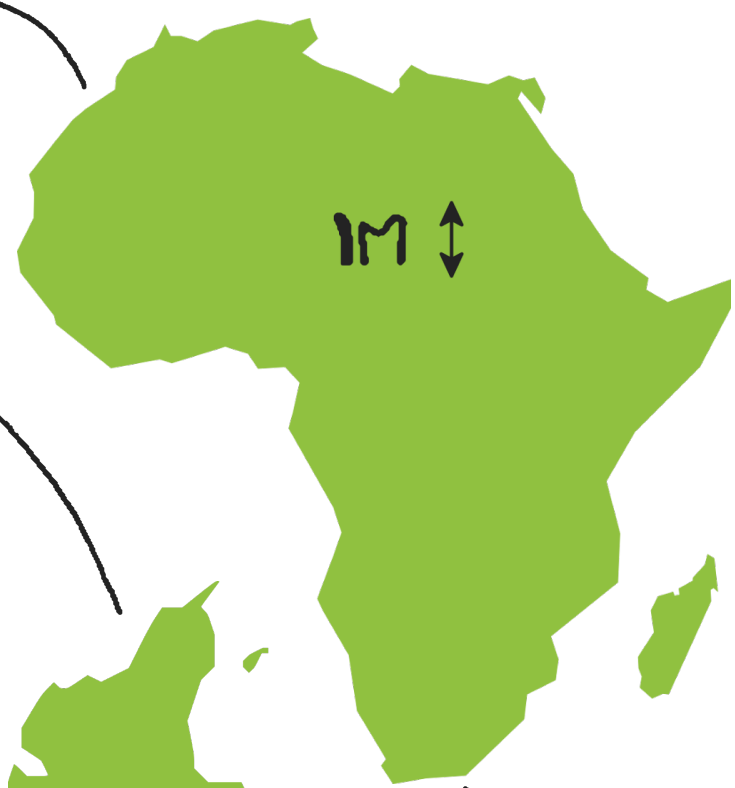
1x CD

1x LAPTOP

50,000 PHONES

TRAFFIC 2016
(WWW.CISCO.COM)

FACEBOOK:
1XPB PER DAY



BYTE
10^0

KILOBYTE
10^3

MEGABYTE
10^6

GIGABYTE
10^9

TERABYTE
10^{12}

PETABYTE
10^{15}

EXABYTE
10^{18}

ZETTABYTE
10^{21}

YOTTABYTE
10^{24}

EARTH BALL!

Today: Satellites

Today/Tomorrow: Satellites

Need Help? FAQ Contact Us About Sentinel Online

Google™ Custom Search

Missions | User Guides | Technical Guides | Thematic Areas | Data Access | Toolboxes

You are here Home

Share

Welcome to Sentinel Online

Sentinel News

- Orbiting in sunshine
- Maintenance on the Data Hubs on 7 March 2017
- Second 'colour vision' satellite for Copernicus
- Sentinel-3 SRAL Level-1A NTC/STC available in

Events

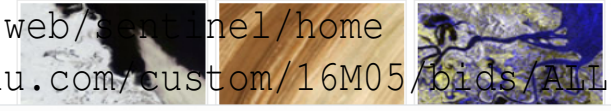
- Sentinel-3 Validation Team Meeting
- See all Sentinel Events

Browse to Oth

- EU Copernicus
- ESA Copernicus
- Observing the E
- Earth Onlin
- CSCDA
- Coper

Sentinel Missions

Thematic Areas

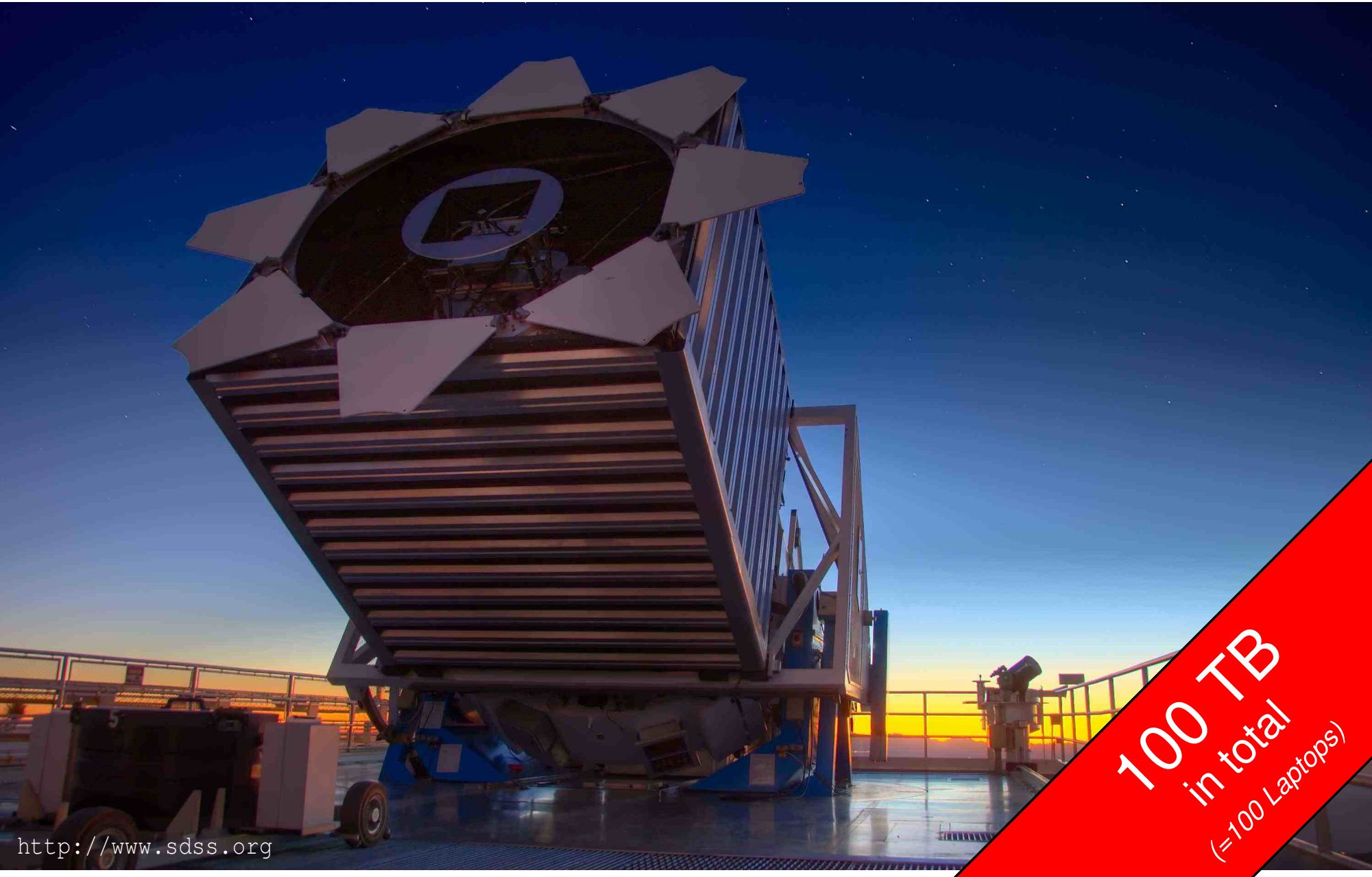


https://sentinel.esa.int/web/sentinel/home

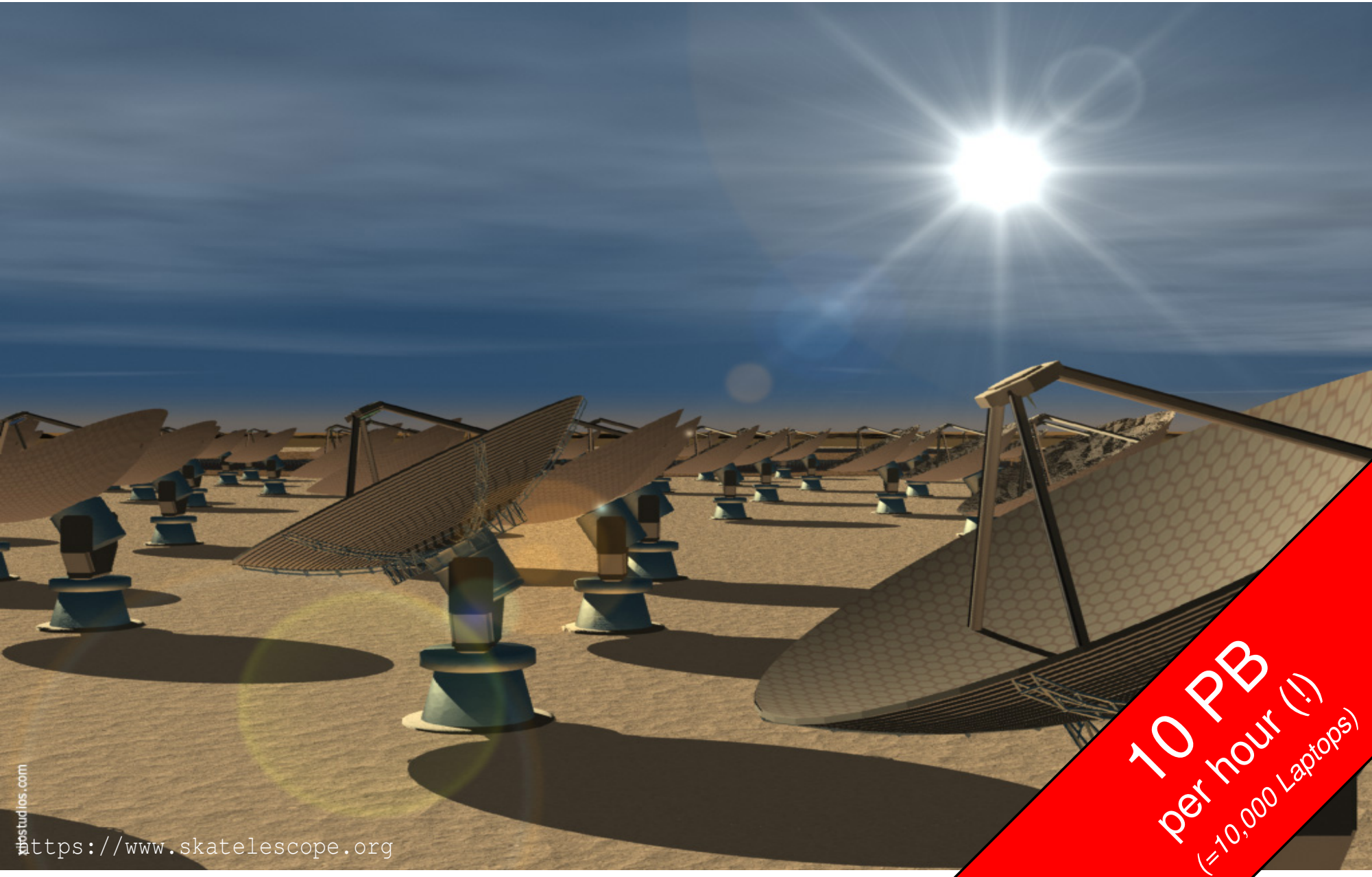
http://esaconferencebureau.com/custom/16M05/bids/ALL/01_094

> 100 TB per day (Data Products)

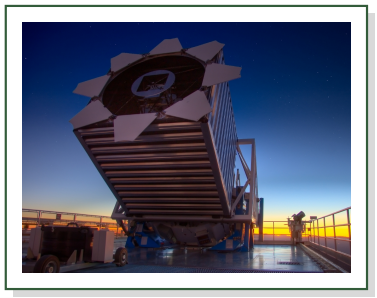
Today: Telescopes



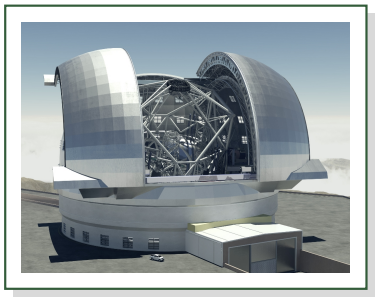
Tomorrow: Telescopes



Data Analysis: Example I

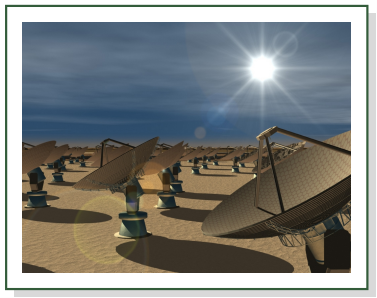


2015 → SDSS
(in total: 100TB)



2024 → EELT
(per night: 2TB)

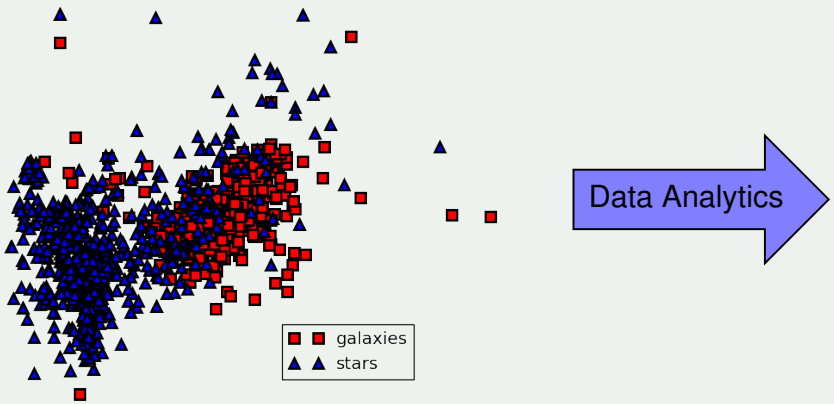
2021 → LSST
(per night: 30TB)



