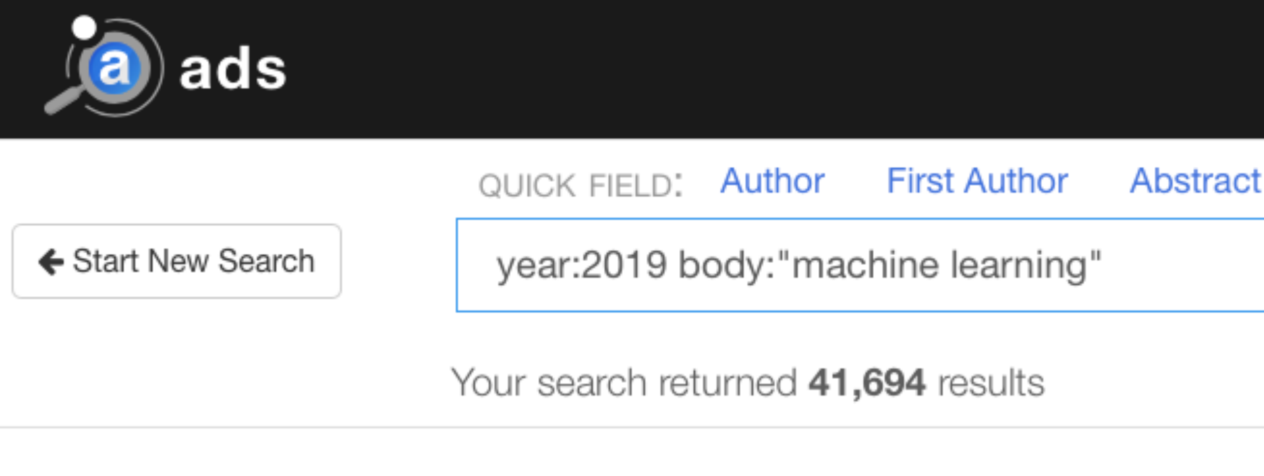
Mario Pasquato

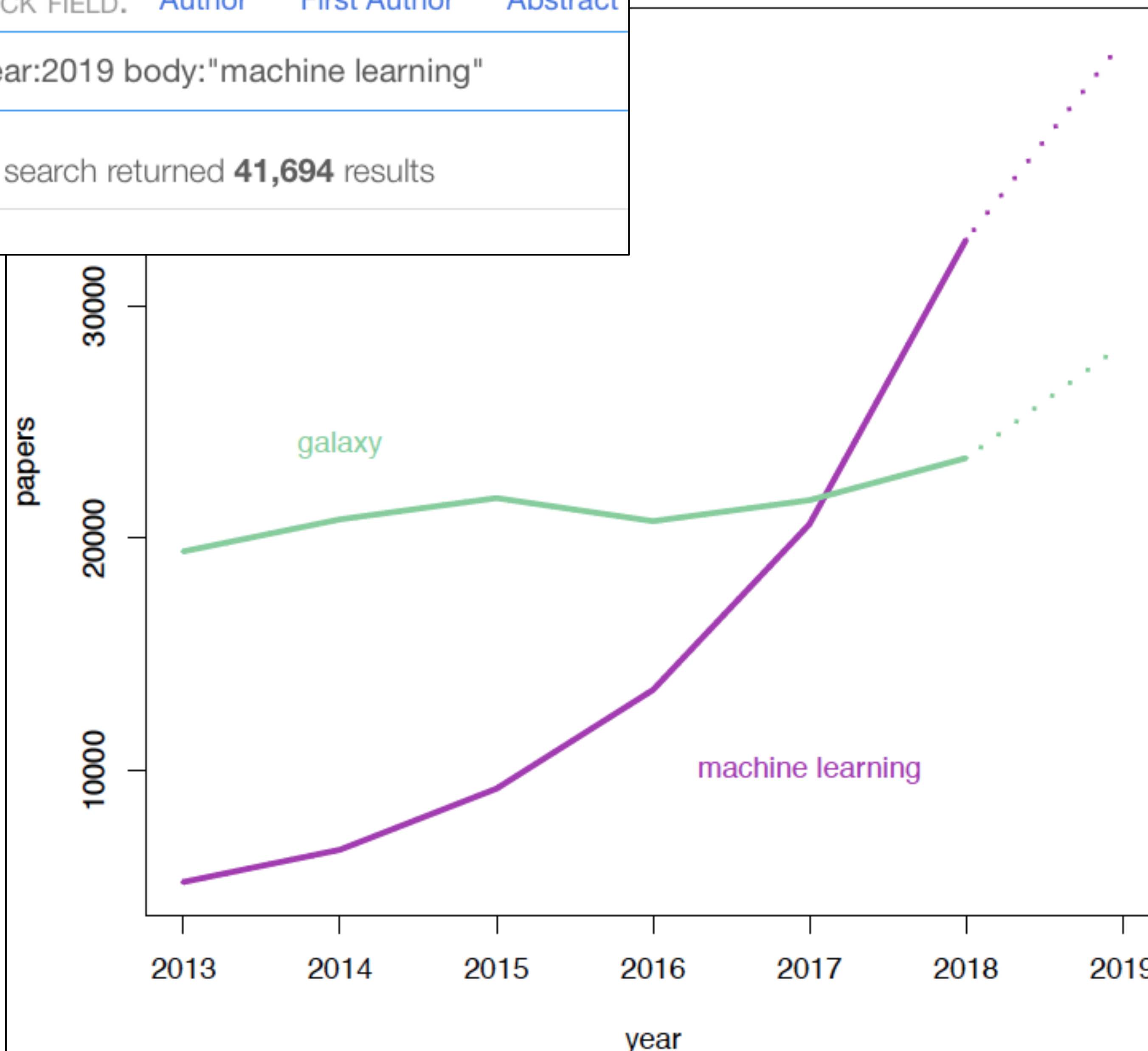
Marie Curie Fellow @ INAF (Padua Observatory) Italy

