FCAI Finnish Center for Artificial Intelligence

REAL AI FOR REAL PEOPLE IN THE REAL WORLD

DATA SCIENCE FOR NATURAL SCIENCES

INAR

Kai Puolamäki
kai.puolamaki@helsinki.fi
17 January 2019

HiDATA HELSINKI CENTRE FOR DATA SCIENCE

HIIT | HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

SUMMER INTERNSHIPS

My group has several 3 month **summer intern positions** available for Bachelor's or Master's degree students from University of Helsinki or other Finnish universities, during summer 2019.

Please contact me if interested!

http://www.edahelsinki.fi/jobs.html

OFFICIAL COURSE BUSINESS

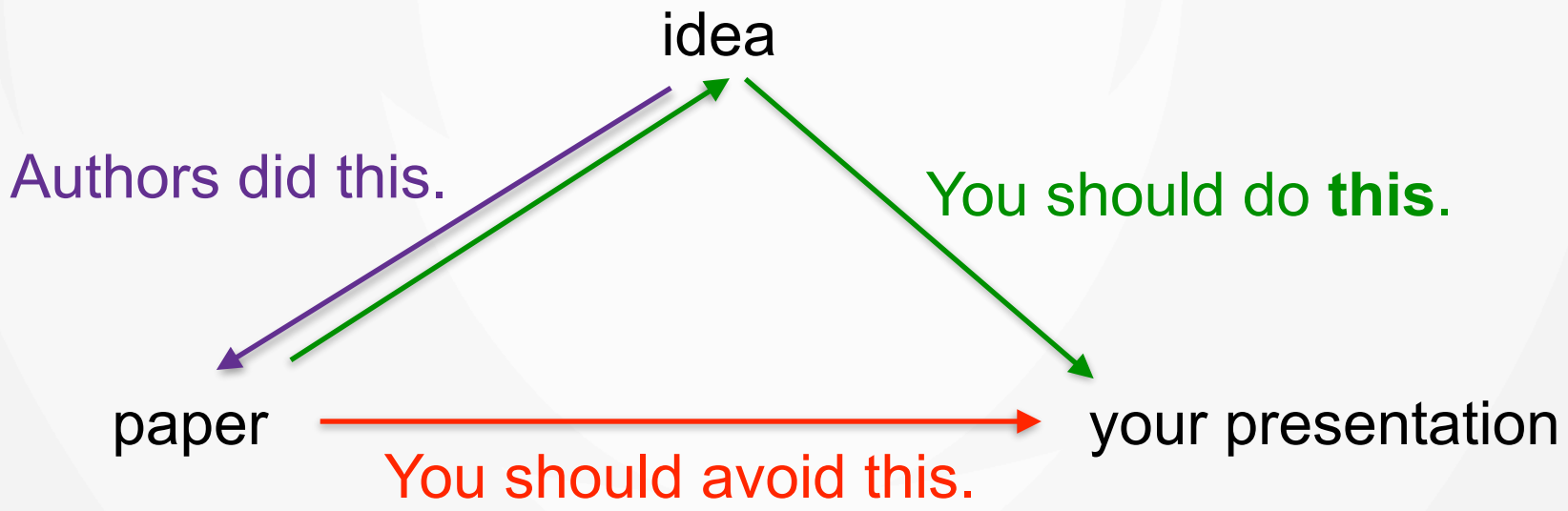
- For details see http://www.edahelsinki.fi/dsns2019/course.html
- Schedule
- Selecting topics (by 24 Jan, *optional Q&A session on 24 Jan*)
- Pitch talks (7 Feb, **next session**)
- Roles:
  - 1 presentation
  - 2 opponent duties
  - participation
- Requirements and grading
- Missed sessions or late submission will affect grading
- Assistant: Anton Björklund (anton.bjorklund@helsinki.fi)

SELECTING TOPICS

- We have a list of proposed topics
- You can also propose own topic
- Idea: computational methods applied to natural sciences

HOW TO MAKE A GOOD PRESENTATION

- Discuss an *idea* instead of a *paper!*
- Do not talk about stuff that you don't understand!



HOW TO MAKE A GOOD PRESENTATION

- Discuss an idea instead of present a paper.
- Do not talk about things that you do not understand.
- Avoid random details.
- If necessary to mention some detail, explain why it is important.
- Give concrete examples.
- Focus on insights and novel results, not on background.
- Know your audience.

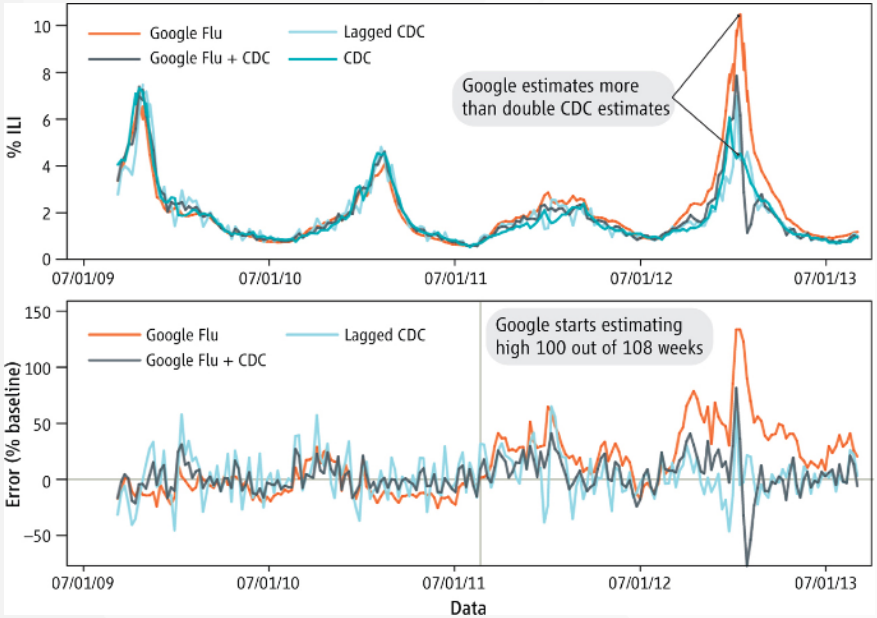
CONTENTS

- Paradigm shift and some issues
- Data management and preprocessing
- Robustness and data samples
- Supervised learning
- Interpretability of supervised learning
- Use of simulations
- (Interactive) data visualisation and dimensionality reduction
- Tools

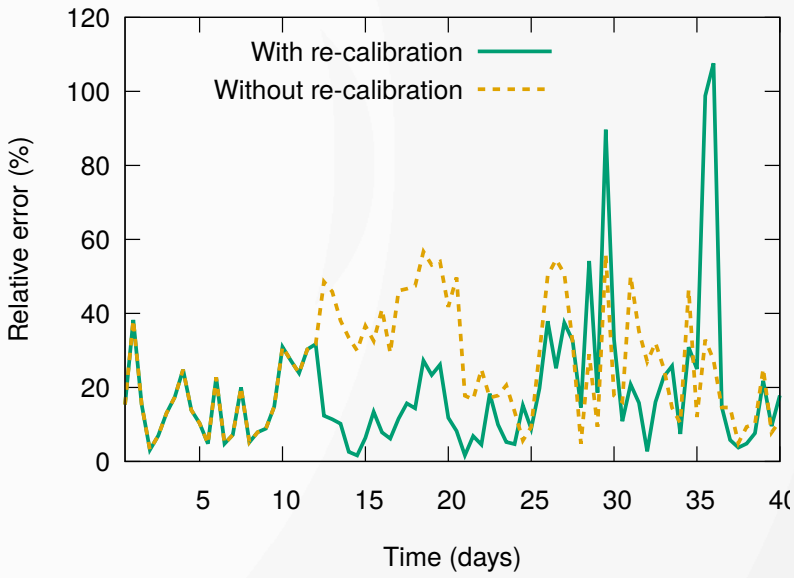
AI IN SCIENCES

- Examples of merging CS with other disciplines:
  - genetics + computer science (AI) = bioinformatics
  - physics + computer science (AI) = computational physics
- New **computational formulations** of research problems
  - p-values for individual genes → gene interaction networks
  - properties of “simple” systems → emergent behaviour of complex systems
- The disciplines become inseparable
  - revolutionary, not evolutionary, the effects are hard to foresee

THE PARABLE OF BIG DATA



From [Lazer et al., Science 2014](#).



From [Lagerspetz et al., MobiCom 2018](#).

AI CAN BE FOOLED

(a) (b)

Figure 5: Adversarial examples generated for AlexNet [9].(Left) is a correctly predicted sample, (center) difference between correct image, and image predicted incorrectly magnified by 10x (values shifted by 128 and clamped), (right) adversarial example. All images in the right column are predicted to be an *"ostrich, Struthio camelus"*. Average distortion based on 64 examples is 0.006508. Plase refer to `http://goo.gl/huaGPb` for full resolution images. The examples are strictly randomly chosen. There is not any postselection involved.

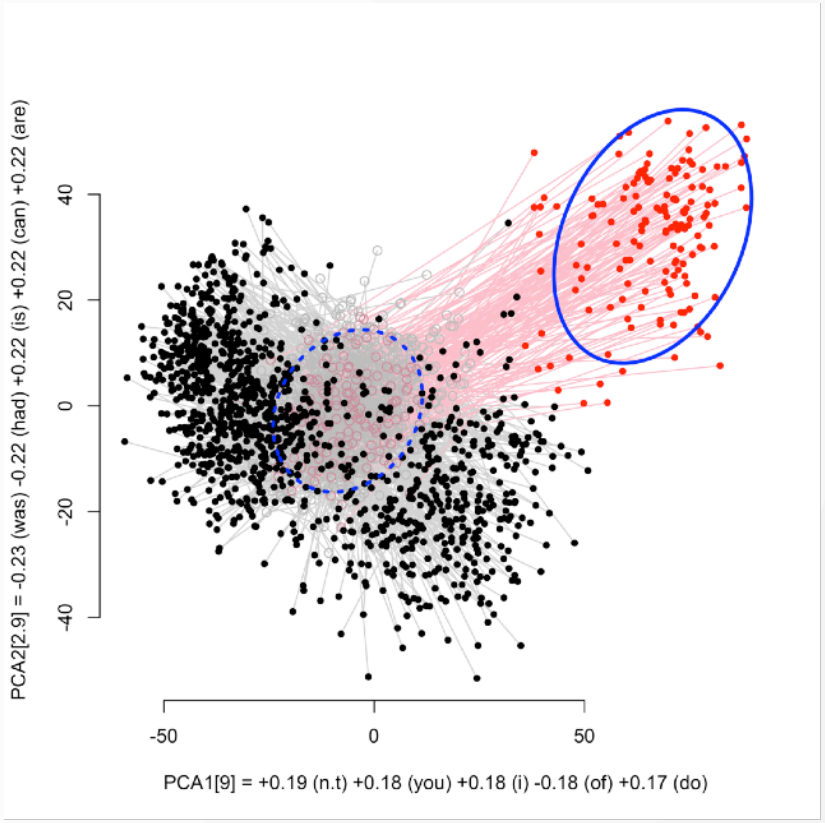
From Szegedy et al. 2013.

NEURAL NETWORK SAYS THAT THIS IS A WOLF (IT IS NOT)

From Ribeiro et al., KDD 2016.