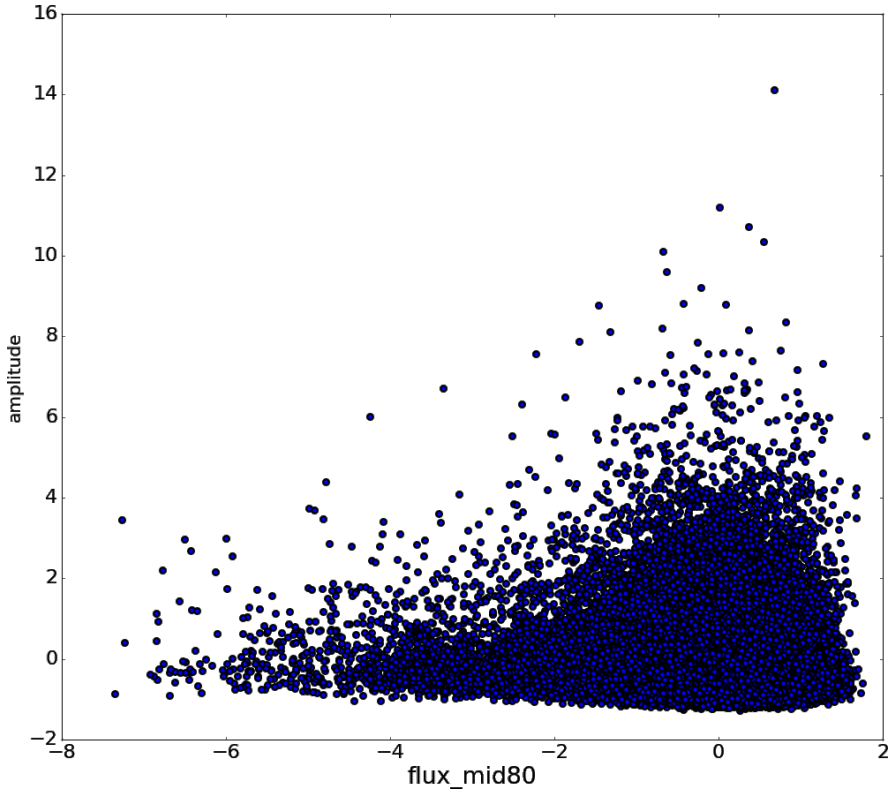
2025 → SKA
(per hour: 10PB)

Challenges

Find interesting objects such as distant galaxies or very rare stars! Combine various data sources! Handle billions of objects per night → Process and analyze all the data efficiently and at low cost!



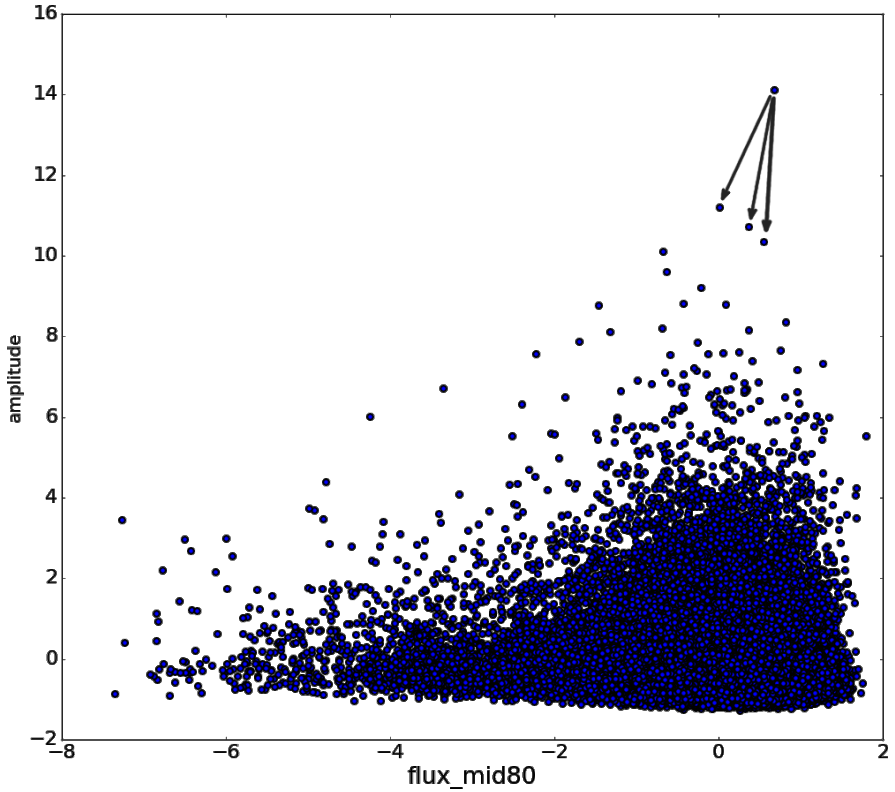
Data Analysis: Example II



Outlier Detection

"I have one billion objects and each of them is described via 10 values (features). Can you find the outliers for me, i.e., objects that are somehow different from the other ones?"

Data Analysis: Example II

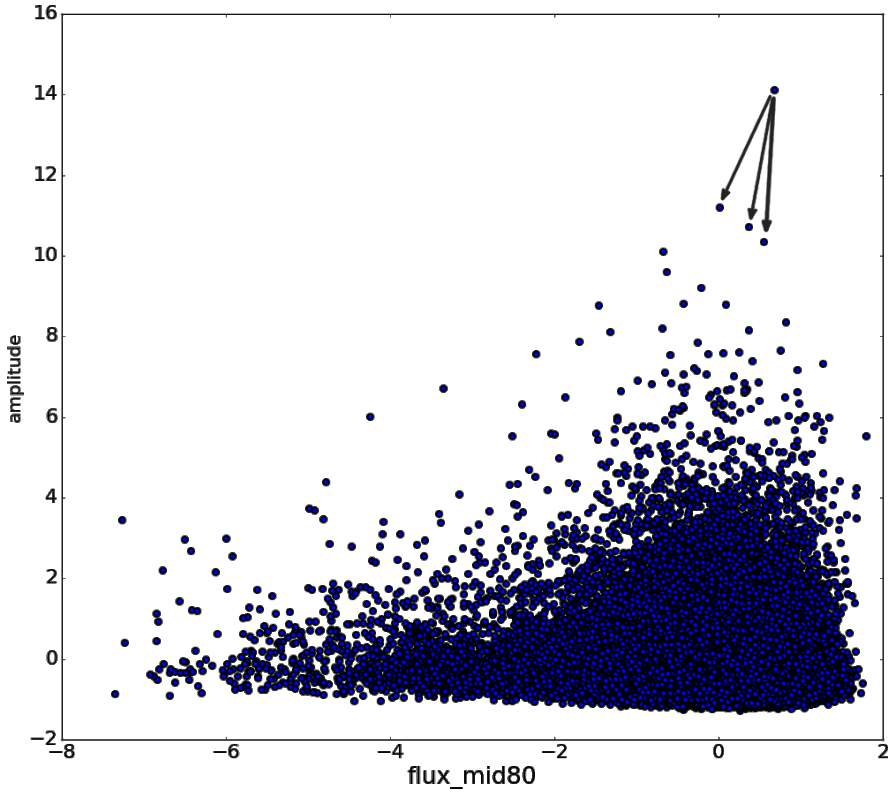


Outlier Detection

"I have one billion objects and each of them is described via 10 values (features). Can you find the outliers for me, i.e., objects that are somehow different from the other ones?"

→ Compute, for each point, the distance to its nearest neighbors!

Data Analysis: Example II

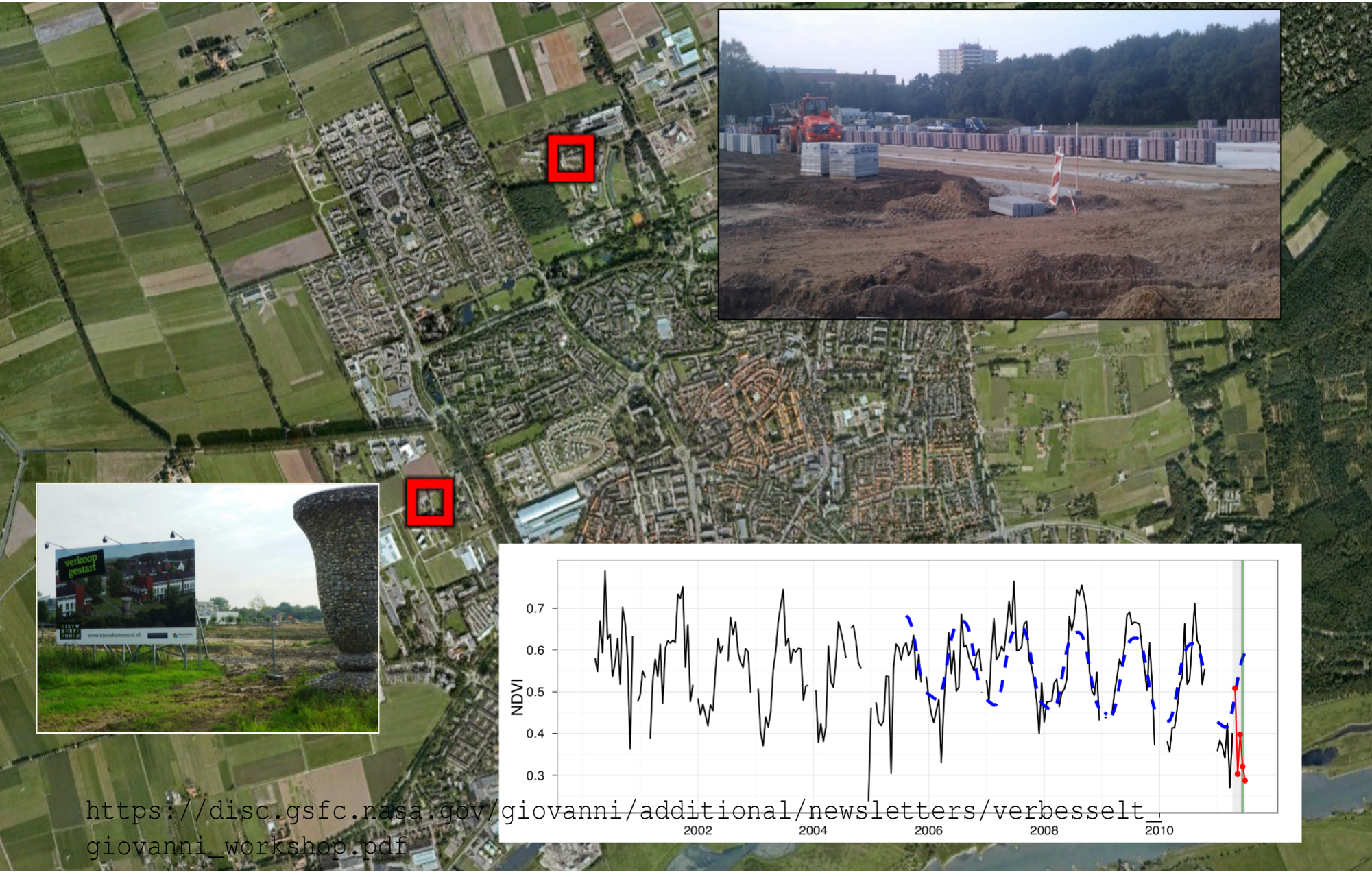


Computing All Nearest Neighbors

```
1  for p in all_points:
2      for q in all_points:
3          d = distance(p, q)
4          ...
```

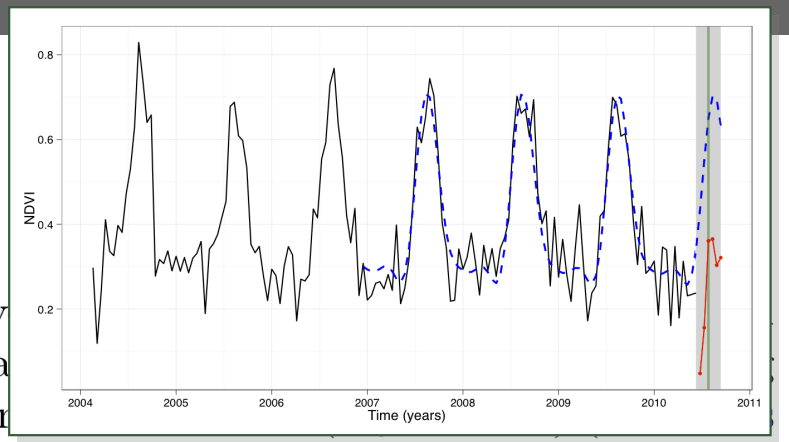
Takes Forever!