Supervised and unsupervised machine learning for astronomy: some concrete examples from my research

Collaborators:
Michela Mapelli
Ammar Askar
Abbas Askar
Mirek Giersz
Alessandro Ballone
Piero Trevisan
Sara Lucatello
José Schiappacasse
Santi Cassisi
Antonella Vallenari
Angela Bragaglia
Antonino Milone

Papers on NASA ads as a function of time

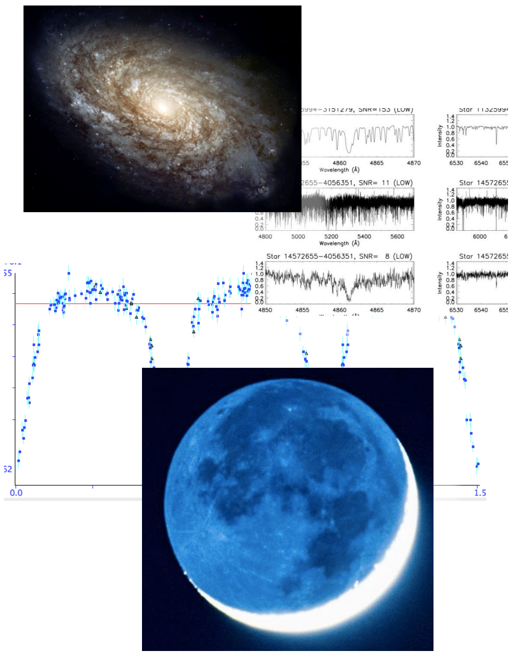




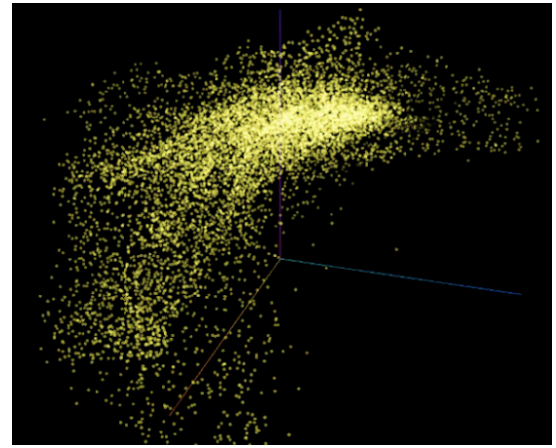
$N \sim 4000\, e^{(y-2012)/3}$

1 paper/minute by 2027
1 paper/second by 2038

ML: what is it?



Astronomical data



Points in some high-dimensional space

Without labels

With labels

Clustering

Dimensionality reduction

Supervised classification or regression

Finding segregated subsystems of stellar-mass black holes in globular clusters

A few BHs in globular clusters were spotted observationally:
M22 (Strader et al. 2012); M62 (Chomiuk et al. 2013);
47 Tuc (Miller-Jones et al. 2015; Bahramian et al. 2017)
M10 (Shishkovsky et al. 2018); NGC 3201 Giesers et al. (2018)

Globular clusters produce thousands of BH:
how many are retained?