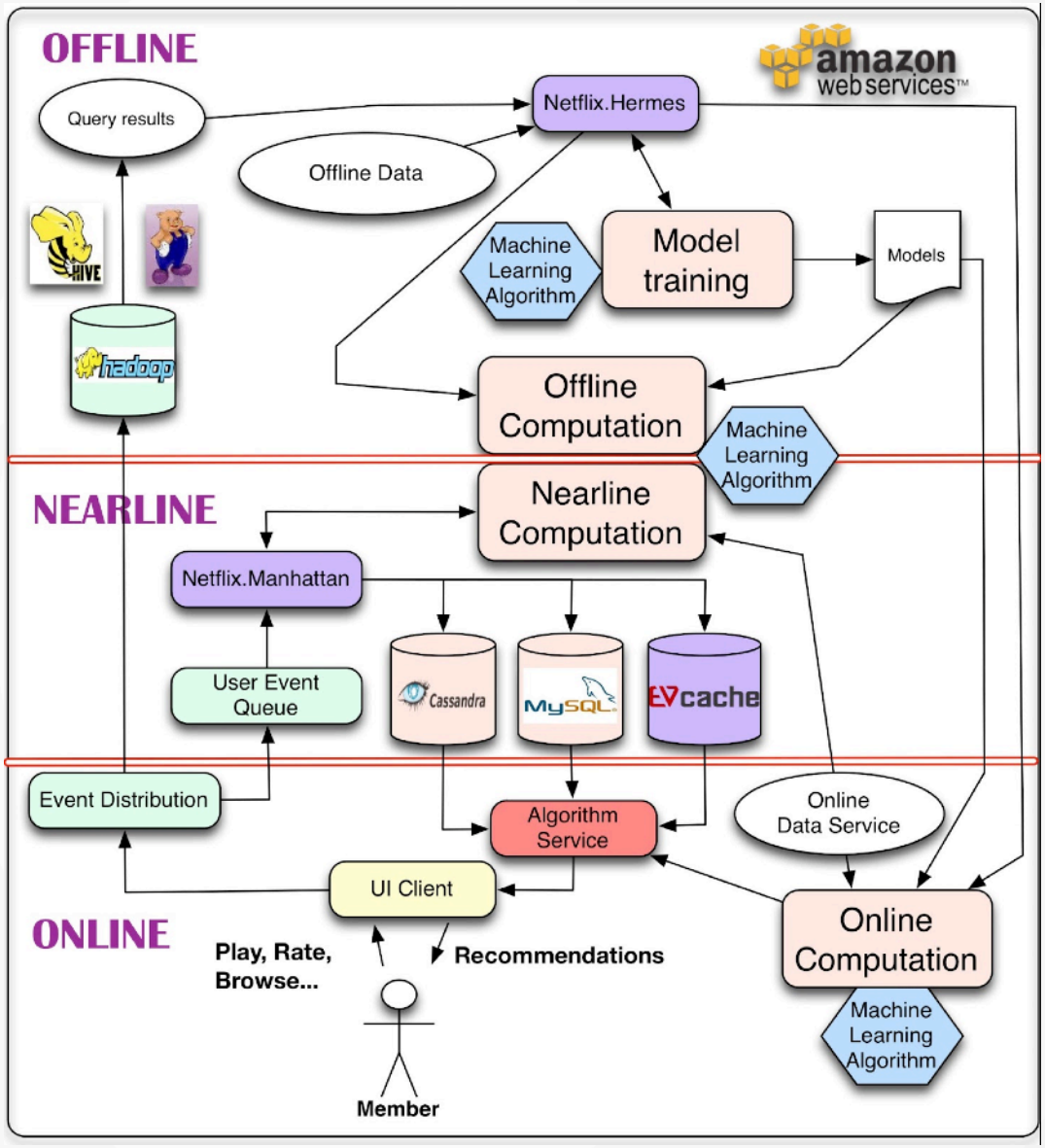
TELL ME SOMETHING OBVIOUS

- Often, AI finds something that is already obvious for the expert
- ***Dimensionality reduction*** finds low-dimensional embedding of the data (e.g., scatterplot)

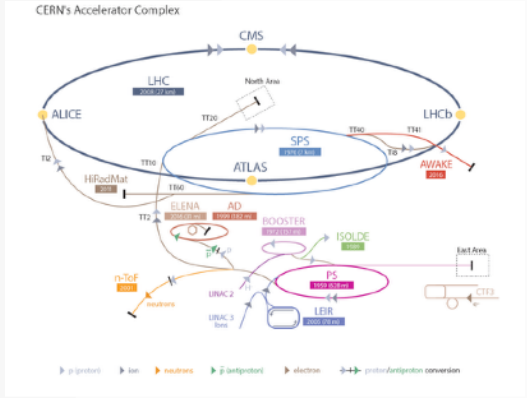


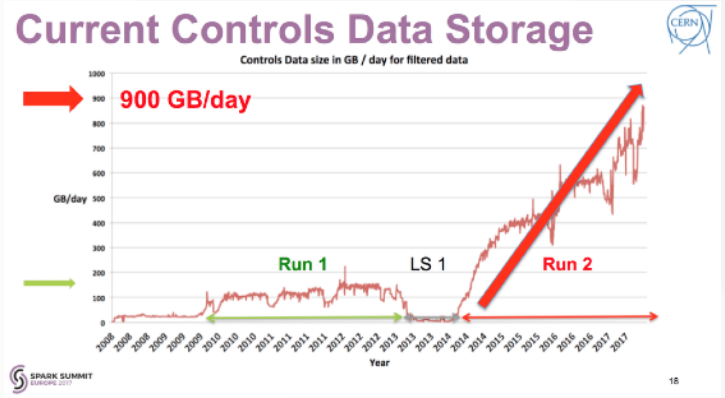
MANAGING DATA AND CODE (BEFORE ANALYSIS)

Netflix system architecture

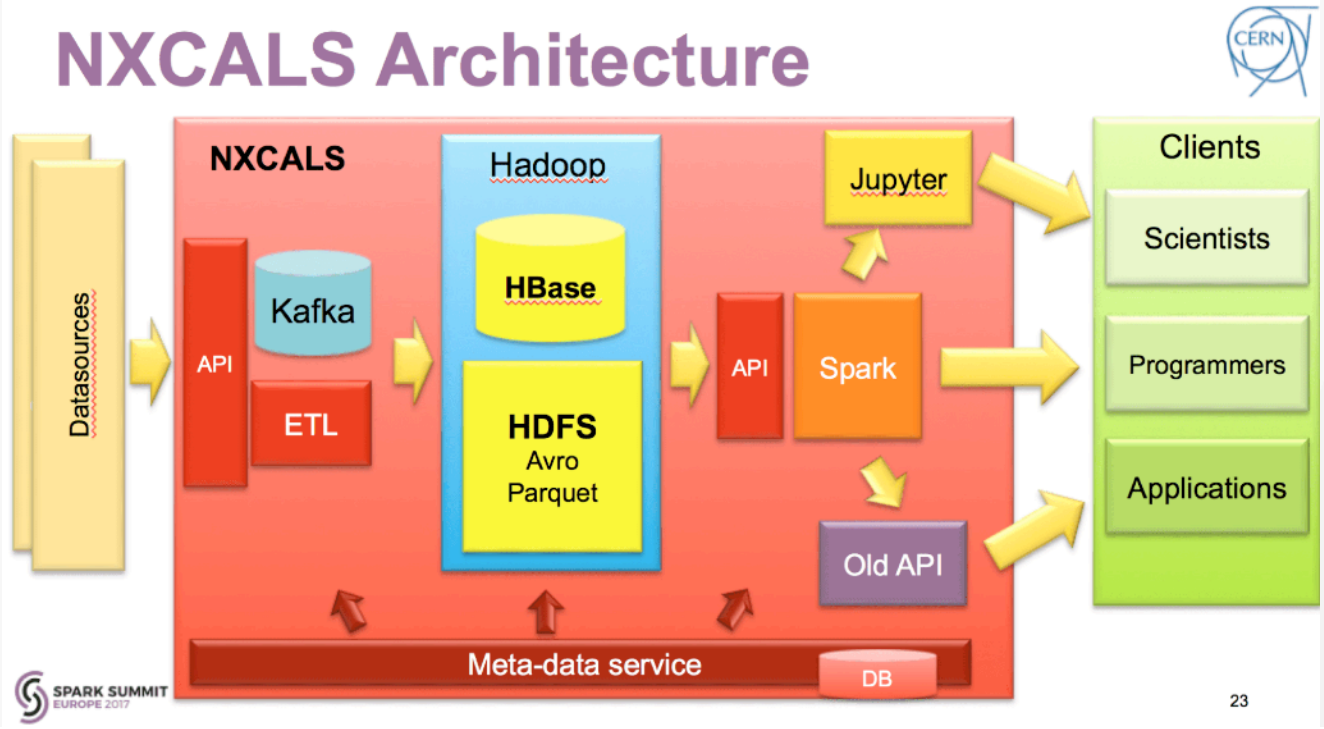


From [Amatriain, Basilico 2013](#).





CERN accelerator logging service



From [Wozniak 2017](#).

DATA UNDERSTANDING AND DATA PREPARATION

- Claim: 80% of the work on data mining project is about **data understanding** and **data preparation**
- Cross Industry Standard Process Data Mining (CRISP-DM)

Figure by K. Jansen.

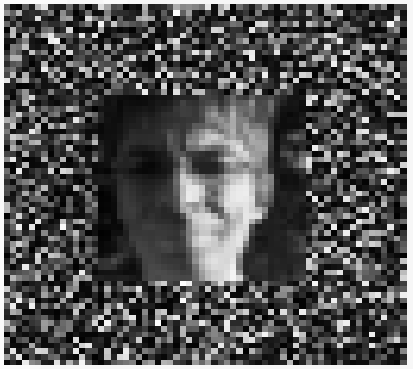
DATA WRANGLING CHALLENGES

- **DP** Data parsing, e.g., converting csv's or tables
- **DD** Obtaining (or inferring) a data dictionary: basic types + semantics
- **DI** Data integration: Combining data from multiple sources
- **ER** Entity resolution: Recognising that two distinct pieces of information in the data concern the same entity. Includes deduplication and record linkage
- **FV** Format variability: e.g. for dates, but also for variability in names (e.g. IBM, I.B.M.).
- **SV** Coping with structural variability in the data, e.g. wide vs tall format. Also variation over time.
- **MD** Identifying and repairing missing data
- **AD** Anomaly detection and repair
- *Credit: Chris Williams*

ROBUSTNESS AND DATA SAMPLES

RANDOMISATION OF DATA



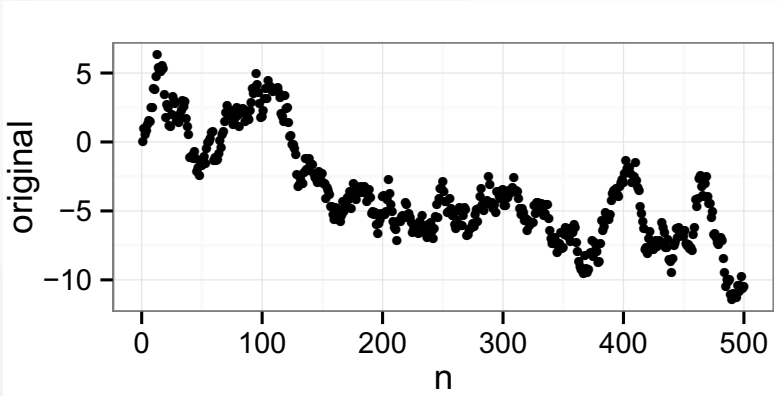


(Extremely simple randomisation schema:
pixels are permuted within a given area.)