Data Analysis: Example III



https://disc.gsfc.nasa.gov/giovanni/additional/newsletters/verbesselt_giovanni_workshop.pdf

Data Analysis: Example III



2.2. Season-trend model

The method proposed here is based on a similar additiv
by Verbesselt *et al.* (2010b) to account for seasonal a
within climate-driven biophysical indicators derived fro
and Henebry 2005). For the observations y_t at time t, a season-trend model is assumed with linear trend and harmonic season:

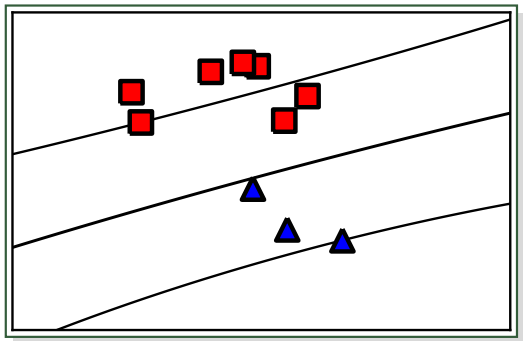
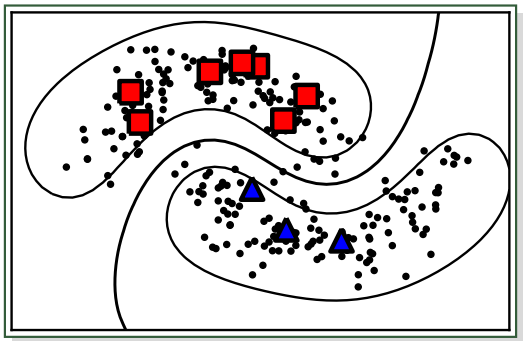
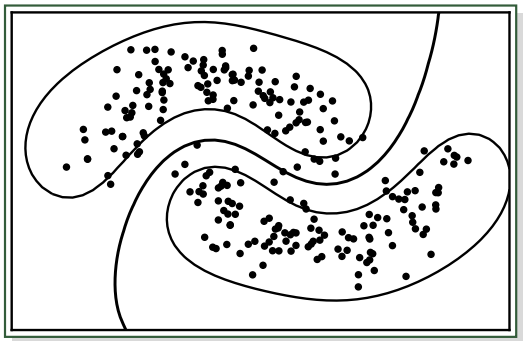
$$y_t = \alpha_1 + \alpha_2 t + \sum_{j=1}^{k} \gamma_j \sin\left(\frac{2\pi j t}{f} + \delta_j\right) + \varepsilon_t, \quad (1)$$

where the intercept α_1, slope α_2 (i.e., trend), amplitudes $\gamma_1, \ldots, \gamma_k$, and phases $\delta_1, \ldots, \delta_k$ (i.e., season) are the unknown parameters, f is the known frequency (e.g., $f = 23$ annual observations for a 16-day time series), and ε_t is the unobservable error term at time t (with standard deviation σ). In the applications below, we employ three harmonic terms (i.e., $k = 3$) to robustly detect disturbances within MODIS NDVI time series, as components four and higher represent variations that occur on a three-month cycle or less (Geerken 2009; Julien and Sobrino 2010). The model (Equation 1) can be written as a standard linear regression model (see e.g. Cryer and Chan 2008, Chapter 3.3):

$$\begin{aligned} y_t &= x_t^\top \beta + \varepsilon_t, \\ x_t &= \{1, t, \sin(2\pi 1t/f), \cos(2\pi 1t/f), \ldots, \sin(2\pi kt/f), \cos(2\pi kt/f)\}^\top, \\ \beta &= \{\alpha_1, \alpha_2, \gamma_1 \cos(\delta_1), \gamma_1 \sin(\delta_1), \ldots, \gamma_k \cos(\delta_k), \gamma_k \sin(\delta_k)\}^\top, \end{aligned}$$

Verbesselt, Zeileis, Herold: *Near real-time disturbance detection using satellite image time series*. Remote Sensing of Environment, 2013.

Data Analysis: Example IV

Combinatorial Optimization Task

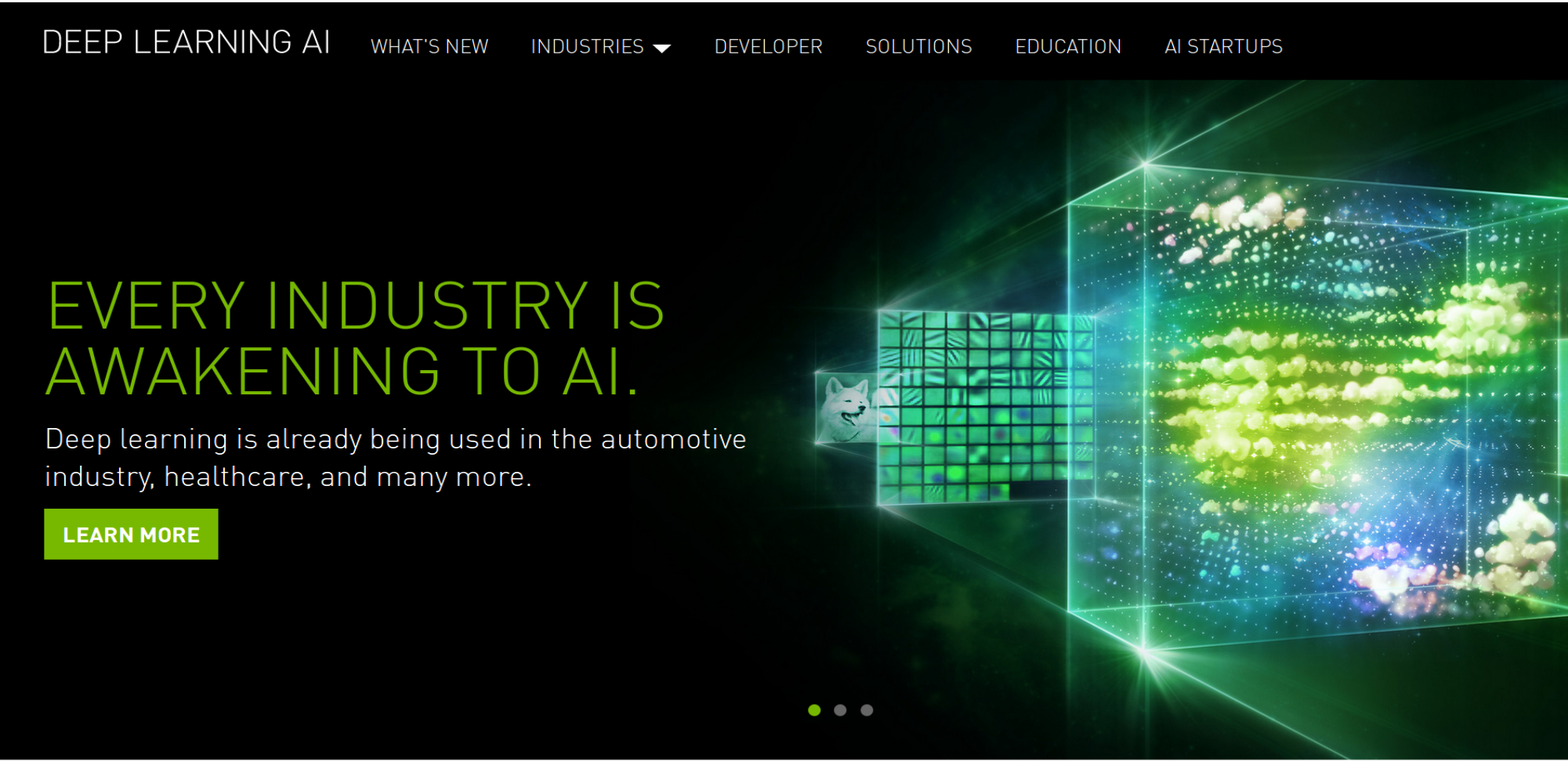
$$\begin{aligned} &\underset{\substack{\mathbf{y}\in\{-1,+1\}^u,\\ \mathbf{w}\in\mathcal{H},\, b\in\mathbb{R},\, \xi'\in\mathbb{R}^l,\, \xi\in\mathbb{R}^u}}{\text{minimize}} && \frac{1}{2}\|\mathbf{w}\|^2 + C'\sum_{i=1}^{l}\xi'_i + C\sum_{i=1}^{u}\xi_i \\ &\text{s.t.} && y'_i\left(\langle \mathbf{w}, \Phi(\mathbf{x}_i)\rangle + b\right) \geq 1-\xi'_i,\ \xi'_i \geq 0, \\ &\text{and} && y_i\left(\langle \mathbf{w}, \Phi(\mathbf{x}_{l+i})\rangle\right) + b) \geq 1-\xi_i,\ \xi_i \geq 0 \end{aligned}$$

Data Analysis: Example V

https://www.nvidia.com/en-us/deep-learning-ai/

NVIDIA

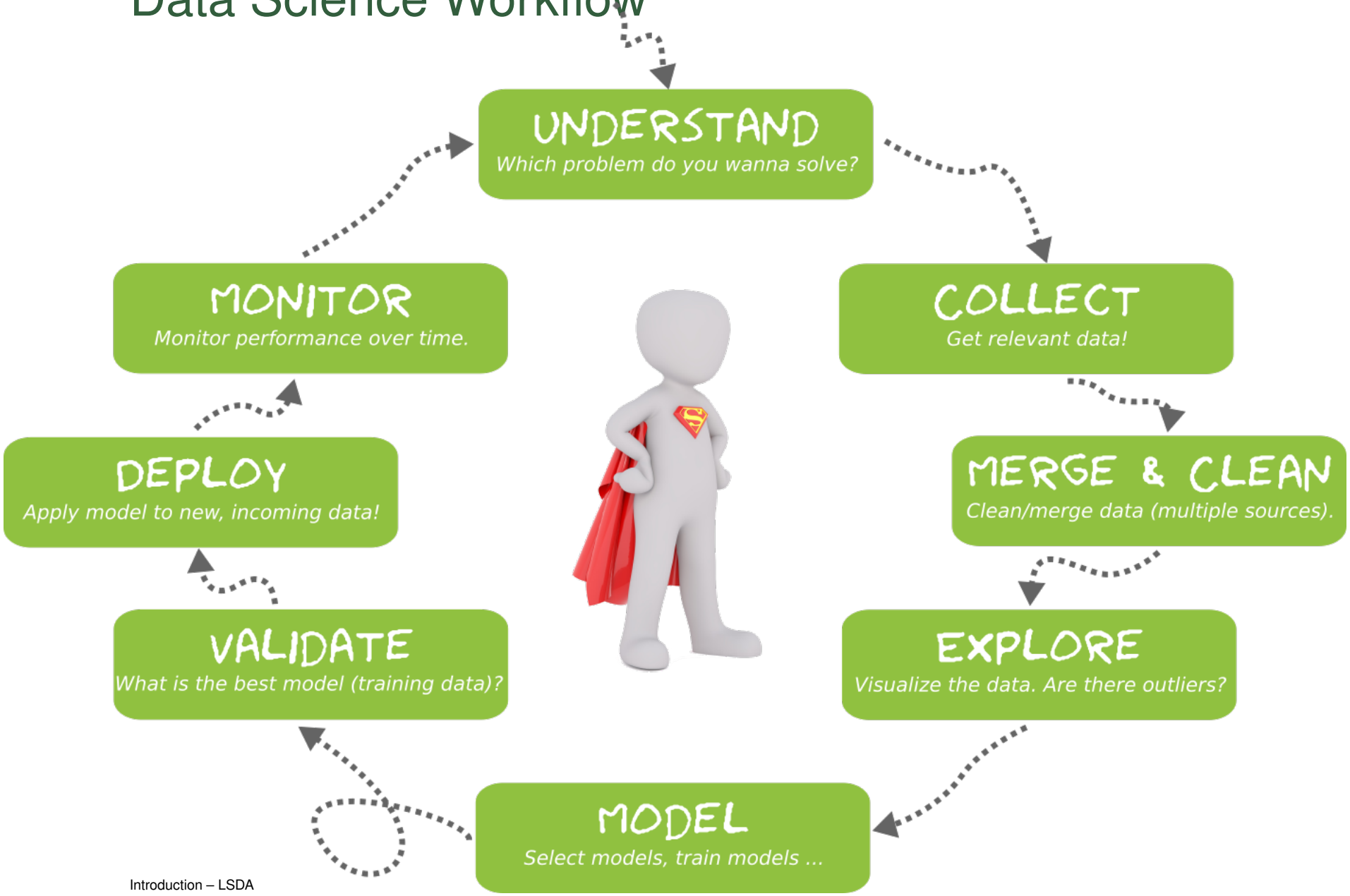
DEEP LEARNING AI WHAT'S NEW INDUSTRIES DEVELOPER SOLUTIONS EDUCATION AI STARTUPS