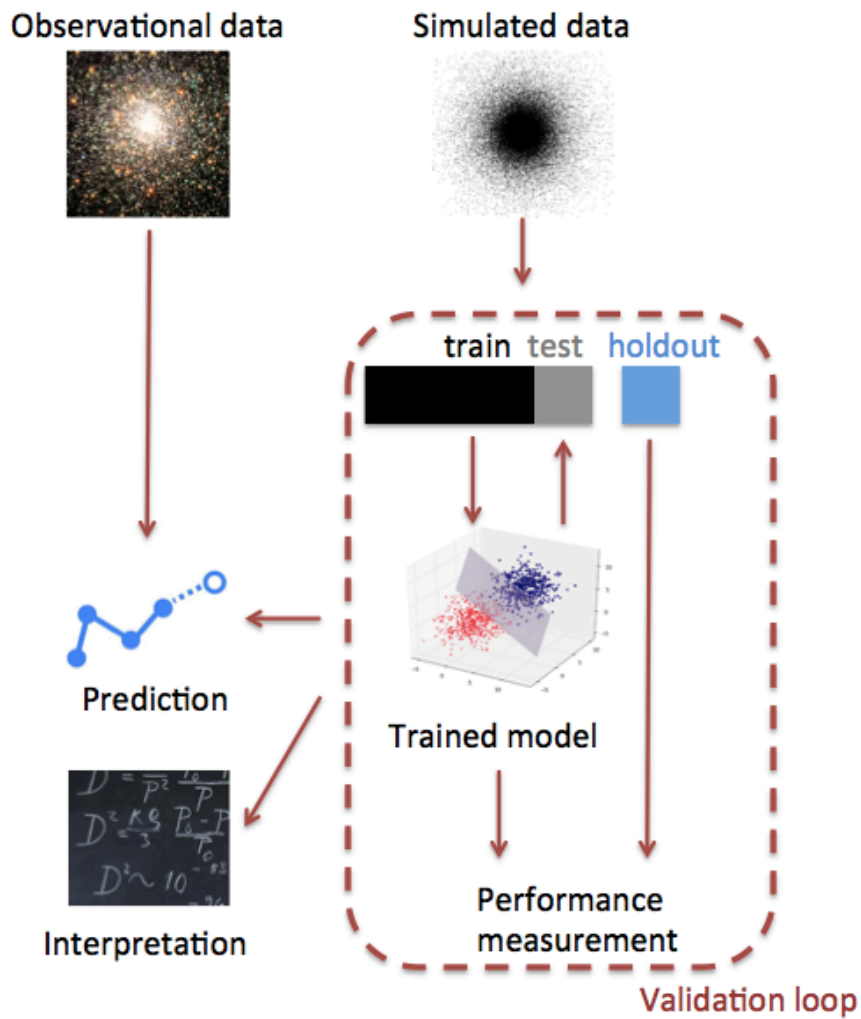
They have **dynamical effects** and may merge and produce gravitational waves

Classification problem with numeric features

- Half-Light Radius
- Central Surface Brightness
- Central Velocity Dispersion
- Total Luminosity
- Relaxation Time
- Core Radius

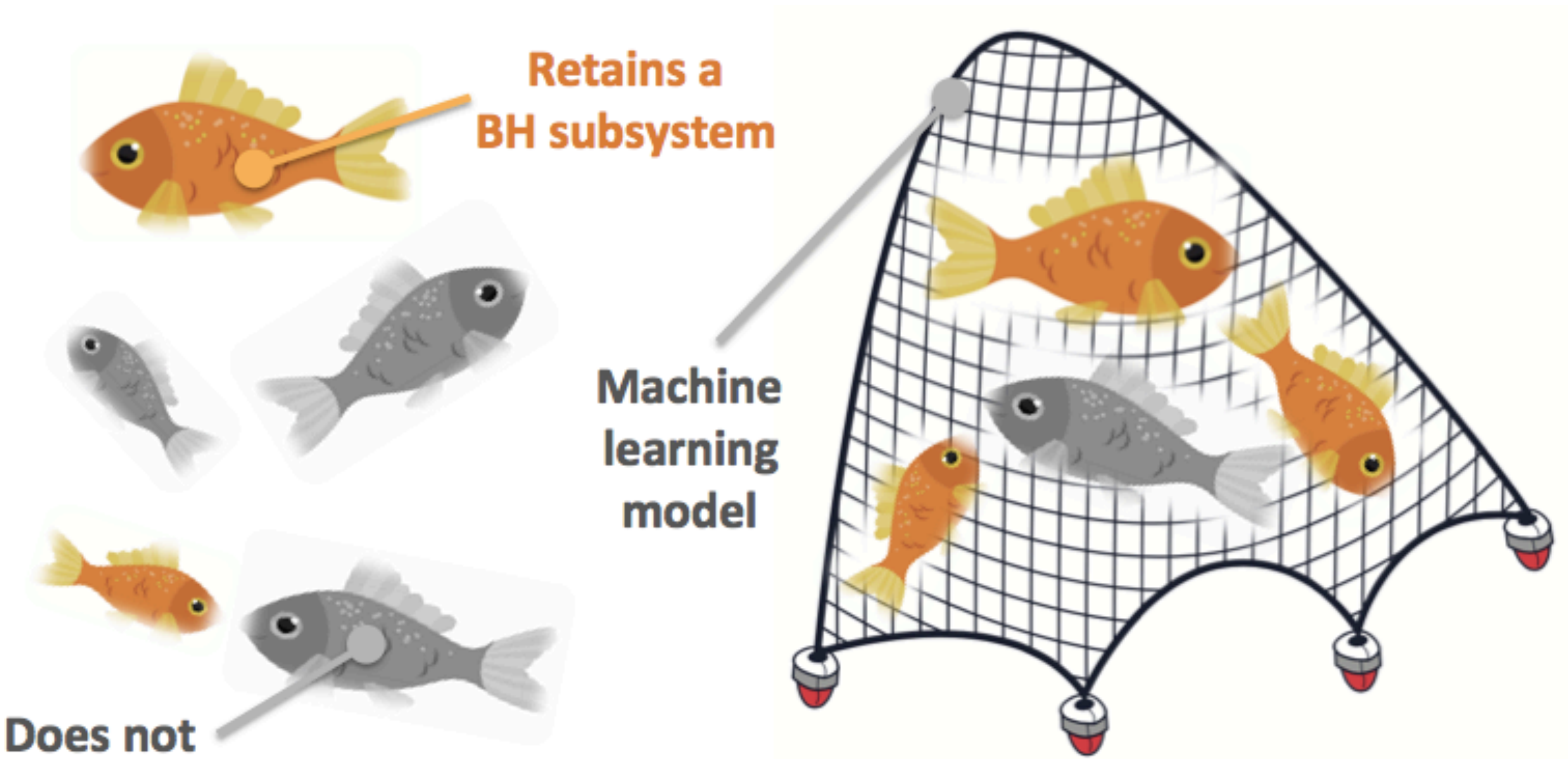
Low-dimensional feature space, observed in actual globular clusters (e.g. Harris 1996), easy to extract from simulations