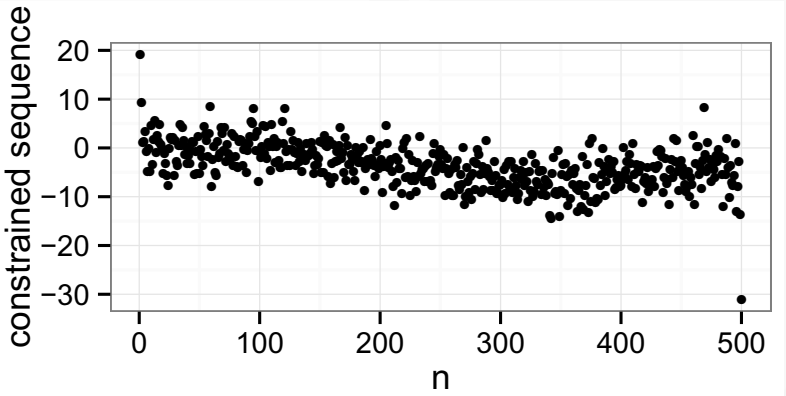
RANDOMISATION OF DATA

- Time series data

original data



randomised data



From Henelius et al., ECML PKDD 2013.

RANDOMISATION OF DATA

- Tabular data
- Columns are permuted uniformly in random

original data

A1	B1	C1	D1	E1
A2	B2	C2	D2	E2
A3	B3	C3	D3	E3
A4	B4	C4	D4	E4
A5	B5	C5	D5	E5
A6	B6	C6	D6	E6

randomised data

A2	B6	C5	D4	E1
A6	B5	C2	D1	E6
A1	B4	C6	D6	E3
A3	B2	C3	D2	E4
A5	B3	C4	D5	E2
A4	B1	C1	D3	E5

Puolamäki et al. 2018.

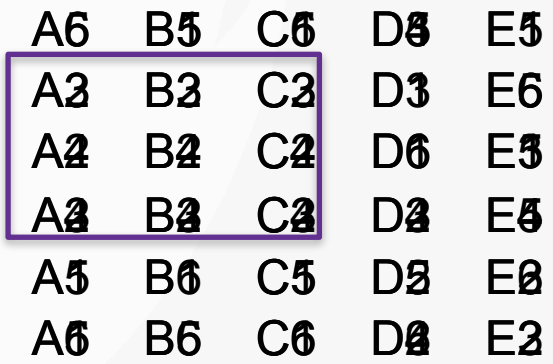
RANDOMISATION OF DATA

- Tabular data
- Randomisation is constrained by a *tile*

original data

A1	B1	C1	D1	E1
A2	B2	C2	D2	E2
A3	B3	C3	D3	E3
A4	B4	C4	D4	E4
A5	B5	C5	D5	E5
A6	B6	C6	D6	E6

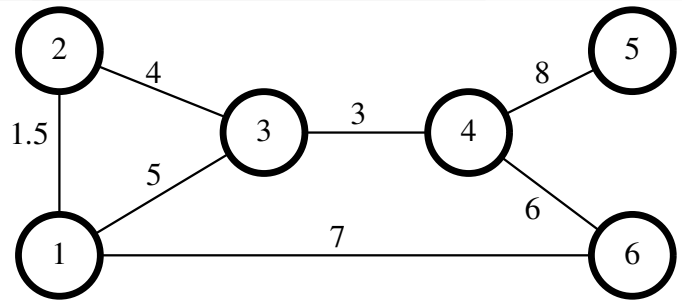
randomised data

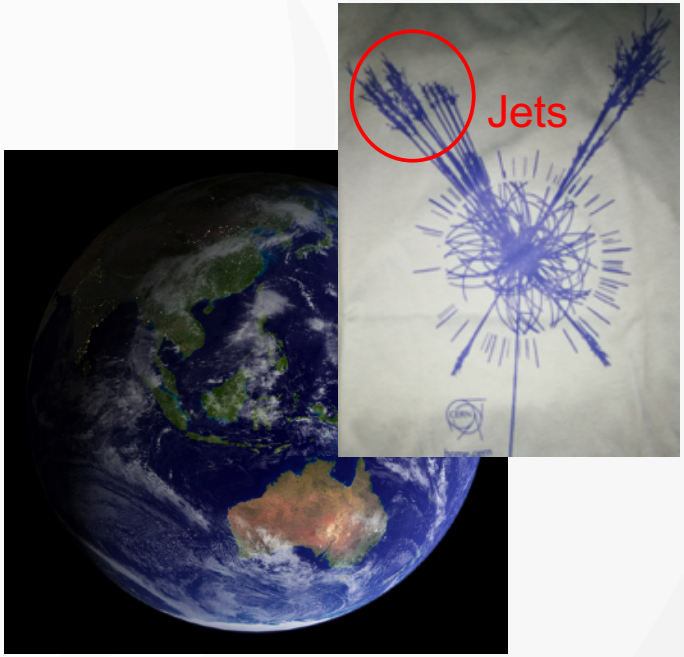


[Puolamäki et al. 2018](#).

RANDOMISATION OF DATA

- Network data
- Outputs of simulators

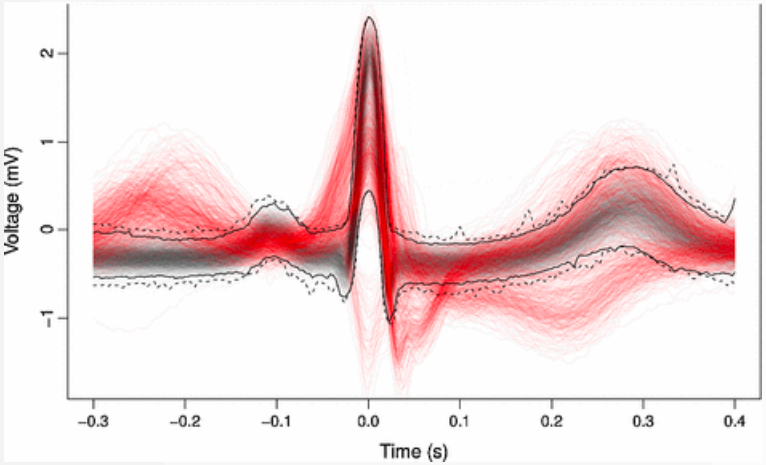




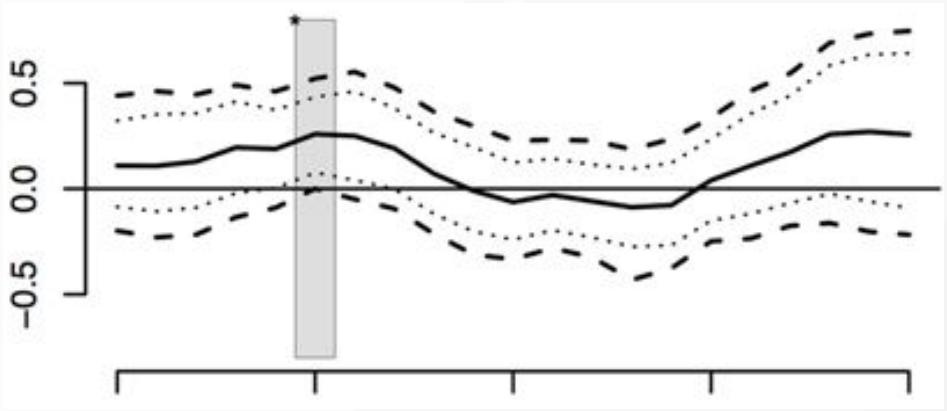
From Puolamäki et al. 2018.

IS IT SOMETHING REAL?

- Randomisations make it possible to see if observations are random (i.e., different from randomised data)
- Traditional statistical methods are often insufficient - new methodology needed!



From Korpela et al., DAMI 2014.



From Ahonen et al., Scientific Reports 2018.

SUPERVISED LEARNING

SUPERVISED LEARNING

- Given:
  - training data
  - testing data
  - family of functions
  - loss function

$$D_{train} = \{(x_i, y_i)\}_{i=1}^{n}$$

$$D_{test} = \{(x_{n+i}, y_{n+i})\}_{i=1}^{m}$$

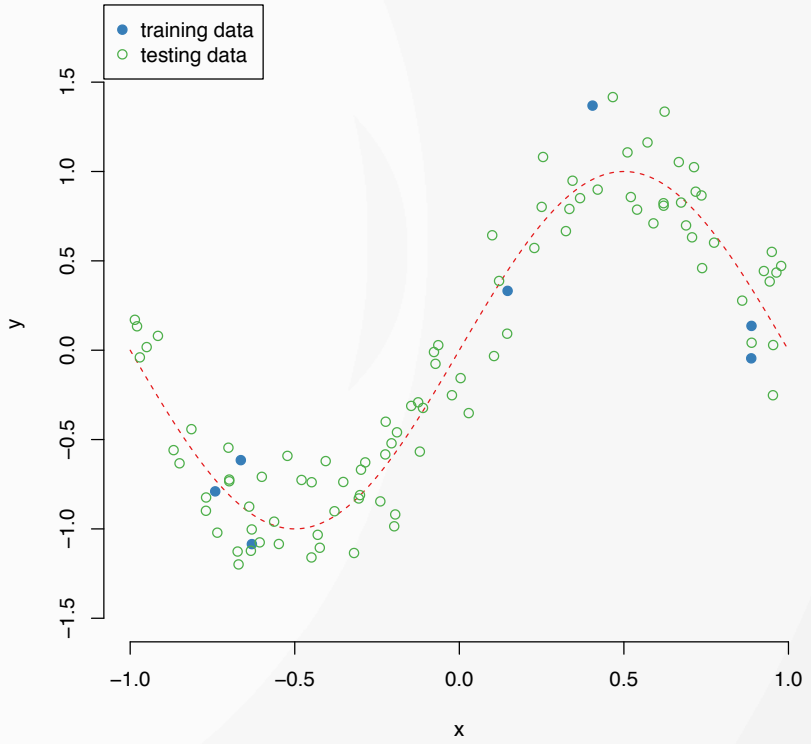
$$f : X \mapsto Y$$

$$L_D(f)$$

- **Supervised learning** problem:
  - Given training data, family of functions, and a loss function, find a function such that loss on testing data is minimised.
- *Y* discrete: **classification**, *Y* continuous: **regression**