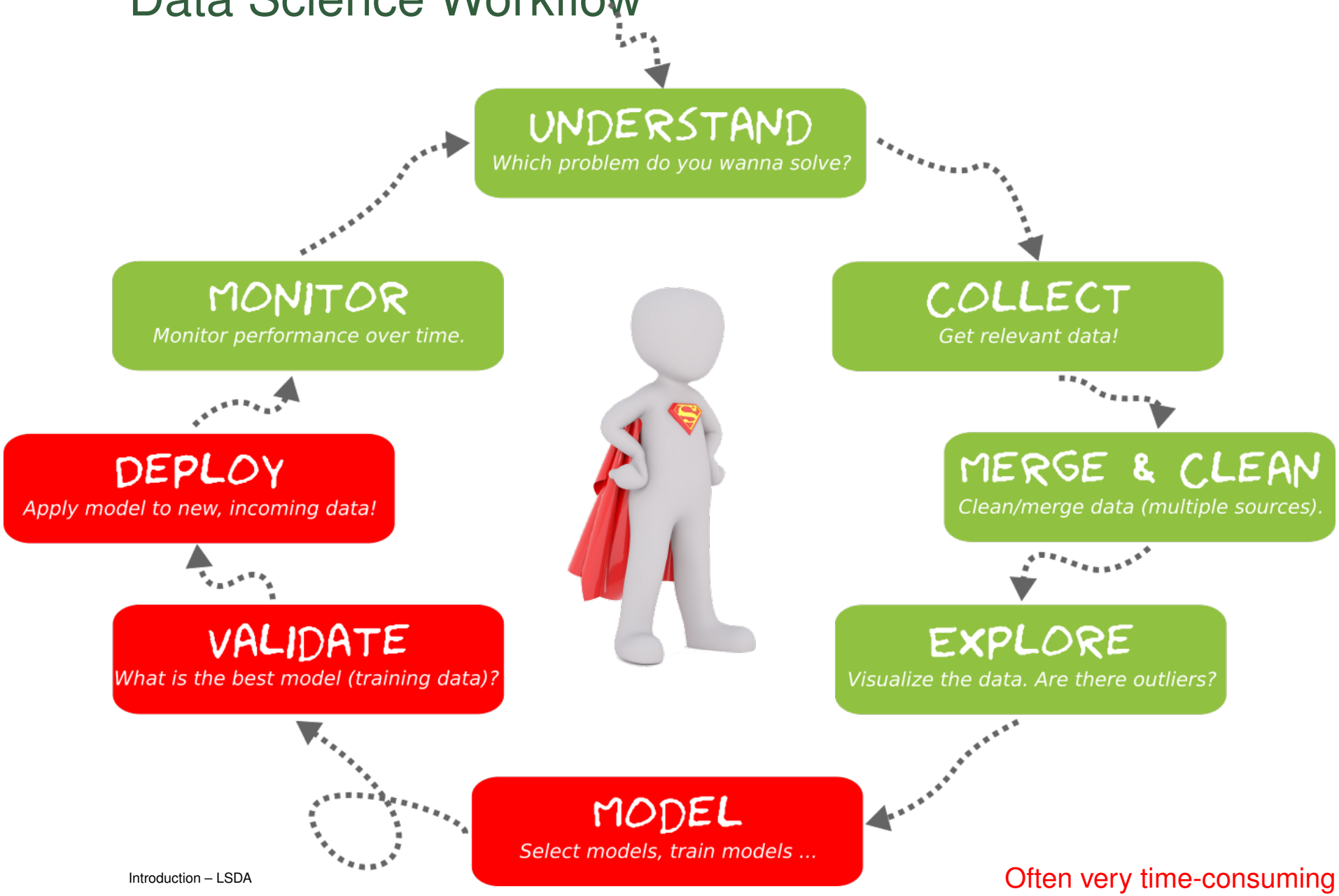
SEE YOUR LIFE'S WORK REALIZED, WITH AI.

Preventing disease. Building smart cities. Revolutionizing analytics. These are just a few things

Data Science Workflow

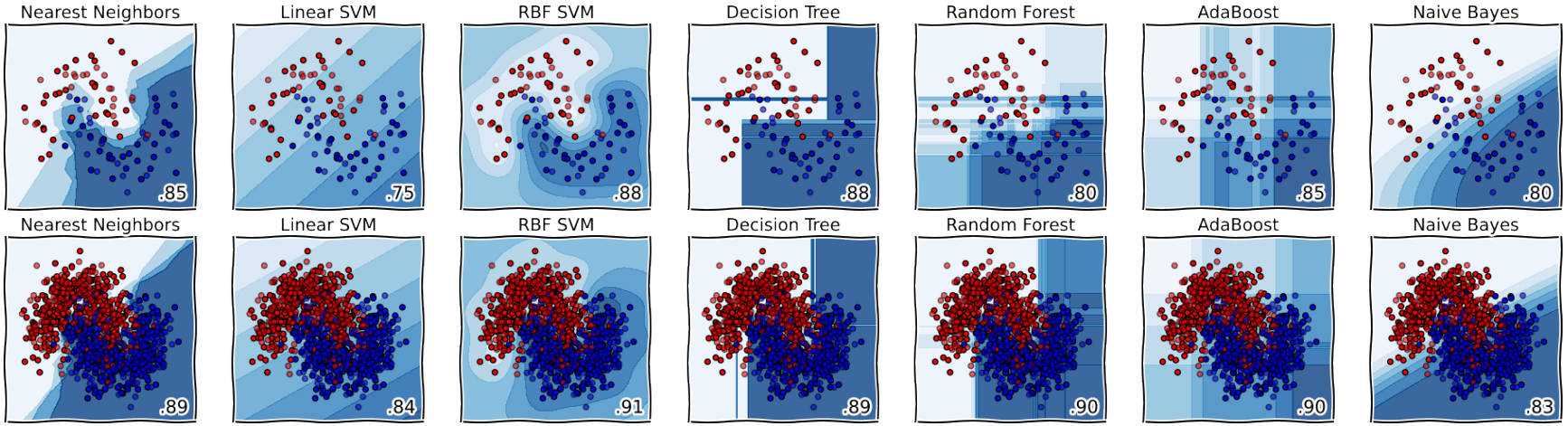


Data Science Workflow

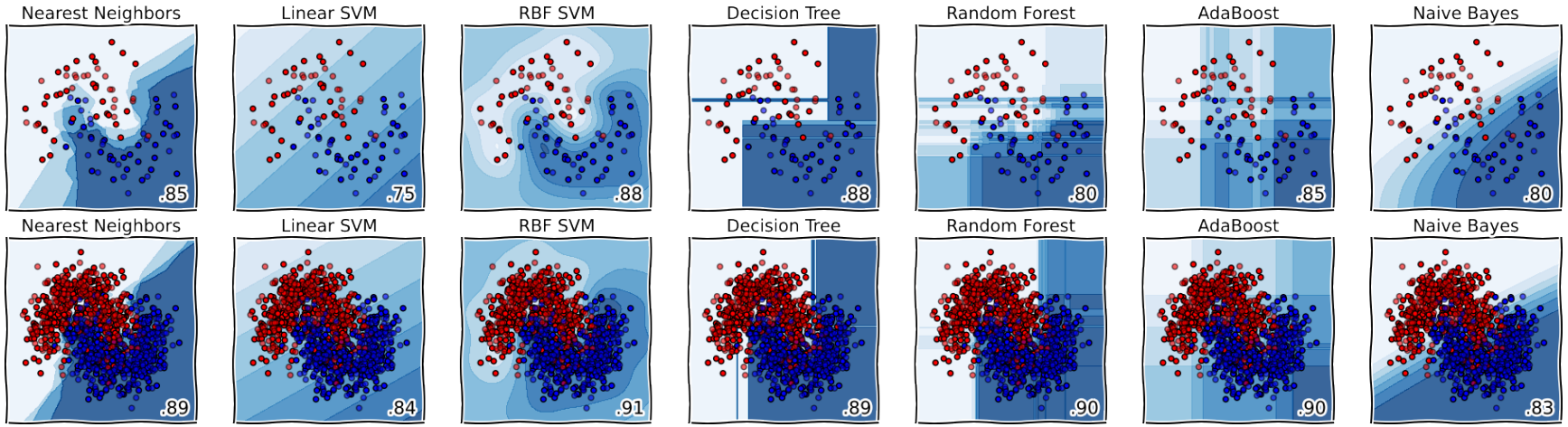


Often very time-consuming!

Why Big Data?


Nearest Neighbors
Linear SVM
RBF SVM
Decision Tree
Random Forest
AdaBoost
Naive Bayes
.85
.75
.88
.88
.80
.85
.80
Nearest Neighbors
Linear SVM
RBF SVM
Decision Tree
Random Forest
AdaBoost
Naive Bayes
.89
.84
.91
.89
.90
.90
.83

Why Big Data?



Large-Scale Data Analysis

1. Interface: Machine Learning + Data Structures + Optimization + HPC + . . .
   (often depends on the particular application domain!)
2. Key Question: How can we analyze all the data efficiently and at low cost?

Outline

About Us

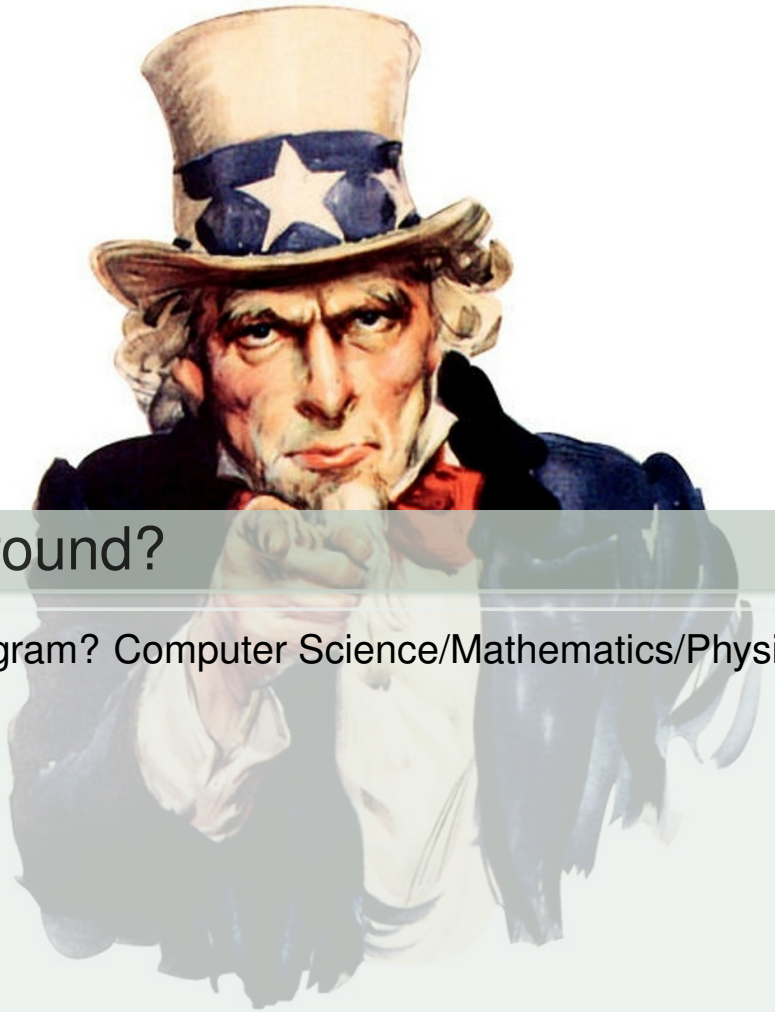
Fabian Gieseke
fabian.gieseke@di.ku.dk

Christian Igel
igel@di.ku.dk

Teaching Assistants

- Frederik Faye (pft556@alumni.ku.dk)
- Mihai Popovici (mtf422@alumni.ku.dk)
- Haining Tong (znr113@alumni.ku.dk)

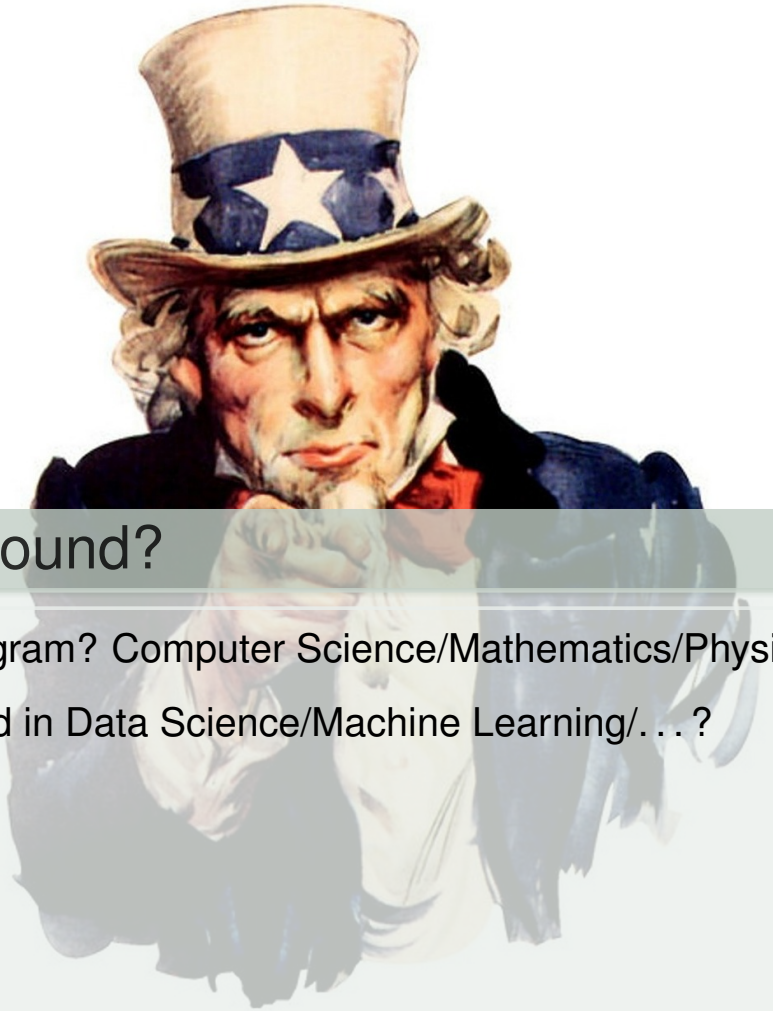
About You



What's your background?