~ 2000 state-of-the-art dynamical simulations from MOCCA Survey I (Askar et al. 2017)



Askar, Askar, Pasquato, Giersz 2018 MNRAS 485, 5345

Classification problem



Metrics: precision (orange fishes caught / total caught fishes);
recall (orange fishes caught / total orange fishes); F-score harmonic mean of precision and recall

Simple tree models perform well

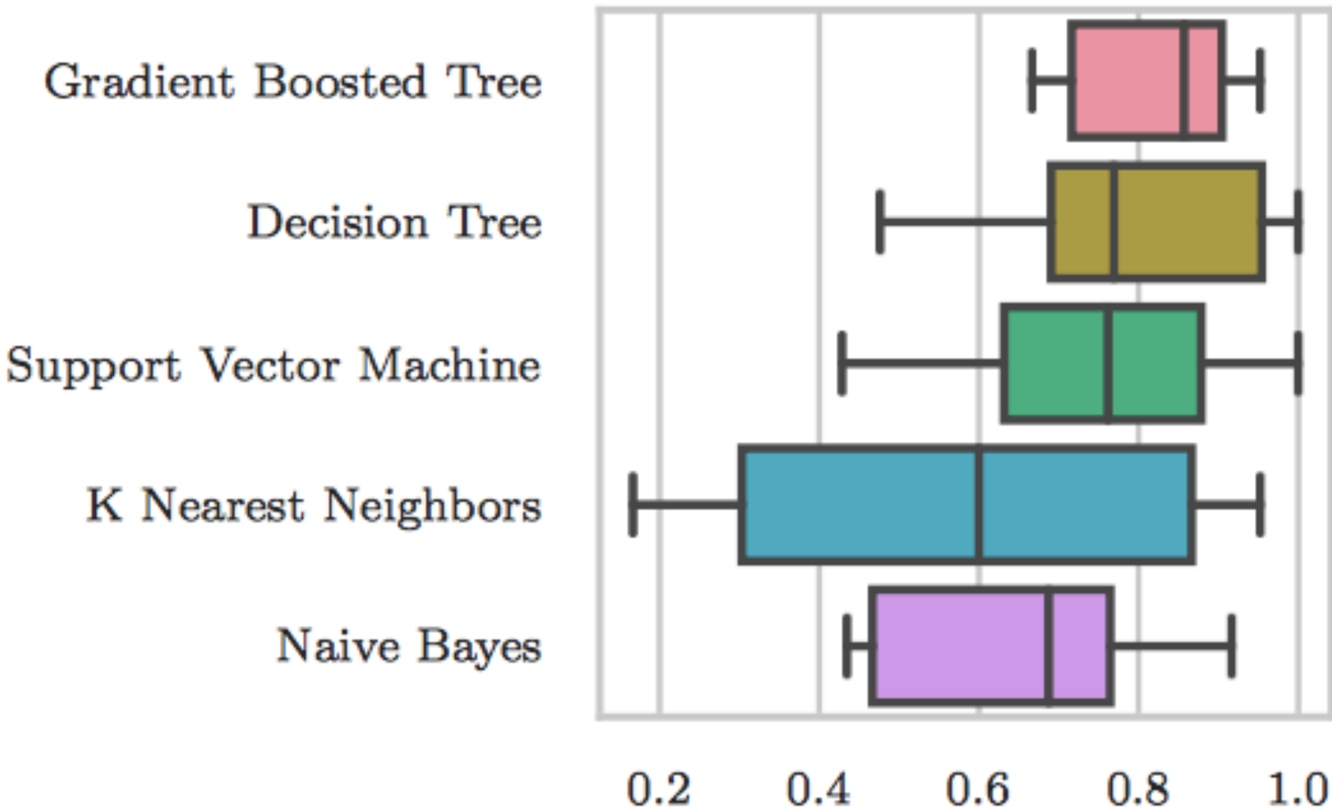
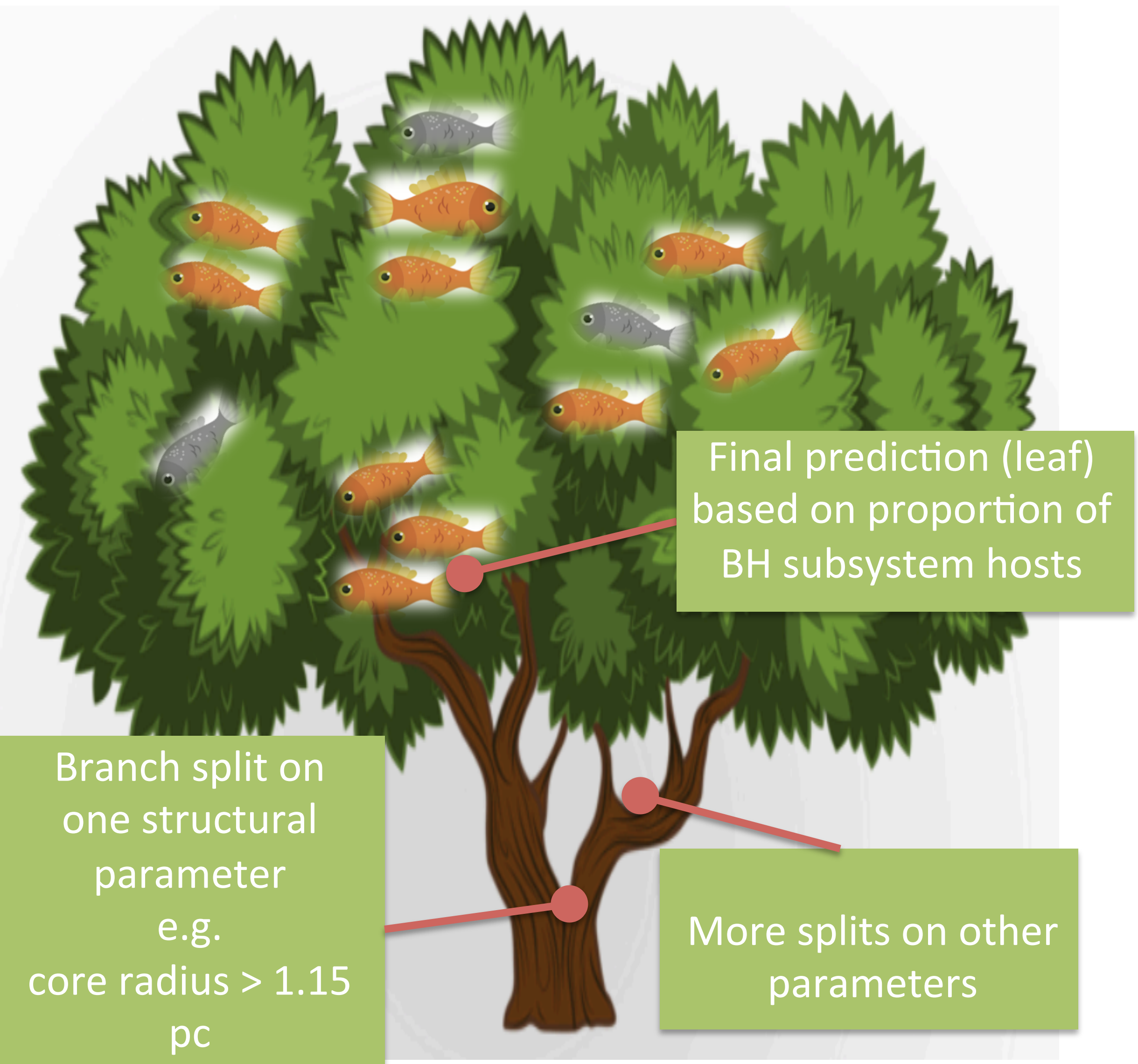
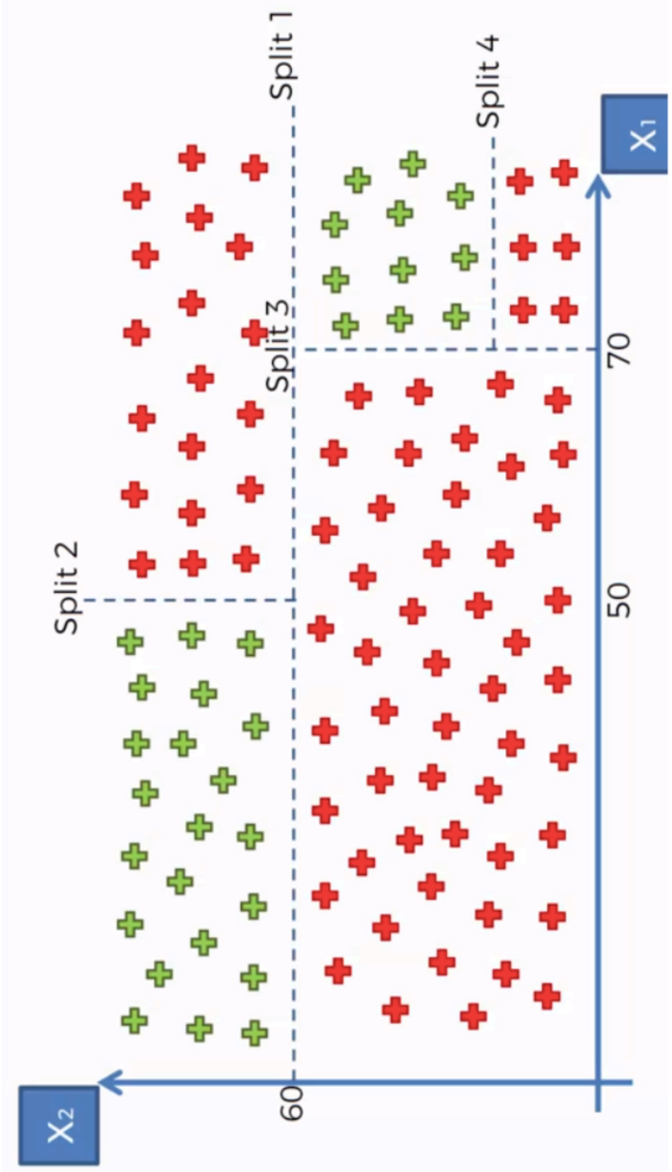