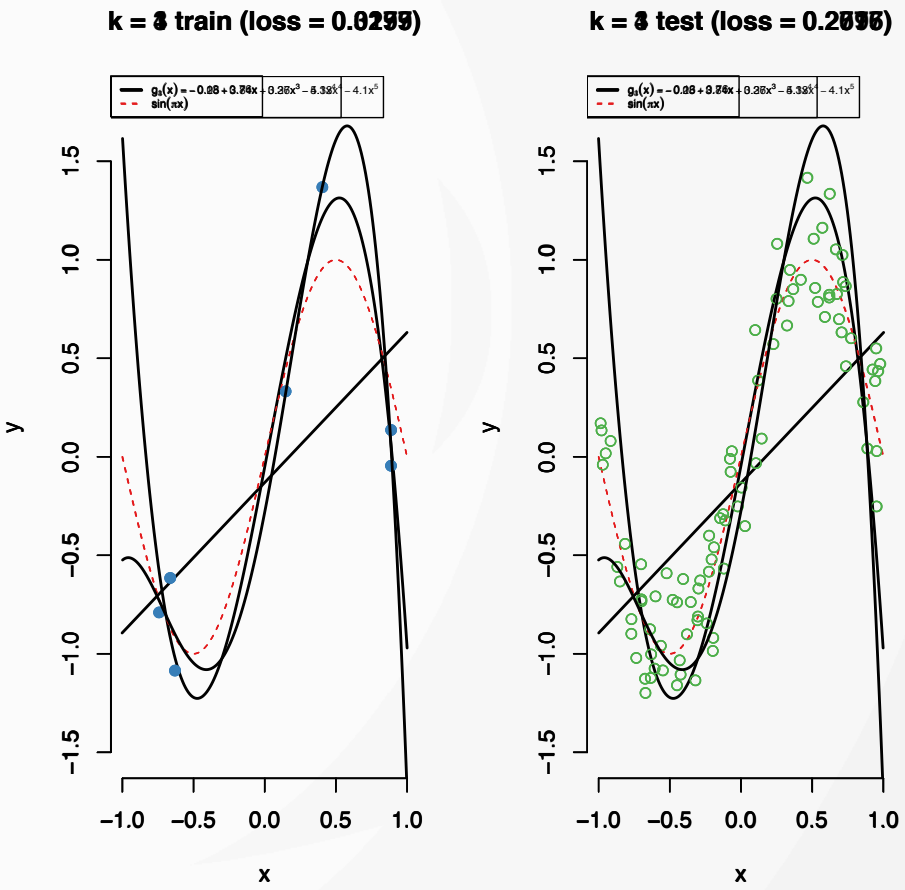
“STANDARD” EXAMPLE: OLS REGRESSION

- training data: 7 xy-points
- testing data: 93 xy-points
- family of functions: polynomials of given degree
- loss function: quadratic loss



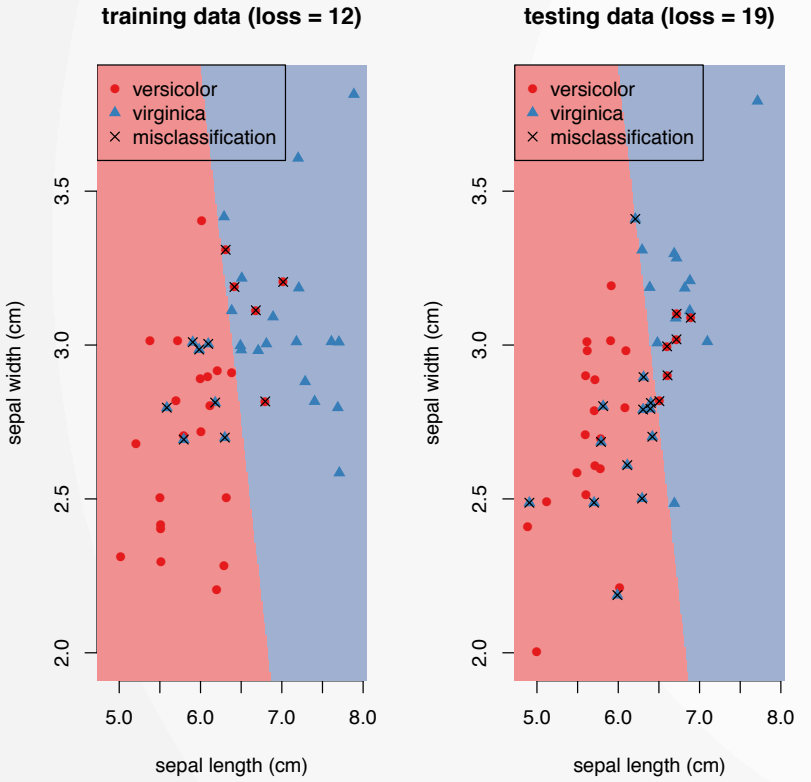
OLS REGRESSION

- **optimisation** problem
  - find polynomial of degree *k* with smallest error on training data
- **statistical** problem
  - what is the error of the estimate of the data and the model
- **learning theoretic** problem
  - how do we get best estimate on test data
- **algorithmic** problem
  - how to make a robust and fast algorithm to solve the above mentioned problems

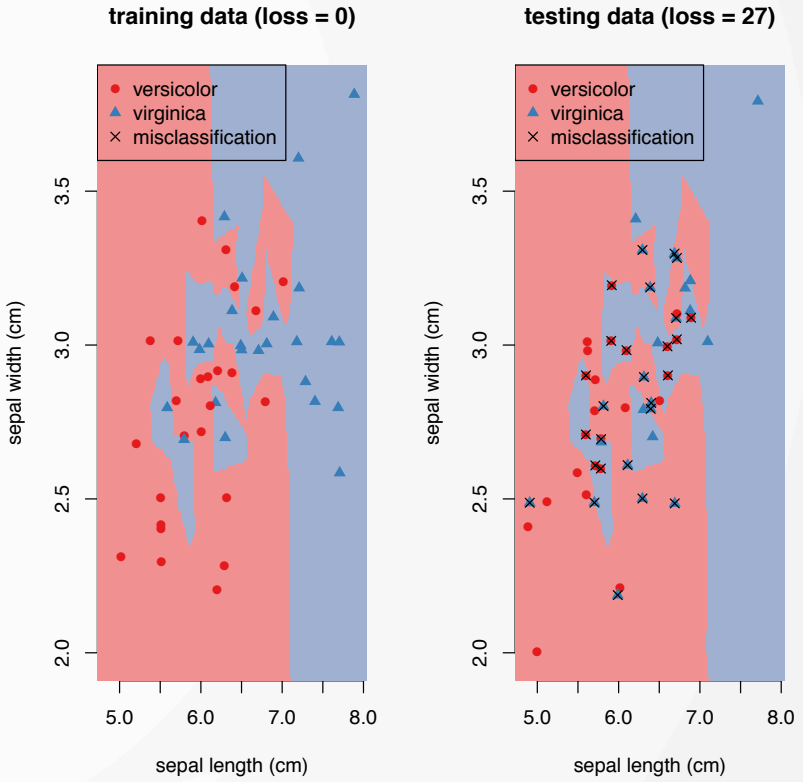


CLASSIFICATION

linear separator

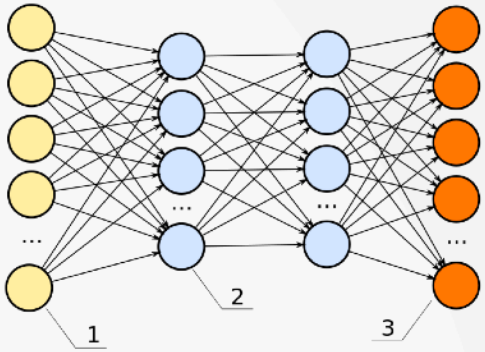


1NN classifier



Training and testing data both have 50 data points. Loss is here the number of misclassified data points.

DEEP LEARNING



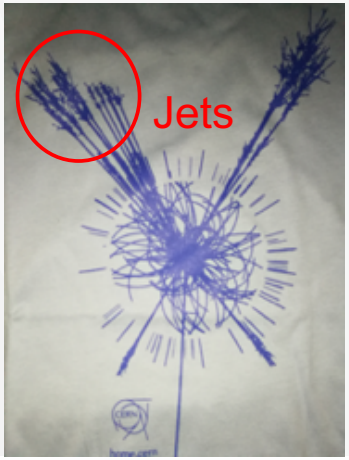
- **Deep learning** can be used, e.g., to supervised learning problems as a family of functions
- Long history, currently at the peak of the hype curve (until the next hype)
- Advantages:
  - Good software libraries: can be used without deep understanding
  - Flexible architecture: lego-like approach to problem solving
  - Good at learning features: for some problems the best-performing approach
- Disadvantages:
  - Deep learning models are usually “black box”, hard to interpret
  - Training is often computationally expensive
  - Not a golden bullet: subject to same issues as any other method, fully benefiting from DL requires also deep CS understanding

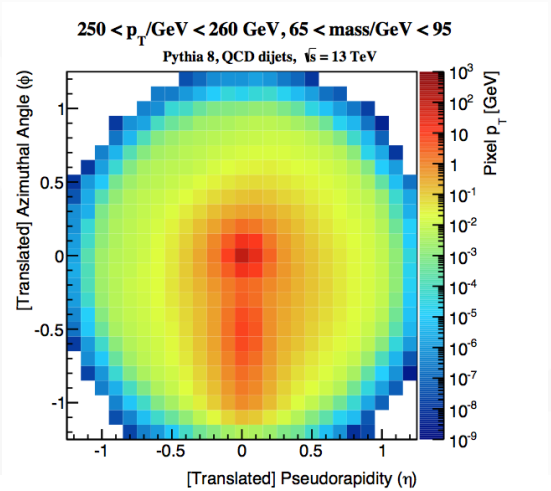
LESSON: ON UNDERSTANDING OF AI SYSTEMS

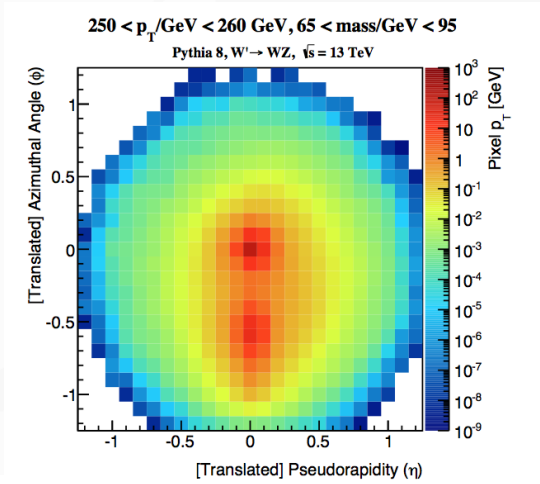
- AI systems often contain ML/AI primitives as supervised learning components
- To understand them requires…
  - Knowledge of the topic system is applied on
  - Optimisation, statistics, learning, theory algorithms, machine learning etc.
- Usually this means a multidisciplinary team that works closely together

APPLICATIONS TO PHYSICAL SYSTEMS

- Supervised learning is used, e.g., to
  - calibrate measurements or
  - classify particle physics events.
- How to make predictions robust, how to understand them?



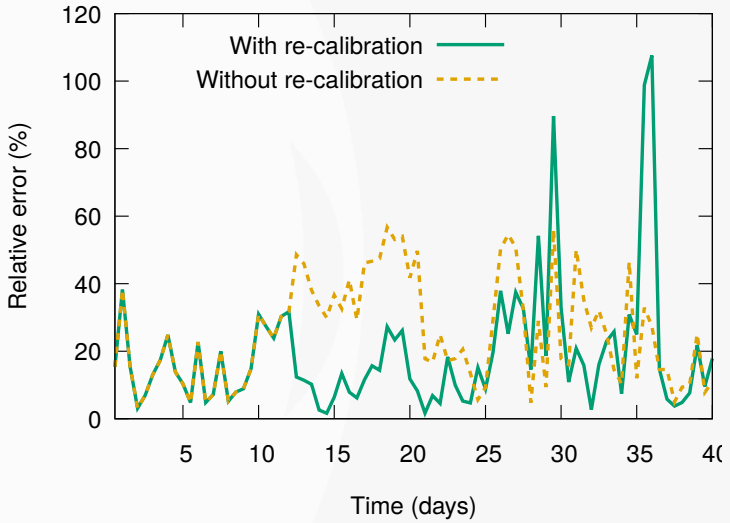




From Guest et al. 2018.