1. What's your study program? Computer Science/Mathematics/Physics/. . . ?

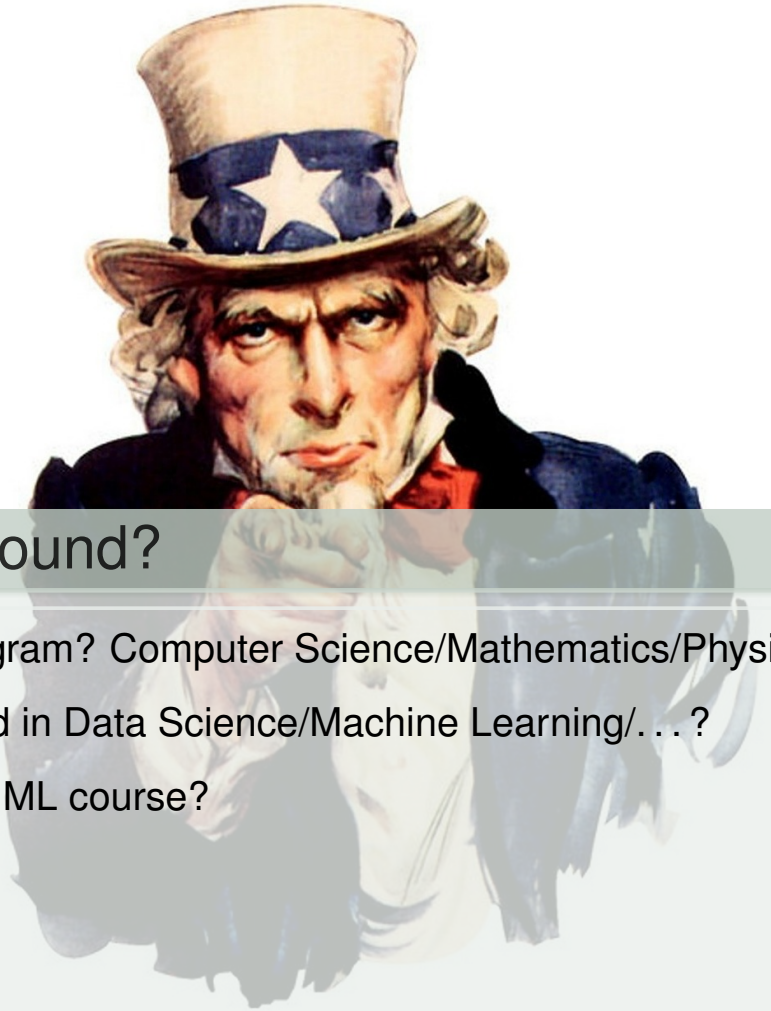
About You



What's your background?

1. What's your study program? Computer Science/Mathematics/Physics/. . . ?
2. Who has a background in Data Science/Machine Learning/. . . ?

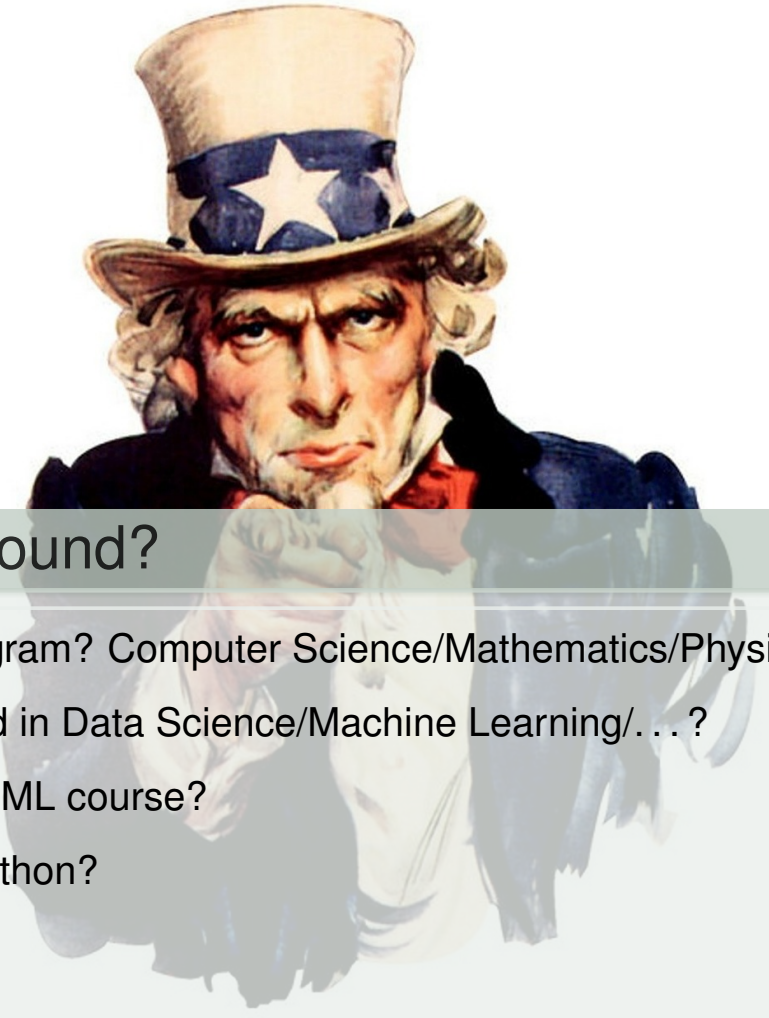
About You



What's your background?

1. What's your study program? Computer Science/Mathematics/Physics/. . . ?
2. Who has a background in Data Science/Machine Learning/. . . ?
3. Who has attended the ML course?

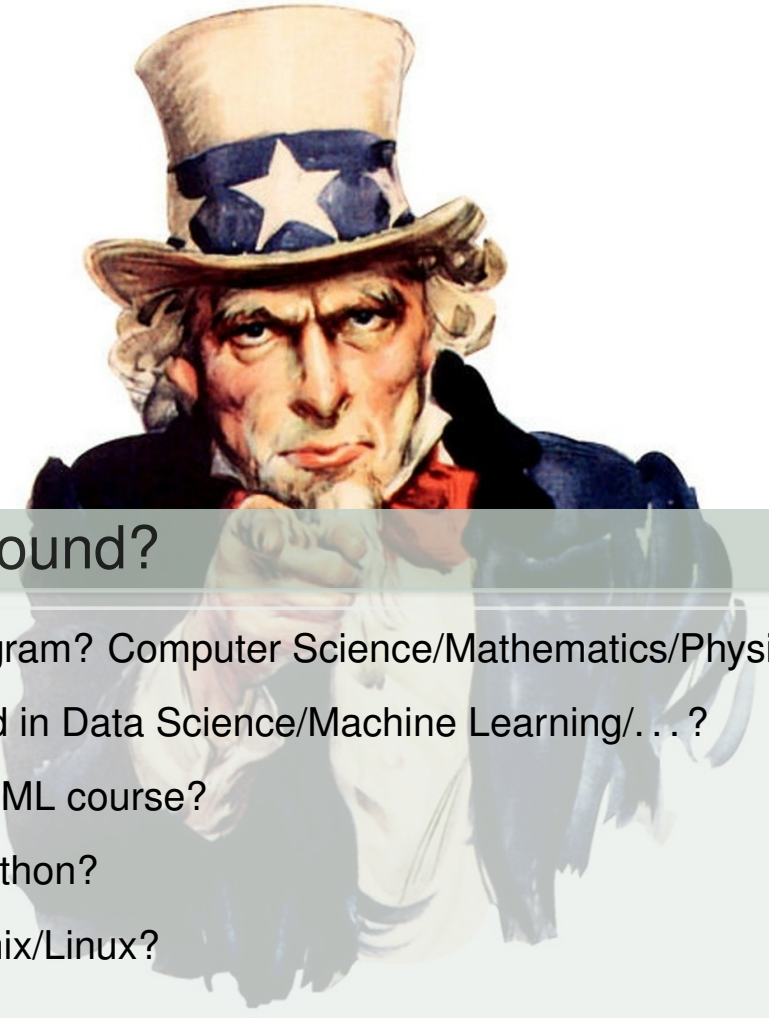
About You



What's your background?

1. What's your study program? Computer Science/Mathematics/Physics/. . . ?
2. Who has a background in Data Science/Machine Learning/. . . ?
3. Who has attended the ML course?
4. Who is familiar with Python?

About You



What's your background?

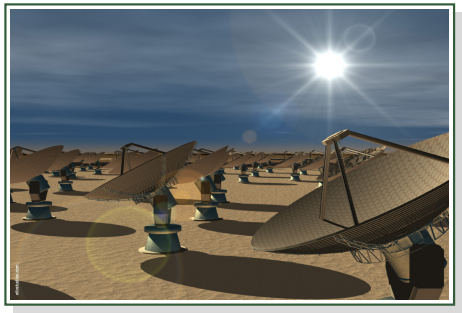
1. What's your study program? Computer Science/Mathematics/Physics/. . . ?
2. Who has a background in Data Science/Machine Learning/. . . ?
3. Who has attended the ML course?
4. Who is familiar with Python?
5. Who is familiar with Unix/Linux?

About You

What's your background?

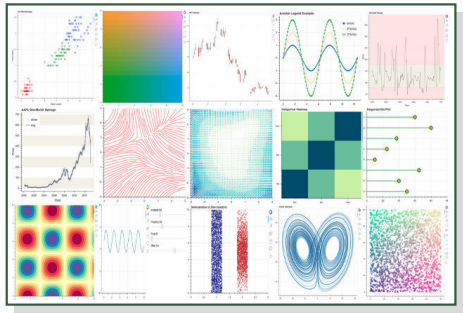
1. What's your study program? Computer Science/Mathematics/Physics/. . . ?
2. Who has a background in Data Science/Machine Learning/. . . ?
3. Who has attended the ML course?
4. Who is familiar with Python?
5. Who is familiar with Unix/Linux?
6. What did you have in mind when you signed up for the course?

This Course



(1) Big Data

- Big Data: Applications, Goals?
- Computer Architectures
- Fundamentals of Data Mining



(2) DA & Little Resources

- Streaming & Online Learning
- Large-Scale Machine Learning
- Parallel Machine Learning

(3) DA & Big Resources

- Deep Learning
- Cloud Computing
- Hadoop & Spark

What you should bring along (or be willing to learn)?

1. Algorithms & Math: Basic concepts (pseudocode, $\mathcal{O}$-notation, calculus, probability, ...)
2. Programming: Basic knowledge (Python $\rightarrow$ main language), C, ...
3. Other: Linux/Unix (ssh, terminal, compiling code, ...), git, ...
4. Machine Learning: Some background in ML (e.g., local course)

BYTE 10^0 KILOBYTE 10^3 MEGABYTE 10^6 GIGABYTE 10^9 TERABYTE 10^{12} PETABYTE 10^{15} EXABYTE 10^{18} ZETTABYTE 10^{21} YOTTABYTE 10^{24}

Tentative Schedule

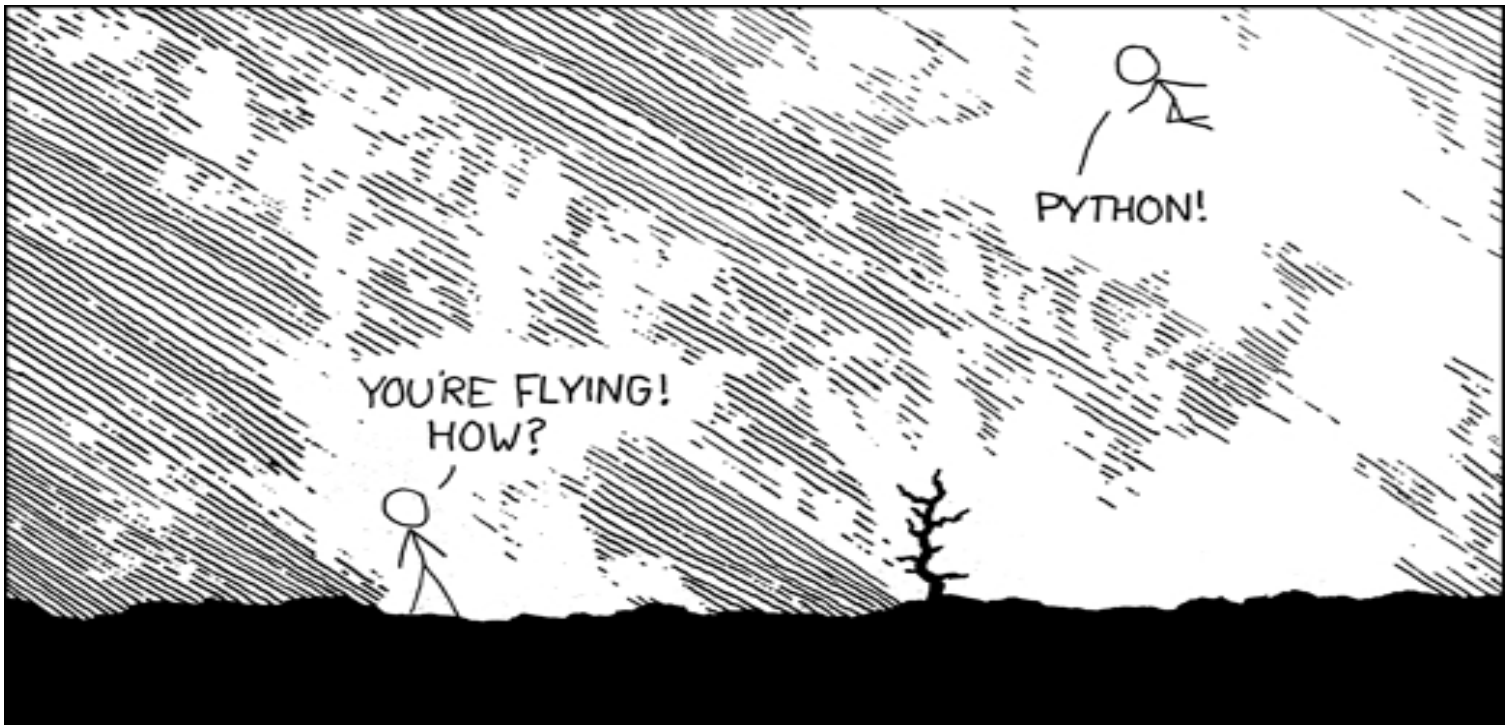
1. Introduction, Big Data, and Competitions (FG)
2. Fundamentals Computing & Large-Scale Nearest Neighbors (FG)
3. Neural Networks I & Tensorflow (CI)
4. Large-Scale Least-Squares & Large-Scale Random Forests (FG)
5. Boosted Trees & XGBoost (FG)
6. Neural Networks II (CI)
7. Neural Networks III (CI)
8. Neural Networks IV (CI)
9. Neural Networks V (CI)
10. Hadoop & Map Reduce (FG)
11. Apache Spark (FG)
12. Distributed Data Analysis with Apache Spark I (FG)
13. Distributed Data Analysis with Apache Spark II (FG)
14. Flash Talks (you) + Wrap-Up (FG)

Tentative Schedule

1. Introduction, Big Data, and Competitions (FG)
2. Fundamentals Computing & Large-Scale Nearest Neighbors (FG)
3. Neural Networks I & Tensorflow (CI)
4. Large-Scale Least-Squares & Large-Scale Random Forests (FG)
5. Boosted Trees & XGBoost (FG)
6. Neural Networks II (CI)
7. Neural Networks III (CI)
8. Neural Networks IV (CI)
9. Neural Networks V (CI)
10. Hadoop & Map Reduce (FG)
11. Apache Spark (FG)
12. Distributed Data Analysis with Apache Spark I (FG)
13. Distributed Data Analysis with Apache Spark II (FG)
14. **Flash Talks (you)** + Wrap-Up (FG)

This course will teach you how to . . .