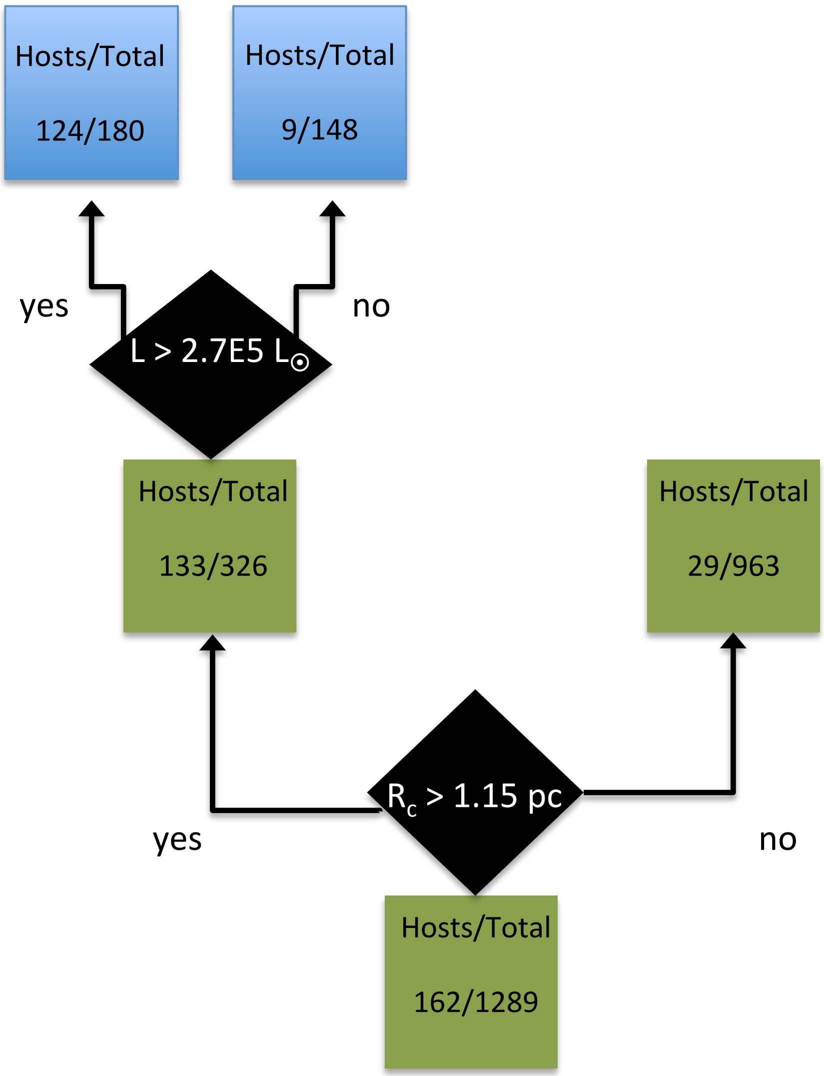
Figure 4. Comparison of each classifiers' f-score with 15-fold stratified CV testing. An f-score of 1.0 is the best possible, 0.0 is the worst. The score is affected by how well the classifier can find all the BH subsystems and whether the identifications are false-positives.