CALIBRATION OF MEASUREMENTS

- Regression functions can be used to “calibrate” low-accuracy measurements, simulator outputs etc.
- More complex regression more unstable it becomes
- Research problems:
  - How to compute confidence intervals for the estimate when the ground truth is not known?
  - How to understand the workings of the regression function etc.



Relative error in raw (low accuracy) O3 measurement (dotted amber line) and measurement calibrated with a regression function using raw measurement, temperature, and pressure as covariates (solid green line). From Lagerspetz et al. 2018.

INTERPRETABILITY OF THE SUPERVISED LEARNING

ADVERSARIAL EXAMPLES

(a) (b)

Figure 5: Adversarial examples generated for AlexNet [9].(Left) is a correctly predicted sample, (center) difference between correct image, and image predicted incorrectly magnified by 10x (values shifted by 128 and clamped), (right) adversarial example. All images in the right column are predicted to be an *"ostrich, Struthio camelus"*. Average distortion based on 64 examples is 0.006508. Plase refer to `http://goo.gl/huaGPb` for full resolution images. The examples are strictly randomly chosen. There is not any postselection involved.

Szegedy et al. 2013

BLACK BOX VS. INTERPRETABLE CLASSIFIERS

Black box

- Neural networks (DNN, RNN, CNN)
- Ensemble methods
  - Random forests
- Support vector machines

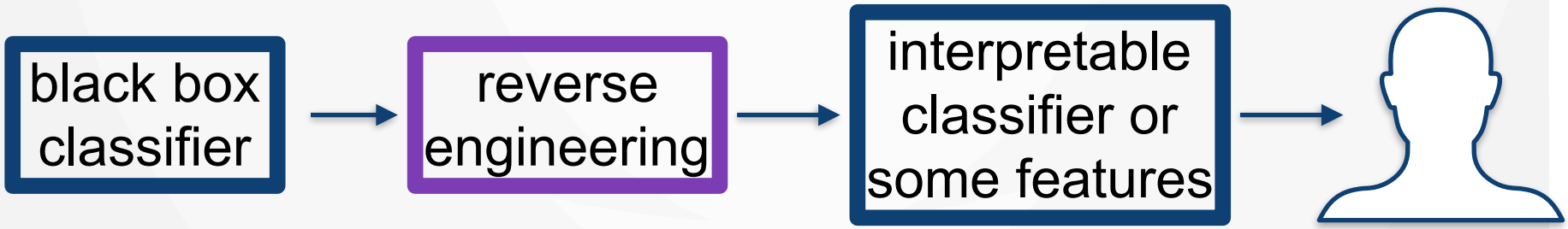
Interpretable

- Decision trees
- Classification rules
  - "Classify to class *+1* if *x>a*, otherwise to class *-1*"
- Prototype based methods
- Even these depend on data and may be non-interpretable
  - "Classify to class *+1* if *x>a*" - choice of *a* may have a drastic effect!

Guidotti et al., 2018. A Survey of Methods for Explaining Black Box Models. ACM CSUR.

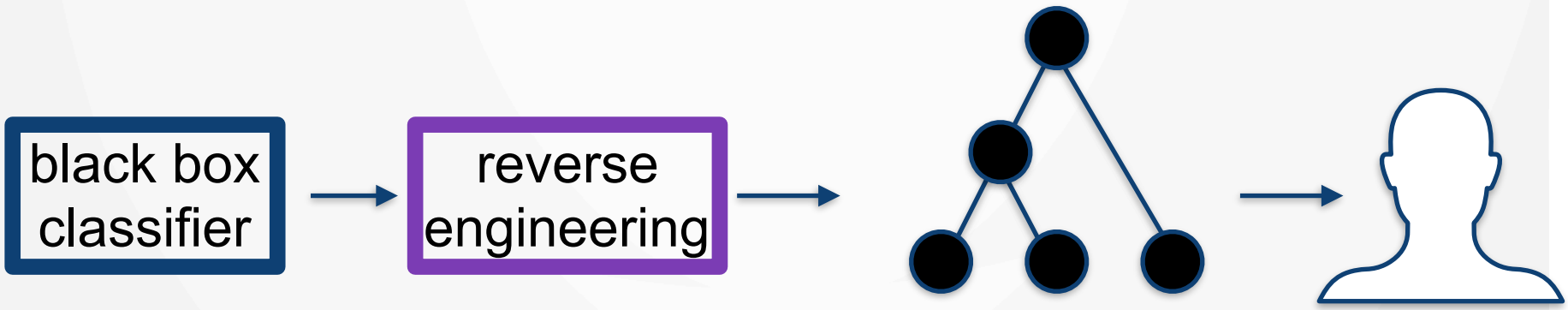
EXPLAINING THE BLACK BOX

- Design an interpretation layer between the classifier and the human user
- Methods can be classified into two categories
  - global explanations
  - local explanations
- Methods can be black-box dependent or black-box agnostic



GLOBAL EXPLAINABILITY

- Global explanations for neural networks date back to the 90s
  - Trepan (Craven et al., 1996) is a black-box agnostic method that induces decision trees by querying the black box
  - Trepan's split criterion depends on entropy and fidelity
  - Further improved in Domingos 1998 and Johansson et al. 2009.

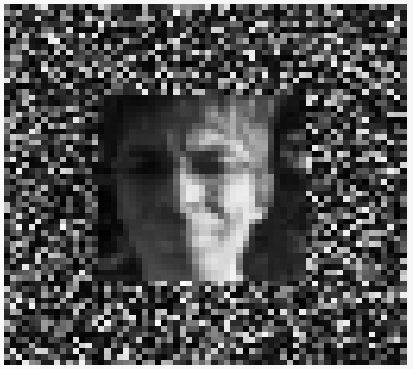


UNDERSTANDING AI ALGORITHMS

Kai

???

Kai



(Extremely simple randomisation schema:
pixels are permuted within a given area.)

GLOBAL EXPLANATIONS

- You can explain black box classifier (or regression function) by observing how it behaves on randomised data

$$f(x) = x_1 \oplus x_2$$

f(x)	x_1	x_2
1	1	0
1	1	0
1	0	1
1	0	1
0	1	1
0	1	1
0	0	0
0	0	0

randomisation breaks the classifier

f(x)	$f(x^*)$	x^*_1	x^*_2
1	0	0	0
1	0	1	1
1	1	1	0
1	0	0	0
0	0	1	1
0	1	1	0
0	1	0	1
0	1	0	1

shuffle rows within a box

randomisation does not break it

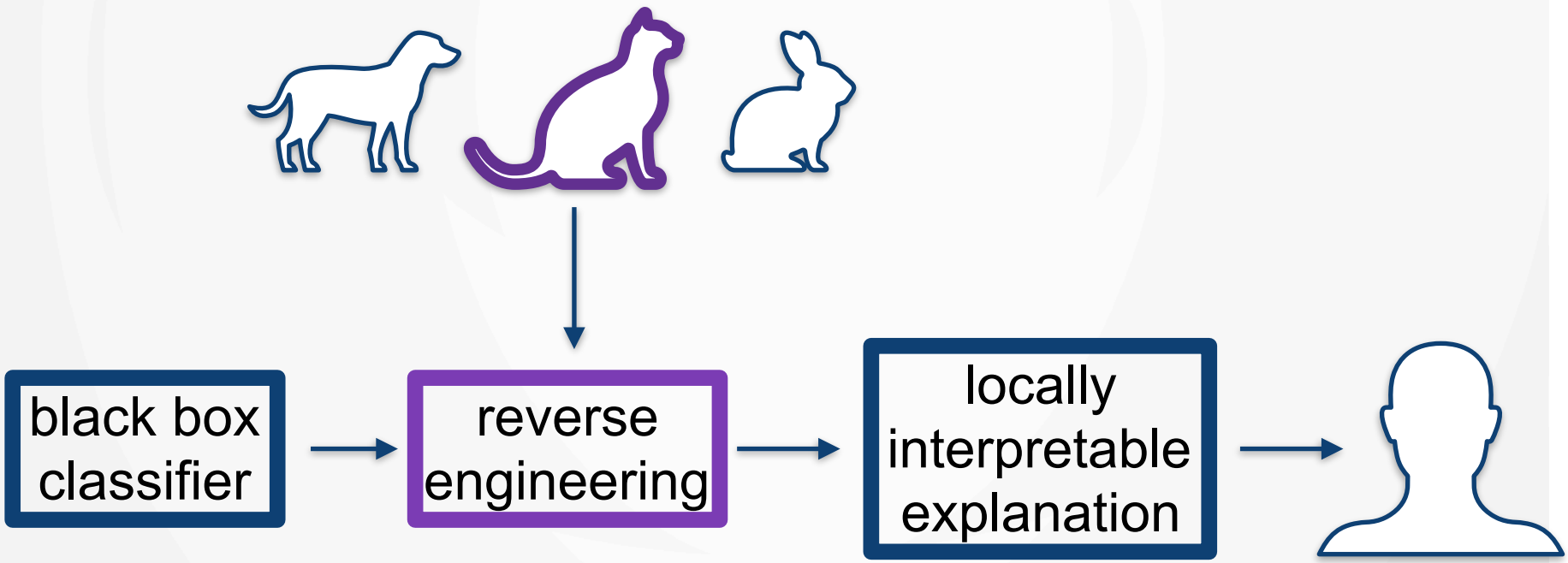
f(x)	$f(x^*)$	x^*_1	x^*_2
1	1	1	0
1	1	0	1
1	1	1	0
1	1	0	1
0	0	1	1
0	0	0	0
0	0	0	0
0	0	1	1

shuffle rows within a box

Henelius et al., DAMI 2014.