1. . . . speed up some well-known data science techniques.
2. . . . make use of modern deep learning models.
3. . . . conduct large-scale distributed analysis.
4. . . . approach real-world data science problems.
5. . . . do most of this using Python!



https://xkcd.com/353/

This course will . . .

1. . . . not be an introduction to programming and Python.
2. . . . not be an introduction to Linux.
3. . . . not be an introduction to machine learning/data science.
4. . . . not cover too many theoretical aspects of machine learning.

This course overlaps with . . .

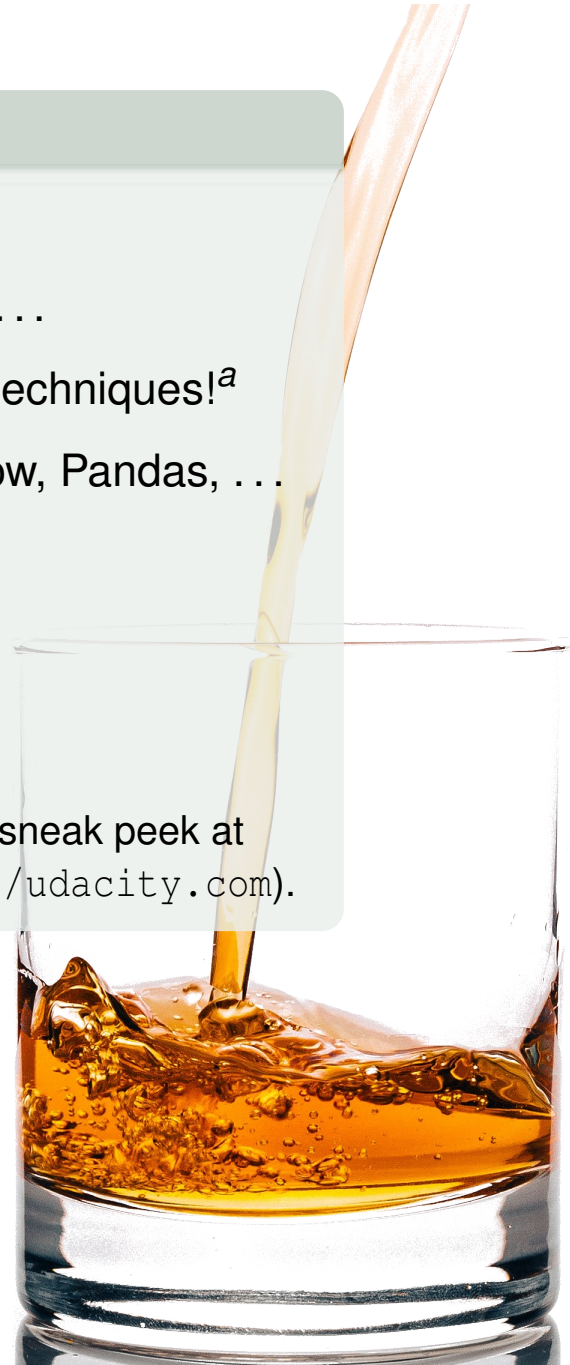
1. Machine Learning
2. Introduction to Data Science
3. Algorithms and Data Structures
4. High Performance Computing
5. Computer Architectures
6. Information Retrieval
7. Databases
8. Linux and Python Programming
9. . . .

Tools!

We will make use of . . .

1. Algorithms: Pseudocode, $\mathcal{O}$-notation, . . .
2. Mathematics: Basic calculus, basic probability theory, . . .
3. Machine Learning/Data Science: Basic concepts and techniques![a]
4. Programming (Python): Numpy, Scikit-Learn, Tensorflow, Pandas, . . .
5. VirtualBox & Linux: VirtualBox, SSH, Terminal, . . .
6. Hadoop & Spark: HDFS, PySpark, Jupyter, . . .
7. . . .

[a] Maybe: Have a look at corresponding DIKU courses (slides) or sneak peek at online courses (e.g, “Introduction to Machine Learning” on `https://udacity.com`).



Note: Getting familiar with these tools can be very time-consuming. The homework assignments also aim at overcoming potential technical difficulties and at learning how to use new tools/techniques!

Homework and Assessment (Tentative!)

Continuous Assessment

"4-6 weekly take-home exercises. The final grade will be the average over all assignments except the one with the lowest score."

1. 3 individual take-home assignments (2 weeks for handing in)
   - HW2: Nearest Neighbors, Tree Ensembles, and Least-Squares
   - HW3: Neural Networks
   - HW4: Hadoop and Spark
2. 1 competition assignment (HW1, in parallel to HW2–HW4)
   - Team work: 3-4 students per team
   - Individual assessment!

Assessment

Correction & Grading

Feedback and tentative grades via Absalon on a regular basis. The **final grade** will be the average over

1. your **3 individual assignments** and
2. your **contribution to the competition assignment** (you will have to point out your individual contribution in the report),

except the one with the lowest score.

Assessment

Correction & Grading

Feedback and tentative grades via Absalon on a regular basis. The **final grade** will be the average over

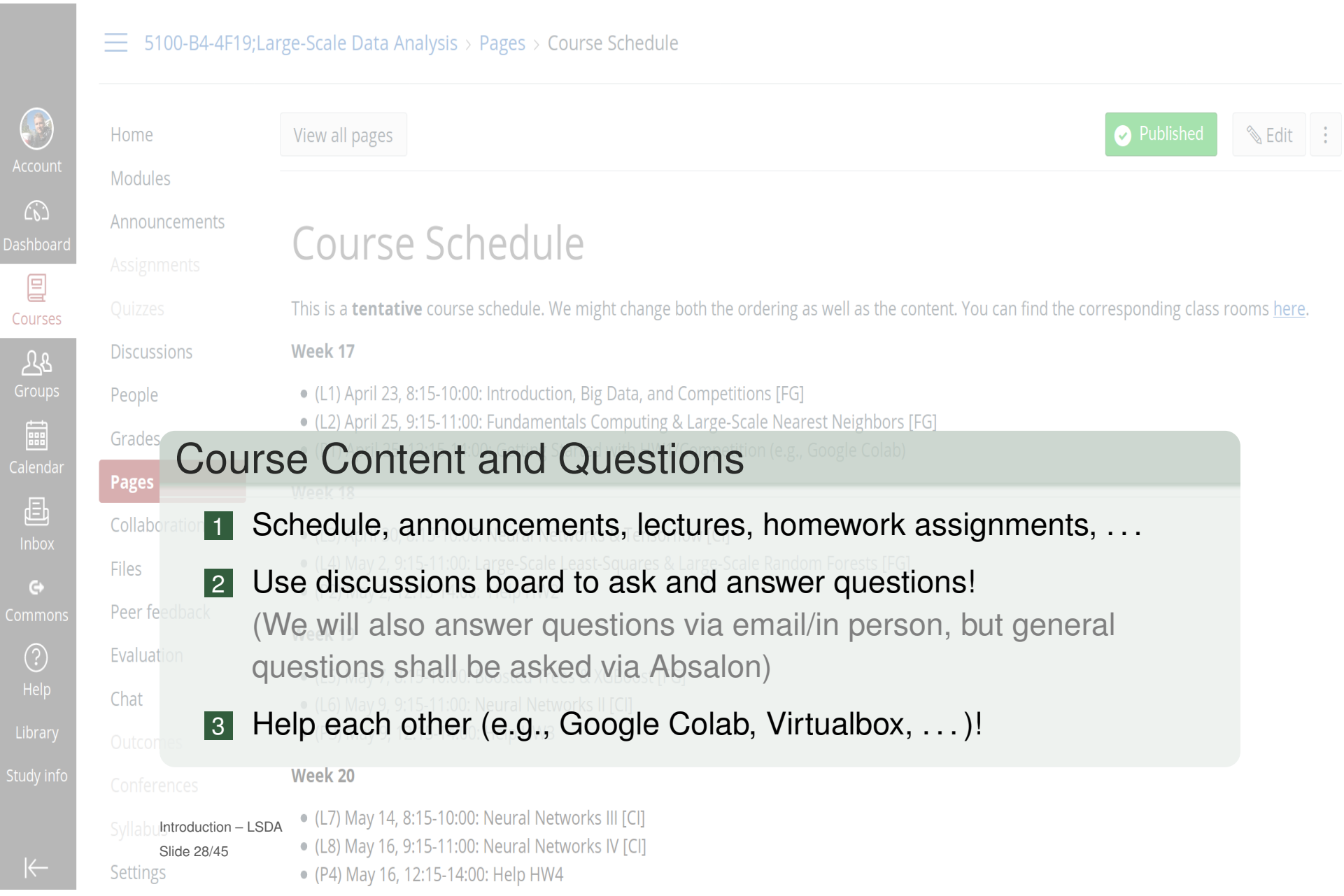
1. your **3 individual assignments** and
2. your **contribution to the competition assignment** (you will have to point out your individual contribution in the report),

except the one with the lowest score.

If you have some concerns regarding the feedback/grading:

1. Assignments: Please let us/the TAs know in case you think that there is a **serious flaw** in the correction of an assignment (e.g., missing points, . . .).
2. Note: We will **double check the scores** at the end when determining the final grades (especially the borderline cases!).

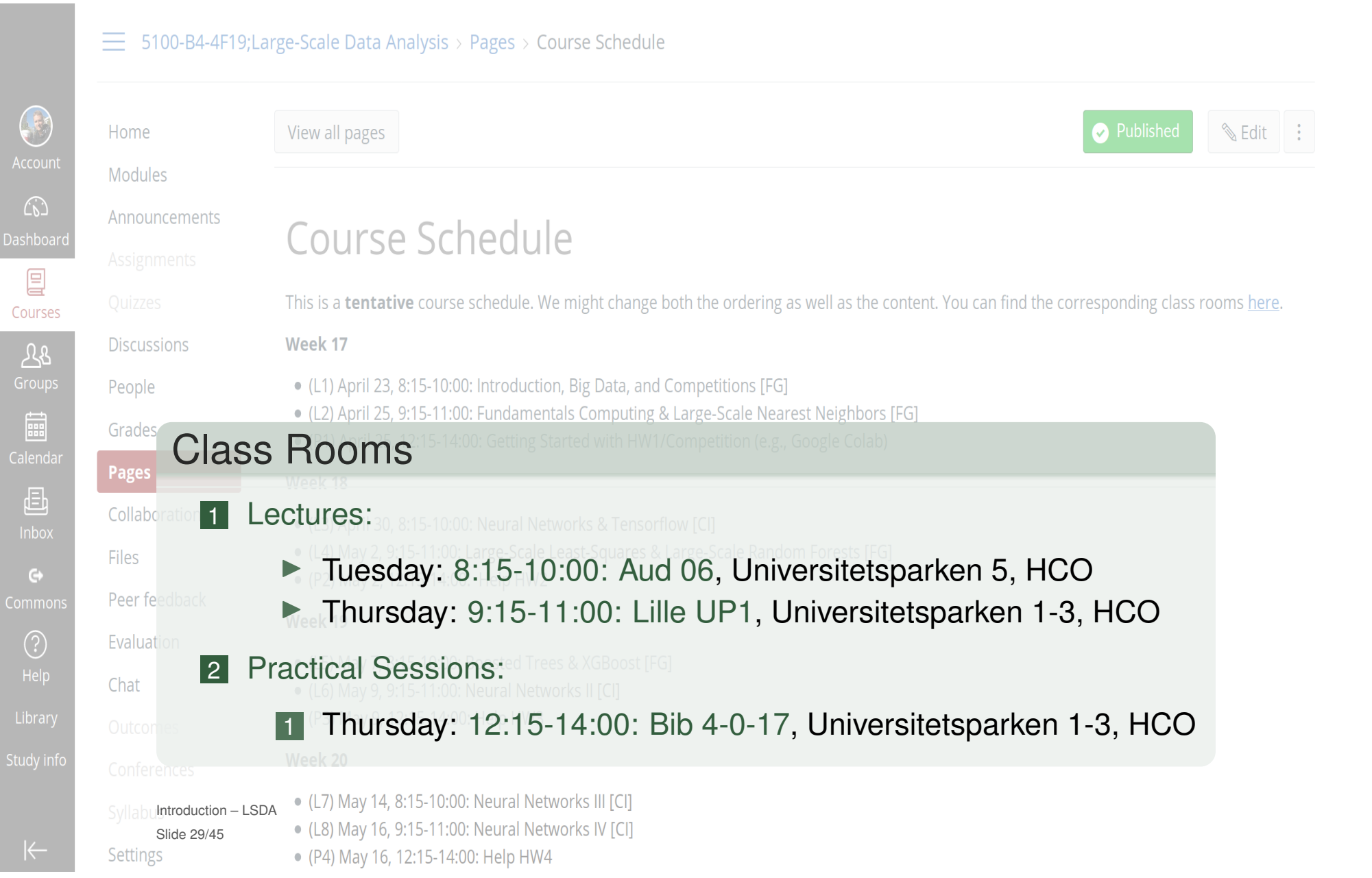
Absalon



Course Content and Questions

1. Schedule, announcements, lectures, homework assignments, . . .
2. Use discussions board to ask and answer questions!
   (We will also answer questions via email/in person, but general questions shall be asked via Absalon)
3. Help each other (e.g., Google Colab, Virtualbox, . . .)!