Decision tree model





Physical interpretation



First few branches of the learned tree

First split is on core radius: black hole subsystem hosts have large cores due to dynamical heating

Second split on total luminosity: big clusters produce more black holes, have higher retention due to higher escape velocity

Comparison with real GCs

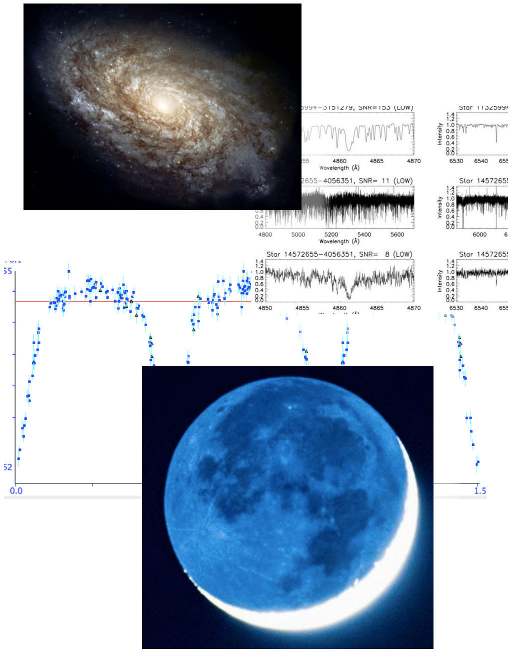
Table 4. Predictions from the Harris (1996, *updated 2010*) and Baumgardt & Hilker (2018) datasets using the gradient boosted decision tree classifier. Entires where BHS presence was classified positively are shown. The *BHS* column represents the classifier trained on all simulation data whereas *Fallback* represents training on models where mass fallback was enabled and BH natal kicks were lower.

Green row = predicted BH subsystem host by all models e.g. NGC 288, M10

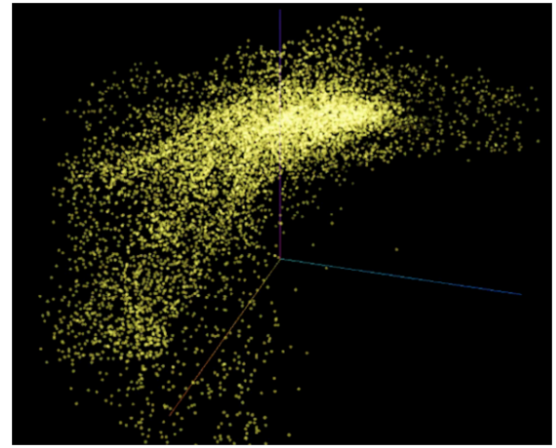
Results compare well with other methods (Askar et al. 2019 marked with *)