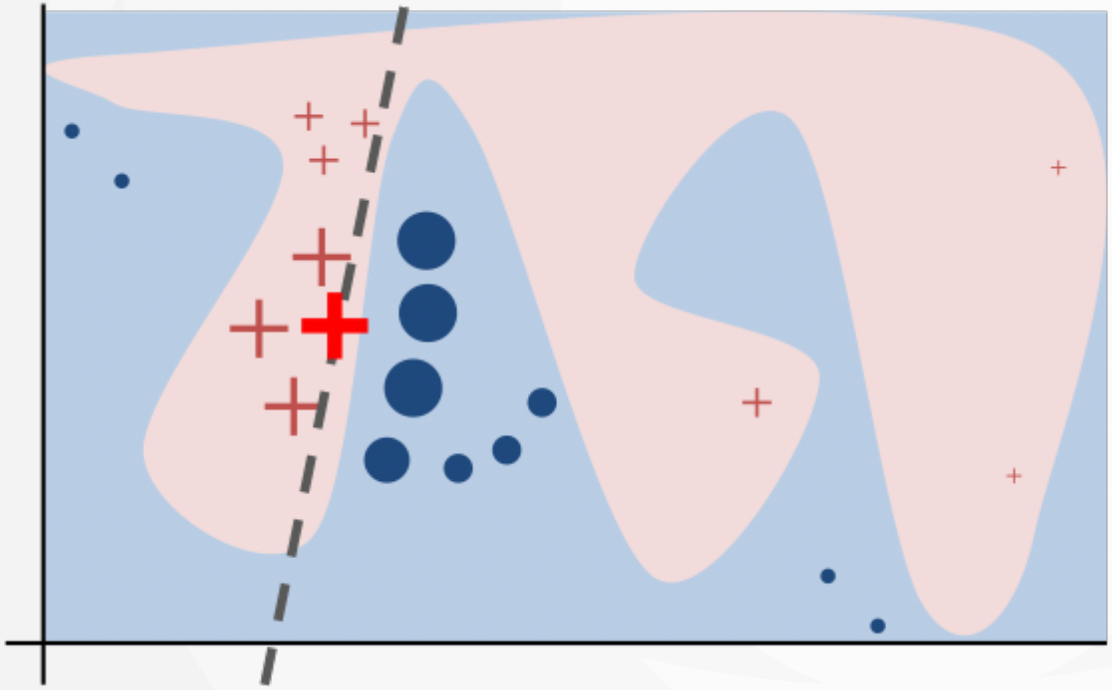
LOCAL EXPLAINABILITY

- Provide explanation only in the vicinity of a point of interest



LOCAL EXPLAINABILITY: LIME

- LIME optimises for local fidelity



Lime approximates the decision boundary at the neighbourhood of the item to be explained by a sparse linear model. From Ribeiro et al. 2016.

LOCAL EXPLAINABILITY: LIME

(a) Original Image

(b) Explaining *Electric guitar*

(c) Explaining *Acoustic guitar*

(d) Explaining *Labrador*

LIME explains various classifications. From Ribeiro et al. 2016.

LOCAL EXPLANATIONS

(a) Husky classified as wolf

(b) Explanation

Real wolf.

"Husky" is classified as "wolf", because the classifier has learned that all pictures of wolves had snow in the background. Therefore, the classifier predicts "wolf" if there is snow and "husky" otherwise, regardless of other features (animal colour, position, pose etc.). From Ribeiro et al., KDD 2016.

USE OF SIMULATIONS

AI FOR SIMULATIONS

- Simulator: “black box” that takes in some parameters and outputs some data
  - E.g., atmospheric simulator, simulator of physics collisions
  - They are often stochastic in nature
- Simulators can be constructed from first principles
- They are often computationally intensive
- They cannot be easily converted, e.g., to a probabilistic model

USING GAN TO SIMULATE PARTICLE SHOWERS

- GANs can be used to learn generative models and sample from them

We trained a Generative Adversarial Network using 30,000 celebrity photos (CelebA-HQ)

The network learned to generate entirely new images that mimic the appearance of real photos

From Karras et al. 2018.

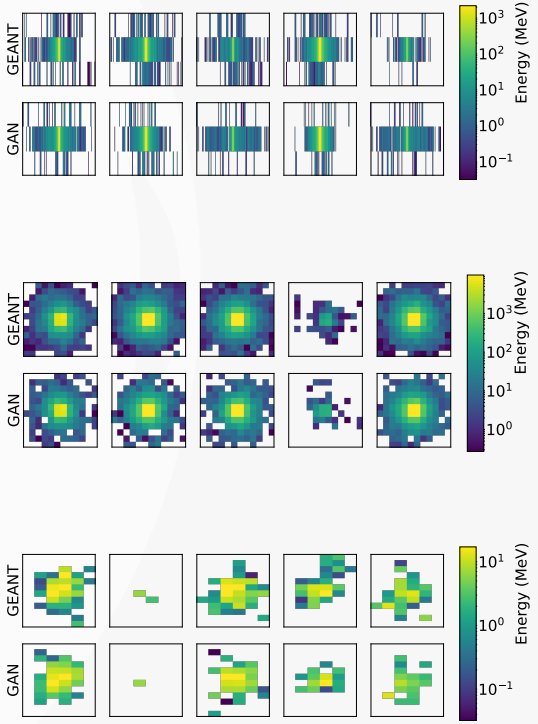


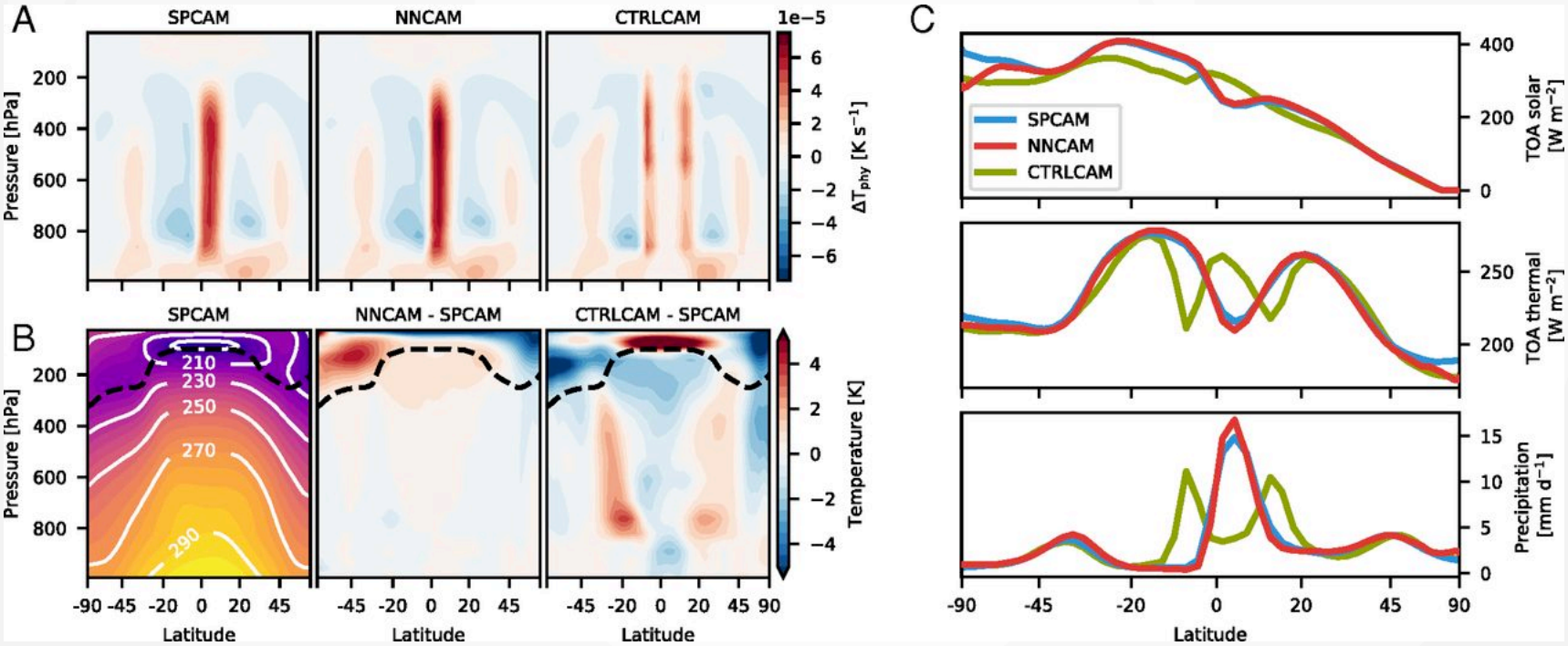
FIG. 2. Five randomly selected γ showers per calorimeter layer from GEANT4 (top) and their five nearest neighbors (by euclidean distance) from a set of CALOGAN candidates.

From Paganini et al. 2018.

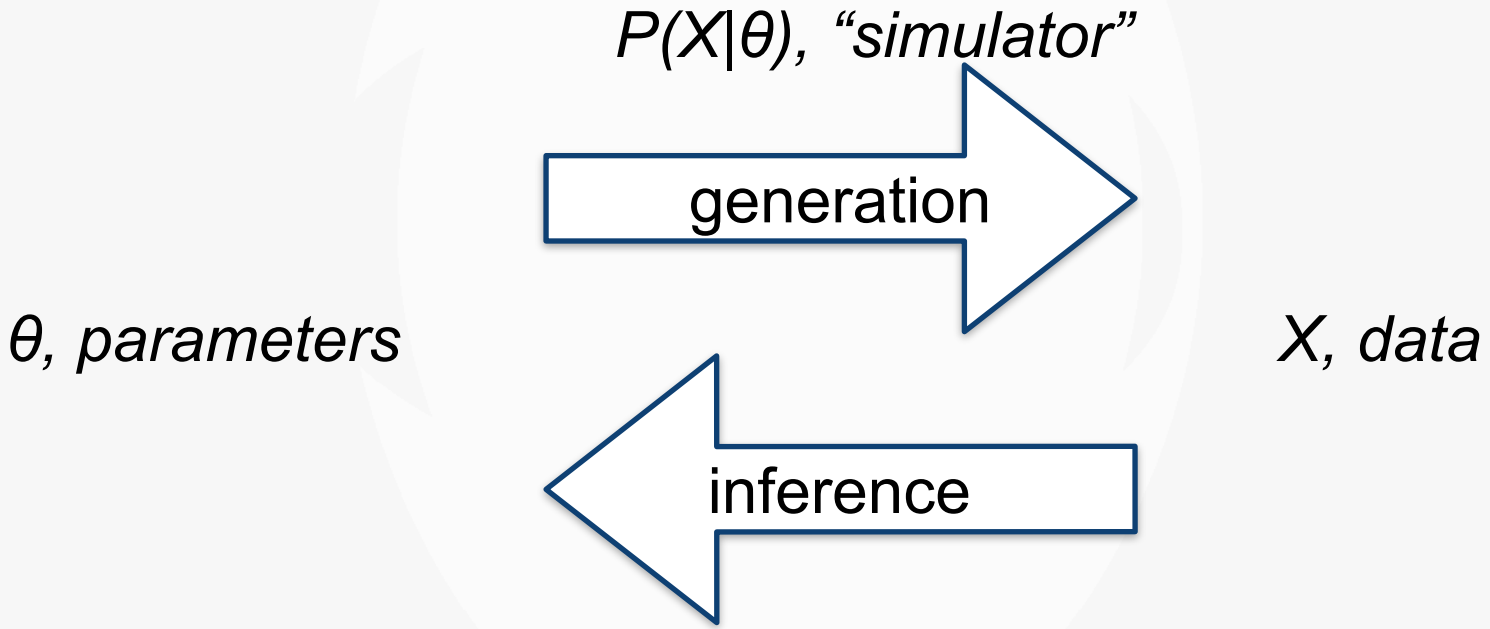
USING DNN TO REPLACE CLIMATE SIMULATION

From Rasp et al. 2018.

simulation DNN



GENERATION VS. INFERENCE



LIKELIHOOD-FREE INFERENCE

- Problem: sample posterior $P(\theta|X)$ when we can sample from $P(X|\theta)$ ("simulator with parameters θ") and prior $P(\theta)$ but the values of $P(X|\theta)$ and $P(\theta)$ are not known.
- Bayes rule: $P(\theta|X) \propto P(X|\theta)P(\theta)$
- *Approximate Bayesian Computation* (ABC) algorithm, input data X:
  - $\theta^* \sim P(\theta)$
  - $X^* \sim P(X|\theta^*) = SIMULATOR(\theta^*)$
  - if $X \approx X^*$ output θ^*
  - repeat
- Claim: **outputs are (approximately) samples from $P(\theta|X)$**.