Where and When?



Class Rooms

1. Lectures:
   - Tuesday: 8:15-10:00: Aud 06, Universitetsparken 5, HCO
   - Thursday: 9:15-11:00: Lille UP1, Universitetsparken 1-3, HCO
2. Practical Sessions:
   1. Thursday: 12:15-14:00: Bib 4-0-17, Universitetsparken 1-3, HCO

Where and When?

Overview (Tentative!)

1. Week 1 (17)
   - April 23, 10:00: HW1 (deadline: June 11, 23:59)
   - April 23: L1 – Introduction, Big Data, and Competitions (FG)
   - April 25: L2 – Fundamentals Computing & Large-Scale Nearest Neighbors (FG)
   - April 25: P1 – Tutorial VirtualBox & Google Colab (Competition)
2. Week 2 (18)
   - April 30, 10:00: HW2 (deadline: May 14, 23:59)
   - April 30: L3 – Neural Networks & Tensorflow (CI)
   - May 2: L4 – Large-Scale Least-Squares & Large-Scale Random Forests (FG)
   - May 3: P2 – Hints HW & Help VirtualBox/Competition
3. Week 3 (19)
   - May 8: L5 – Boosted Trees & XGBoost (FG)
   - May 9: L6 – Neural Networks II (CI)
   - May 9: P3 – Hints HW/Competition
4. Week 4 (20)
   - May 14, 10:00: HW3 (deadline: May 28, 23:59)
   - May 14: L7 – Neural Networks III (CI)
   - May 16: L8 – Neural Networks IV (CI)
   - May 16: P4 – Hints HW/Competition

Where and When?

Overview (Tentative!)

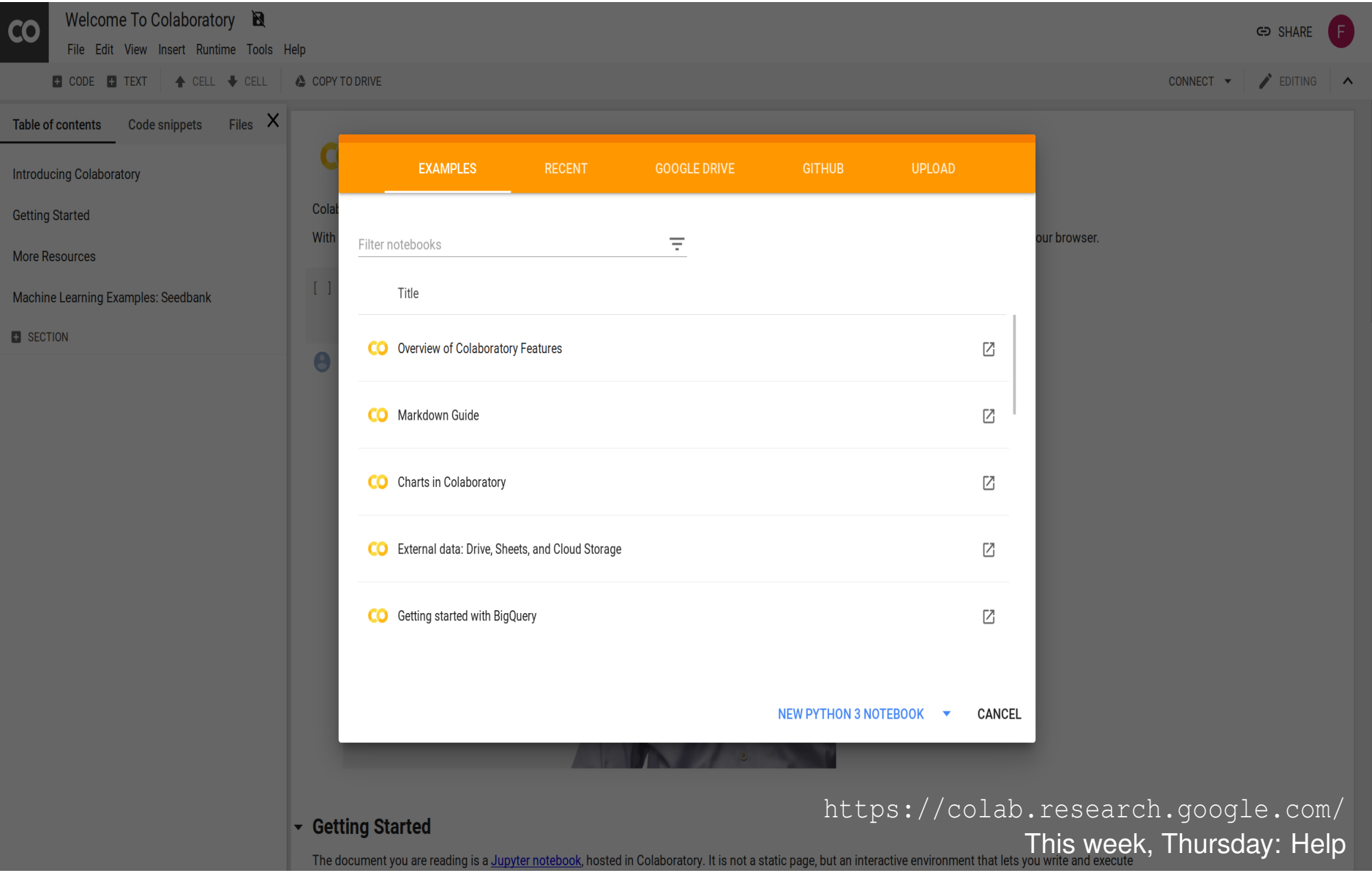
5 Week 5 (21)
May 21: L9 – Neural Networks V (CI)
May 23: L10 – Hadoop, Map Reduce, and Spark I (FG)
May 23: P5 – Hints HW/Competition

6 Week 6 (22)
May 28, 10:00: HW3 (deadline June 11, 23:59)
May 29: L11 – Hadoop, Map Reduce, and Spark II (FG)
May 30: Ascension Day

7 Week 7 (23)
June 4: L12 – Distributed Data Analysis with Apache Spark I (FG)
June 6: L13 – Distributed Data Analysis with Apache Spark II (FG)
June 6: P6 – Hints HW/Competition

8 Week 8 (24)
June 11: (no lecture)
June 13: L14 – Flash Talks & Wrap Up (you, FG)

HW1 (Competition) → Suggestion



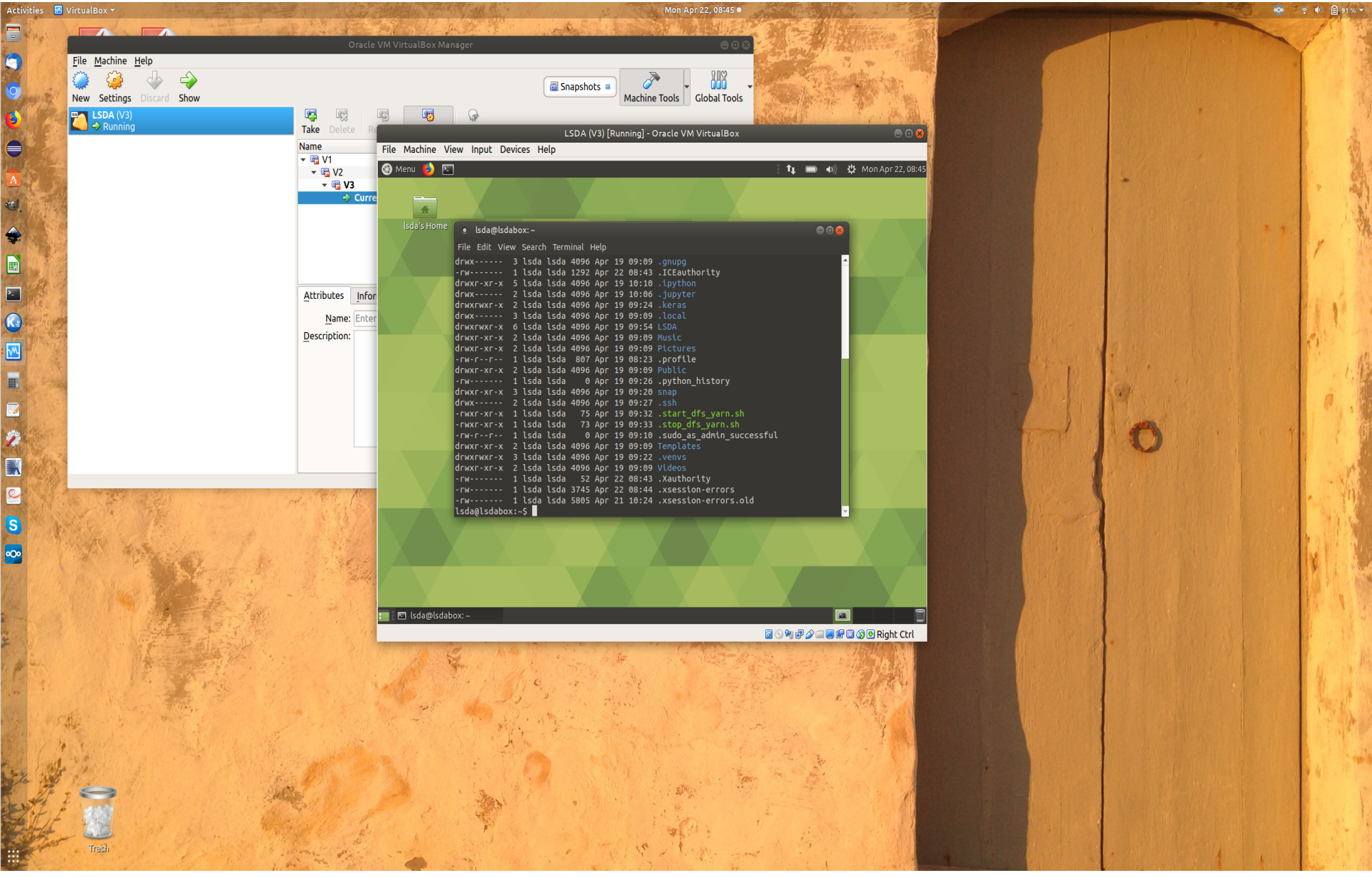
https://colab.research.google.com/

This week, Thursday: Help

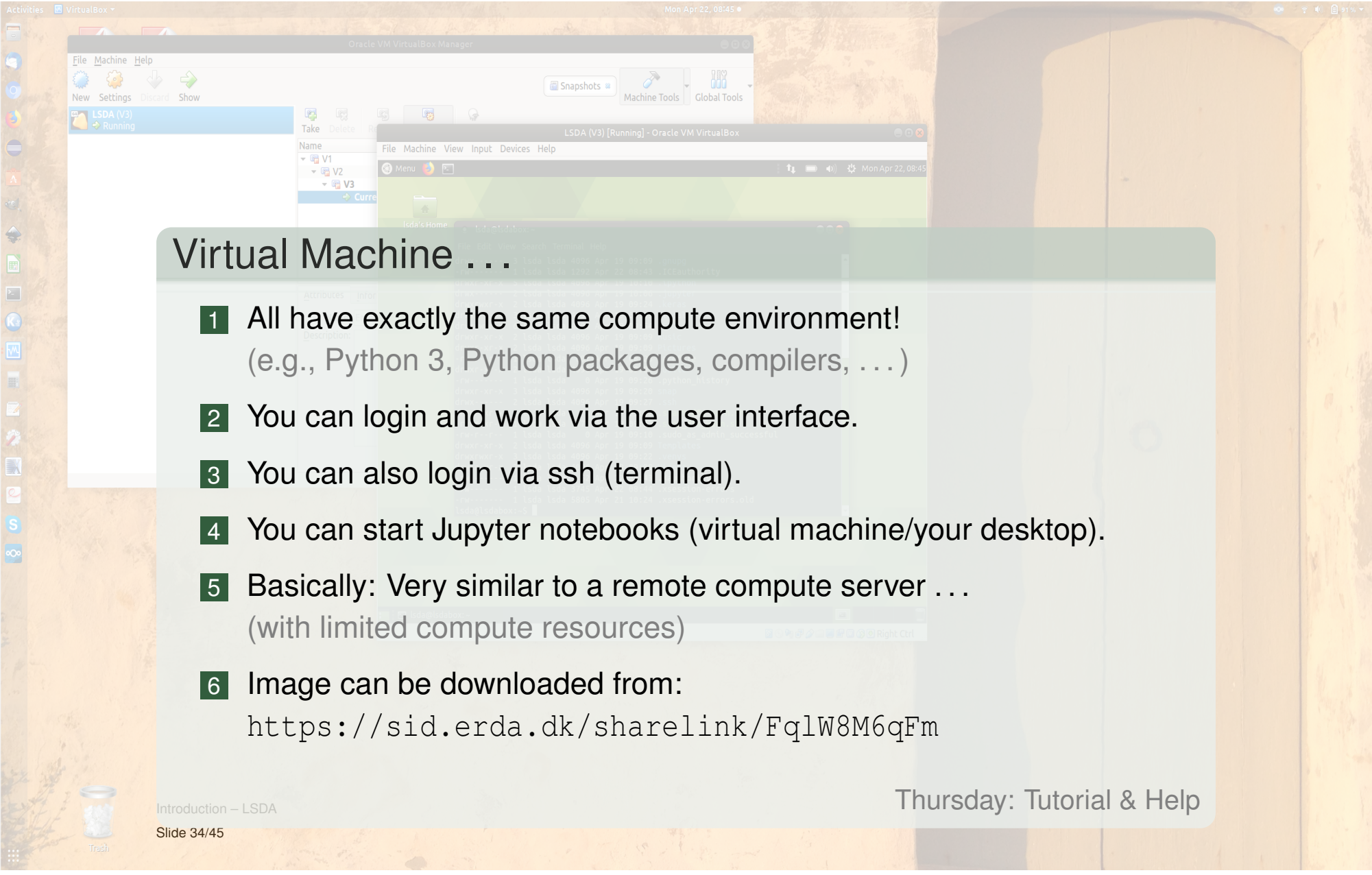
Compute Environment → HW2–HW4

https://www.virtualbox.org/

Compute Environment → HW2–HW4



Compute Environment → HW2–HW4



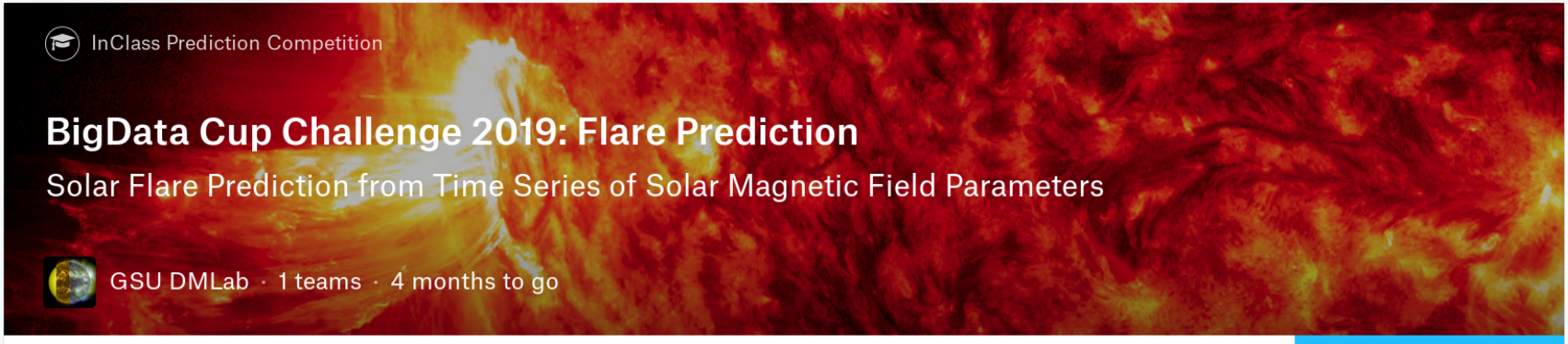
Virtual Machine . . .

1. All have exactly the same compute environment!
   (e.g., Python 3, Python packages, compilers, . . .)
2. You can login and work via the user interface.
3. You can also login via ssh (terminal).
4. You can start Jupyter notebooks (virtual machine/your desktop).
5. Basically: Very similar to a remote compute server . . .
   (with limited compute resources)
6. Image can be downloaded from:
   `https://sid.erda.dk/sharelink/FqlW8M6qFm`

Thursday: Tutorial & Help

Outline

HW1: Competition Assignment



Overview

- Introduction
- Evaluation
- Description
- Working With Kaggle

Introduction

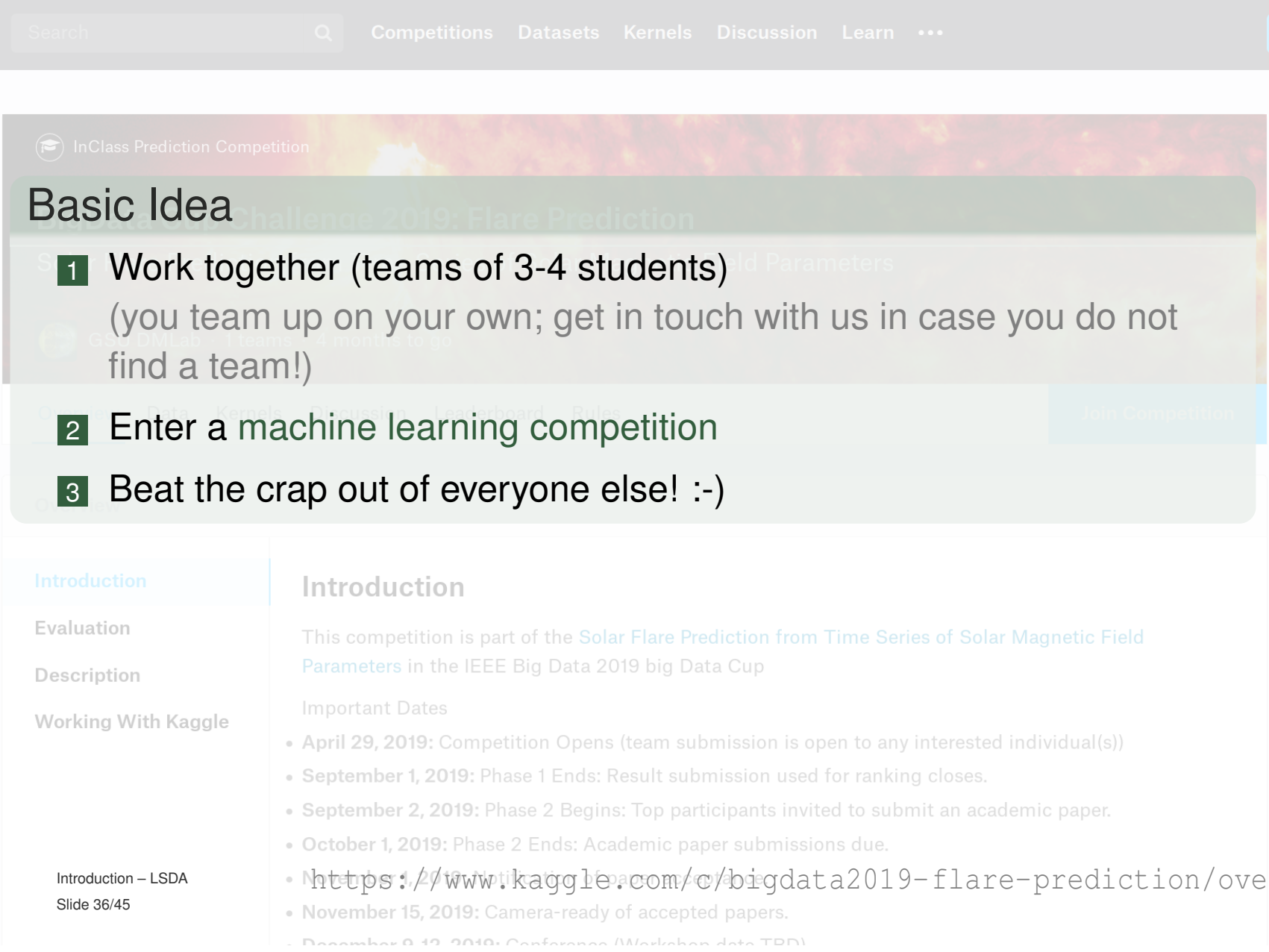
This competition is part of the Solar Flare Prediction from Time Series of Solar Magnetic Field Parameters in the IEEE Big Data 2019 big Data Cup

Important Dates

- **April 29, 2019:** Competition Opens (team submission is open to any interested individual(s))
- **September 1, 2019:** Phase 1 Ends: Result submission used for ranking closes.
- **September 2, 2019:** Phase 2 Begins: Top participants invited to submit an academic paper.
- **October 1, 2019:** Phase 2 Ends: Academic paper submissions due.
- **November 1, 2019:** Notification of paper acceptance.
- **November 15, 2019:** Camera-ready of accepted papers.
- **December 9-12, 2019:** Conference (Workshop date TBD)

https://www.kaggle.com/c/bigdata2019-flare-prediction/overview

HW1: Competition Assignment



Basic Idea

1. Work together (teams of 3-4 students)
   (you team up on your own; get in touch with us in case you do not find a team!)
2. Enter a machine learning competition
3. Beat the crap out of everyone else! :-)

https://www.kaggle.com/c/bigdata2019-flare-prediction/overview

HW1: Remark 1

For participating in the competition, at least one of the team members will have to create a Kaggle account. You might also need a Google account in order to use Google Colabs. If no member of a team is willing to do so, please get in touch with us!

HW1: Remark 2

IEEE BigData 2019

Solar Flare Prediction from Time Series of Solar Magnetic Field Parameters
IEEE BigData 2019
Los Angeles, CA, USA

Overview
Important Dates
Data
Submissions
Leaderboard
Organizers
Sign Up

Solar Flare Prediction from Time Series of Solar Magnetic Field Parameters

A Track in the IEEE Big Data 2019 Big Data Cup

Submissions

The data competition is hosted through Kaggle and the participants classification results are submitted through their platform. Teams will be limited to 2 submissions a day.

https://www.kaggle.com/c/bigdata2019-flare-prediction/submissions

Evaluation of Results

The competition will utilize the built-in FBeta-Score of Kaggle, with Beta set to one, as a way to evaluate the performance of each submission. There are both public and private leaderboards that are based on different partitions of the testing dataset, so participants are encouraged to keep this fact in mind and not overtrain their model for achieving the best results on the public leaderboard. The winning prediction method(s) will be evaluated on the following: (1) 30% coming from their rank on the private leaderboard, (2) 10% from their rank on the public leaderboard, and (3) 60% from the quality of the accompanying paper describing their methods and results. As this task is intended to help identify physical mechanisms that indicate the possible occurrence of or are the cause of solar flares, interpretability of results shall be given extra weight in the evaluation of the accompanying paper.

Your Code

Source code will also be required to be submitted, either through a publicly available repository on a Git based version control hosting service such as GitHub or BitBucket, or as code directly shared on Kaggle as a Kernel. Since our work is publicly funded, all source code is expected to be released as open-source software, utilizing some generally accepted licensing such as Apache License 2.0, GNU General Public License, MIT license, or others of similar acceptance by the Open Source Initiative.

Your Paper

HW1: Assessment

The assessment of the competition assignment will be based on:

1. Report: Short report (max 4 pages), **one** report per team
   (point out individual contributions, make use of Latex template `https://www.ieee.org/conferences/publishing/templates.html`)
2. Presentation: **One** "flash talk" per team (1-2 minutes)
   (each team member has to participate in the presentation)
3. Kaggle rank: Internal comparison between all LSDA teams.

Important: Individual scores!

- Team score as basis
- Individual contributions to report/presentation/code
- We might invite teams for a personal discussion.
- Again: We will double check the scores at the end! In particular, we will have another look at the team score and the individual contributions.

Workflow & Hints → See HW1

1. Start as soon as possible!!!
2. Find a team and enter one of the groups on Absalon ("Competition (HW1) X")
3. Decide for a **Kaggle team name**. Use "LSDA_" as prefix (e.g., LSDA_CoolTeam).
4. **Send one e-mail per team to me until Friday, April 26, 12:00**:
   1. Subject: "Kaggle Details"
   2. Content: (1) Kaggle team name, (2) Absalon group name/number ("Competition (HW1) X"), and (3) team members (KU ids)
5. Get acquainted with competition task and data.
   - `https://www.kaggle.com/c/bigdata2019-flare-prediction/kernels`
6. You **are allowed** to use any programming language/tools! Also, you can use publicly available code! Check competition rules.
7. Advice: Make use of Google Colab!
8. Advice: Try to come up with a **simple (!)** baseline method
9. **Advice: Make your first submission on Kaggle! This week?**
10. Try, evaluate, improve, repeat
11.

Some Links . . .

- https://www.kaggle.com/c/bigdata2019-flare-prediction/kernels
- http://colab.research.google.com/
- https://towardsdatascience.com/setting-up-kaggle-in-google-colab-ebb281b61463
- https://www.kaggle.com/c/bigdata2019-flare-prediction/overview/working-with-kaggle

Next Steps

What you have to do next ...

1. HW1: We have created slots for teams for up to 4 students on Absalon. Find a team and enter one of the slots!
   (let us know if you cannot find any team mates asap)
2. Get started with the assignments:
   - Team work: Get started with HW1
     (Hint: Start as soon as possible!)
   - Individually: Get started with HW2 (next week)
3. Install VirtualBox and LSDA image (see Absalon)
   - Help: Tutorial on Thursday, April 25 (12:15 – 14:00)
4. Check Absalon regularly!
5. Next lecture: Thursday, 9:15 (Lille UP1)

Questions?

Outline

Summary & Outlook

Today

1. Big Data
2. Organization
3. Competition

Next Steps

1. Get started with HW1. Get in touch with me if you cannot find a team until Thursday (come to me during the break).

Outlook

- Thursday (9:15-11:00): Fundamentals Computing & Large-Scale Nearest Neighbors (FG)
- Thursday (12:15-14:00): Practical sessions
  (Help Google Colab, . . .)