FORWARD MODELLING IN COSMOLOGY

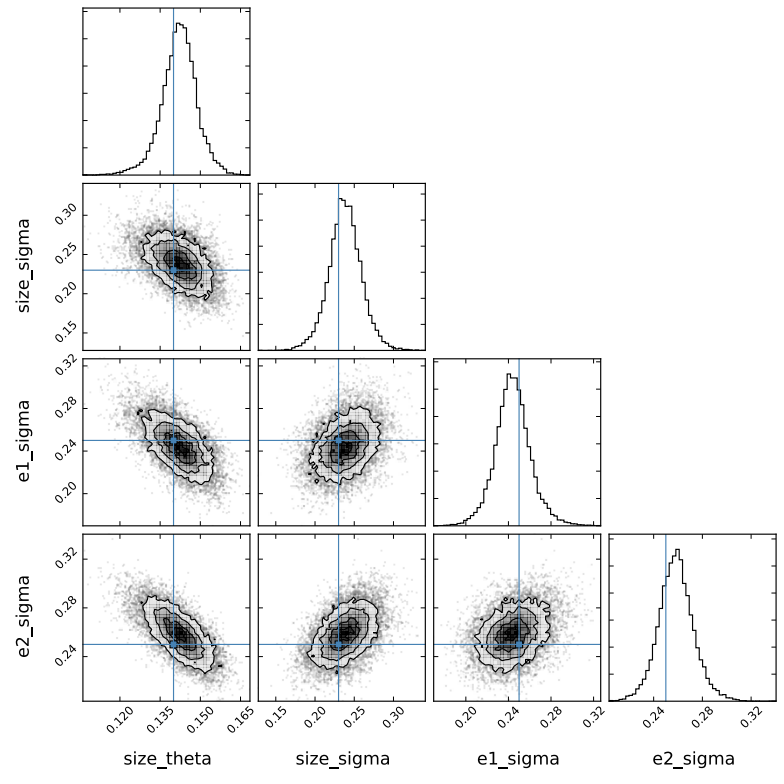


Figure 5. The one- and two-dimensional marginal distributions of the approximate `UFig` parameter posterior. The blue lines denote the true initial parameter configuration. Created with `triangle.py`. [65]

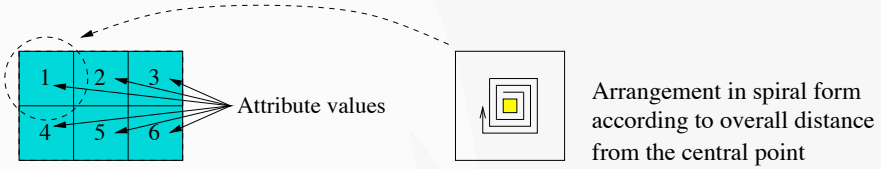
From Akeret et al. 2015.

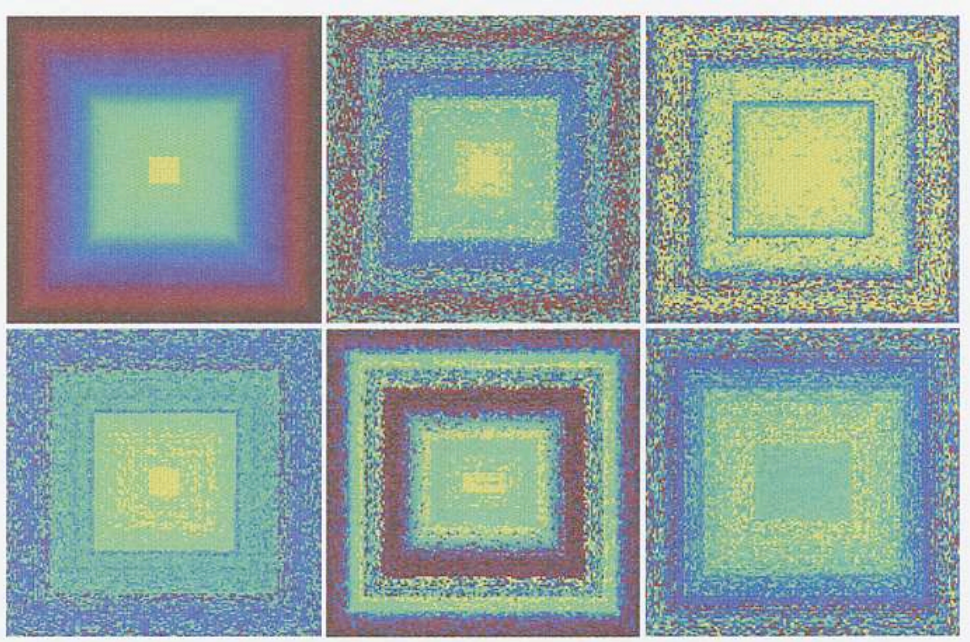
VISUALISATION

- Extremely large and important topic (NB: I had a course on the topic https://mycourses.aalto.fi/course/view.php?id=16959…)
- Human feedback in tuning machine learning model parameters
- Visual analytics
- Software tools and techniques
- Visualisation techniques
- Human perception
- Color
- Cognition
- User testing
- Interactivity etc.

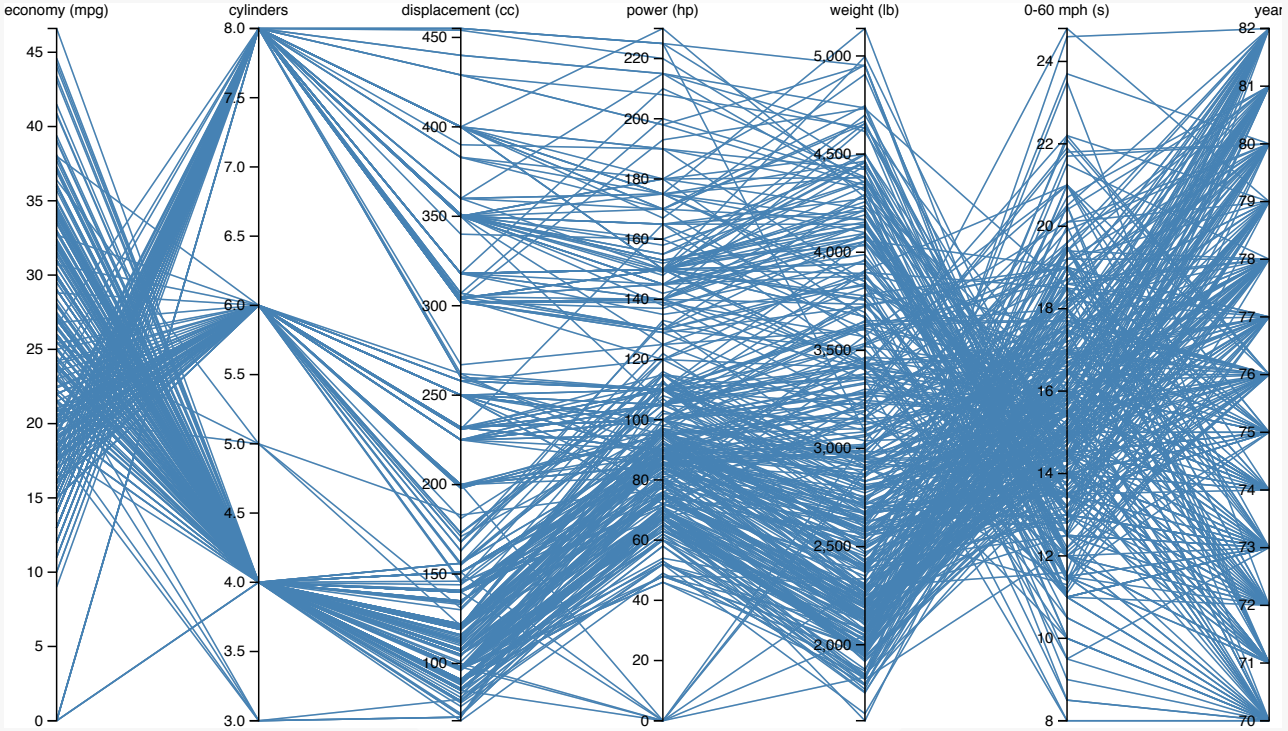
PIXEL-ORIENTED TECHNIQUES

- Each attribute value is represented by one pixel (the value ranges are mapped to a fixed colour-map)
- The attribute values for each attribute are represented in separate sub-windows





PARALLEL COORDINATES



https://bl.ocks.org/jasondavies/1341281

PERCEPTION MATTERS

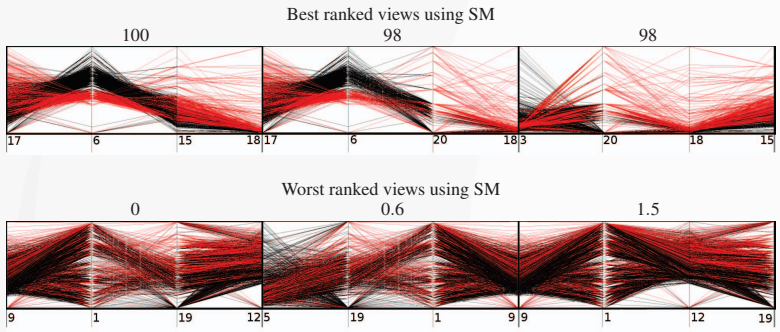


Fig. 6. Ranked list of four-dimensional parallel coordinates. Best ranked on top, worst ranked on the bottom [48].

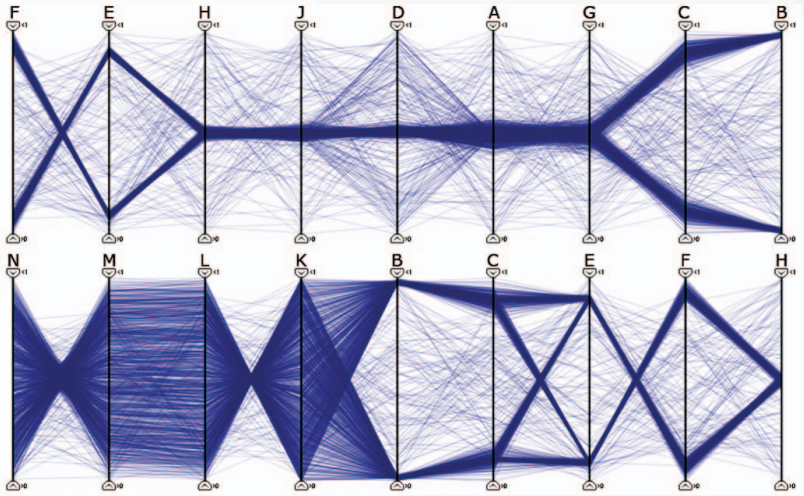
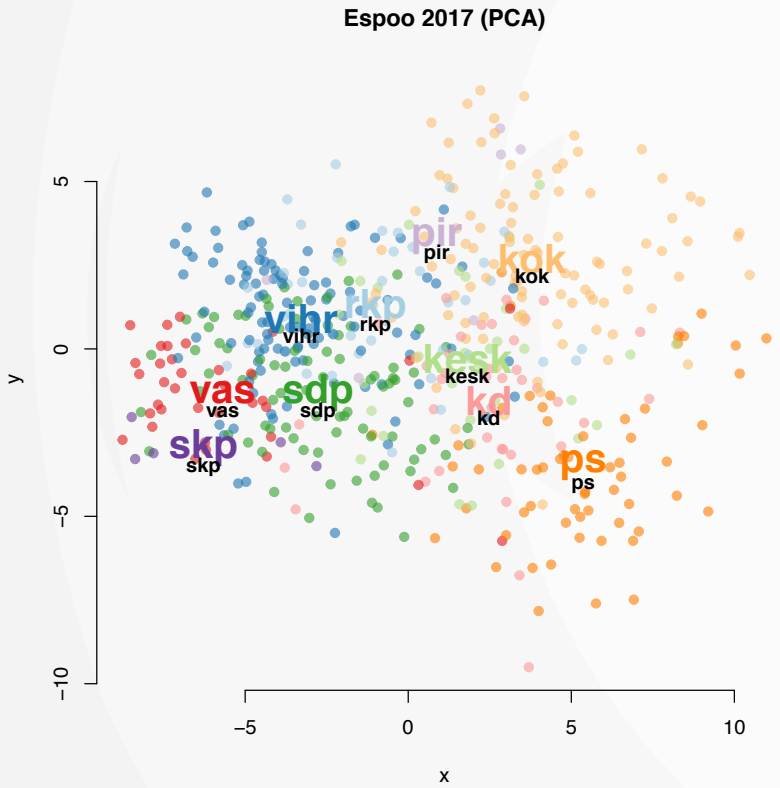


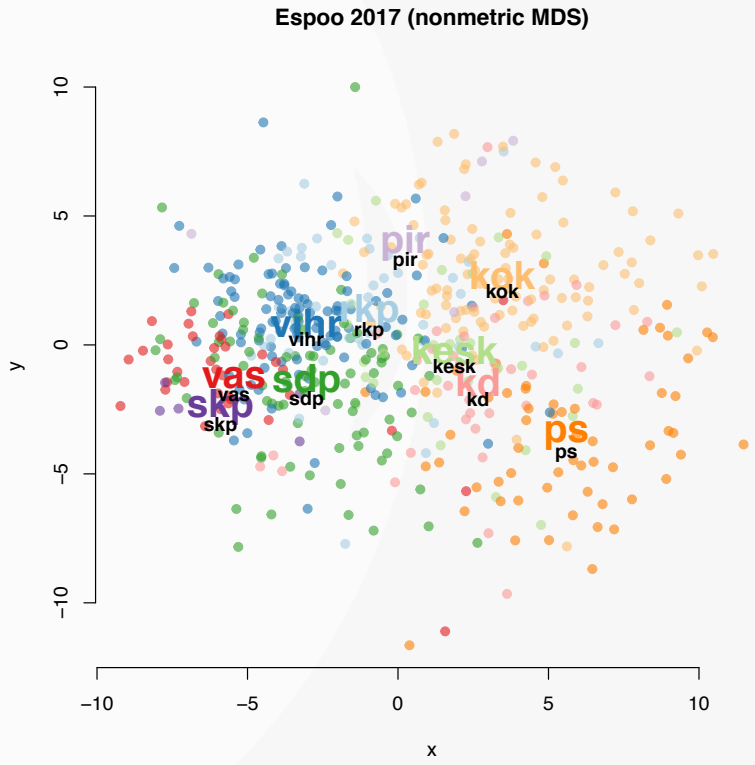
Fig. 9. Top: best ordering to enhance clustering. Bottom: best ordering to enhance correlation [30].

- Algorithmic accuracy vs. perceptual accuracy
- Figures from Bertini et al. 2011

DIMENSIONALITY REDUCTION

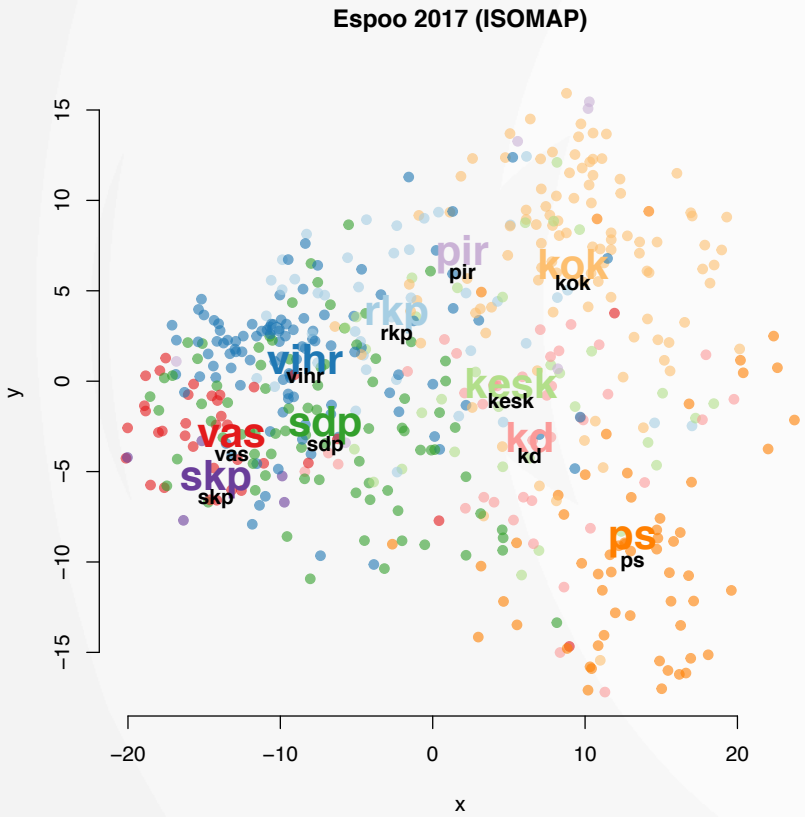


Dimensionality reduction: linear projection pursuit

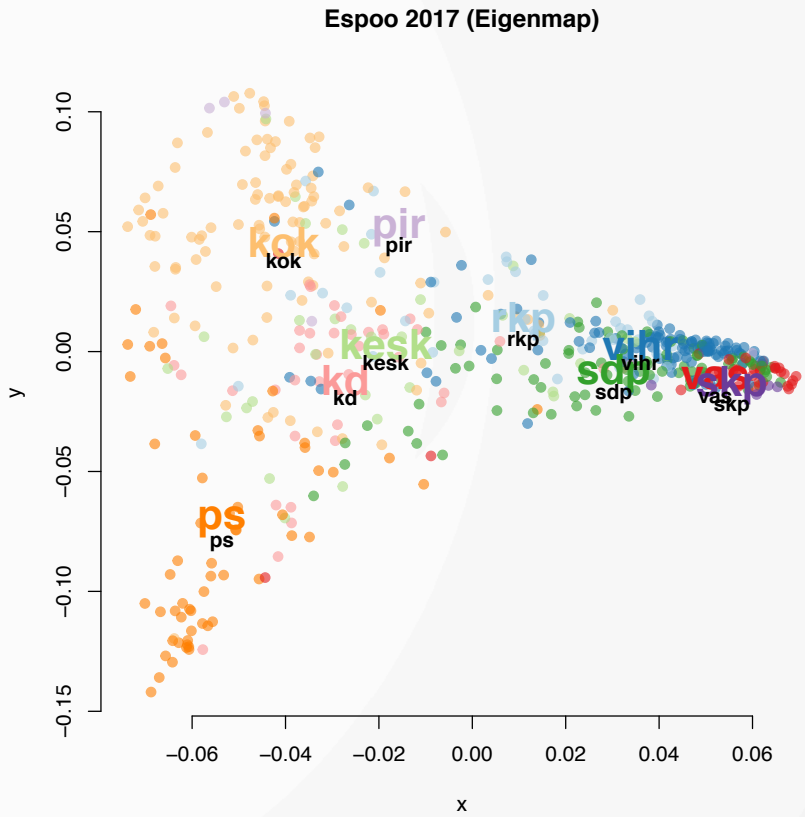


Dimensionality reduction: nonlinear global

DIMENSIONALITY REDUCTION



local manifold



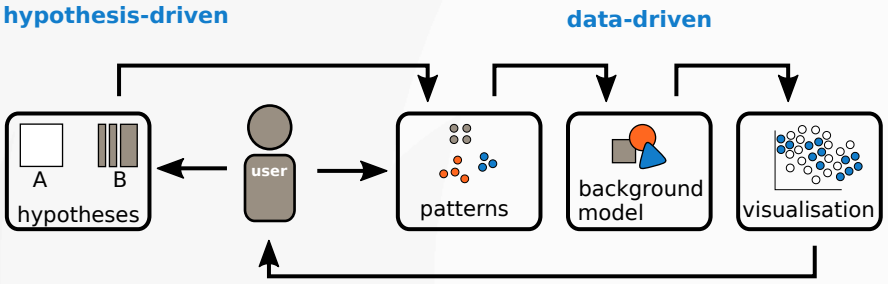
local manifold

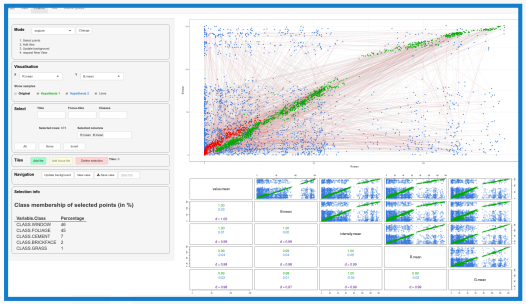
DIMENSIONALITY REDUCTION

“Controllability and interaction are two concepts that are mostly absent from dimensionality reduction."

Verleysen, Lee, 2013

MODELING USER'S KNOWLEDGE AND OBJECTIVES





- User's knowledge: we can model user's knowledge as data distribution
- User's objectives: We can also model the parts of distributions are interesting.
- Dimensionality reduction criteria: show the user the interesting things (wrt. objectives) that he or she does not already know (wrt. knowledge)
- Puolamäki et al. 2018 and references therein. See Sacha et al. 2017 for review of other approaches.

MOST INFORMATIVE VIEWS

- **Tell me something I don't already know:** show the user the largest differences between the background knowledge and the data

[Puolamäki et al. 2018, In Proc ICDE 2018.](#)

https://github.com/edahelsinki/sideR

SOFTWARE TOOLS FOR AI

- One of the major developments during last decades is introduction of software tools to implement AI methods
  - PhD no longer needed to run a complex algorithms (deep learning etc.)
  - PhD may still be needed to understand what happens
- Scientific publications often include open source libraries to implement the methods
- Take-away:
  - use these tools
  - contribute to these tools
  - recognise more the general computational problems and solve them instead of separately. tacking specific problems
- Examples: Keras for deep learning, Stan for probabilistic reasoning etc.

Gravitational Waves Detected 100 Years After Einstein's Prediction

News Release • February 11, 2016

- Probabilistic programming
- Reliable inference from limited data
- Uncertainty quantification for (autonomous) decision-making
- Democratises AI: replaces difficult mathematical derivations with programming tasks
- Stan (10.000+ users) mc-stan.org

FCAI Finnish Center for Artificial Intelligence

REAL AI FOR REAL PEOPLE IN THE REAL WORLD