Cluster Name	BHS (Harris)	Fallback (Harris)	BHS (B&H)	Fallback (B&H)
IC 4499 *	✓	✓	✓	✓
NGC 288 *	✓	✓	✓	✓
NGC 3201 *	✓	✓	✗	✓
NGC 4372 *†	✓	✓	✓	✓
NGC 4590 (M68)	✗	✓	✗	✗
NGC 4833 *†	✗	✓	✓	✓
NGC 5139 (ω Cen)	✓	✓	✗	✓
NGC 5272 (M3) *	✗	✓	✓	✓
NGC 5286	✗	✓	✗	✓
NGC 5466 *	✗	✓	✗	✗
NGC 5897 *†	✓	✓	✗	✓
NGC 5904 (M5)	✗	✓	✓	✓
NGC 5927	✗	✗	✓	✓
NGC 5986 *†	✓	✓	✗	✓
NGC 6101 *†	✓	✓	✗	✗
NGC 6139 †	✓	✓	✗	✗
NGC 6144 *†	✗	✓	✓	✓
NGC 6205 (M13) *	✗	✓	✓	✓
NGC 6218 (M12)	✓	✓	✗	✗
NGC 6254 (M10)	✓	✓	✓	✓
NGC 6266 (M62)	✗	✗	✓	✗
NGC 6273 (M19) †	✗	✓	✓	✓
NGC 6287 †	✓	✓	✗	✗
NGC 6304 †	✗	✓	✓	✓
NGC 6316 †	✓	✓	✗	✗
NGC 6333 (M9) †	✓	✓	✗	✗
NGC 6356 †	✗	✓	✗	✓
NGC 6362 *	✗	✓	✗	✓
NGC 6380 †	✓	✓	✗	✗
NGC 6388	✗	✗	✓	✗
NGC 6401 *	✓	✓	✗	✗
NGC 6402 (M14) †	✓	✓	✓	✓
NGC 6426 *†	✗	✓	✗	✗
NGC 6440 †	✓	✓	✗	✗
NGC 6496 *†	✗	✓	✗	✗
NGC 6517 †	✗	✓	✗	✗
NGC 6539 (GCL 85)	✓	✓	✗	✗
NGC 6553	✓	✓	✗	✗
NGC 6569 *†	✓	✓	✓	✓
NGC 6584 *†	✓	✓	✗	✗
NGC 6656 (M22) *	✓	✓	✓	✓
NGC 6712 *	✗	✓	✓	✓
NGC 6723 *†	✓	✓	✓	✓
NGC 6760 †	✓	✓	✗	✗
NGC 6779 (M56) *	✗	✓	✓	✓
NGC 6809 (M55) *	✗	✓	✗	✓
NGC 6934 *	✓	✓	✗	✗
NGC 6981 (M72) *	✗	✓	✗	✗
NGC 7078 (M15)	✗	✗	✓	✓
NGC7089 (M2)	✗	✓	✗	✓
Pal11 *†	✓	✓	✓	✓
Terzan5 †	✗	✓	✗	✗

Now some unsupervised ML



Astronomical data



Points in some high-dimensional space

With labels

Supervised classification or regression

Without labels

Clustering

Dimensionality reduction

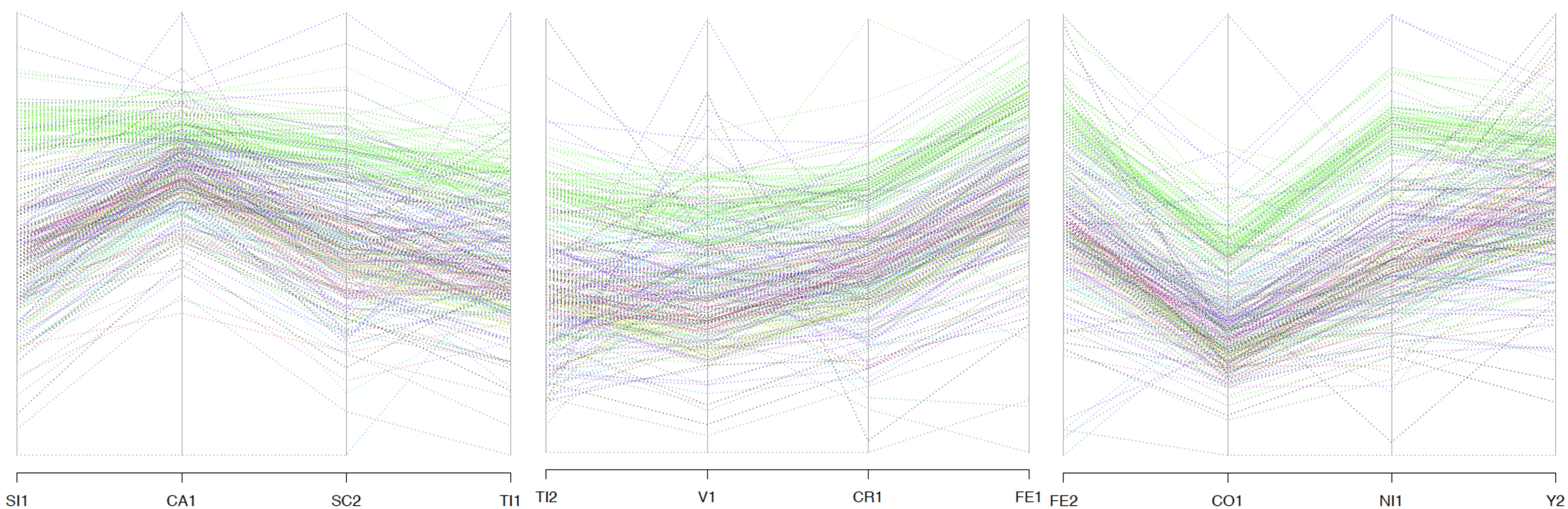
What can we do without labels

- Reduce data-points to fewer representative data-points (finding groups in data; clustering)
  - does k-means ring a bell?
- Reduce coordinates/dimensions to fewer representative coordinates (dimensionality reduction)
  - You all know PCA, right? nonlinear version of it

reduce rows VS reduce columns

Example: chemical tagging

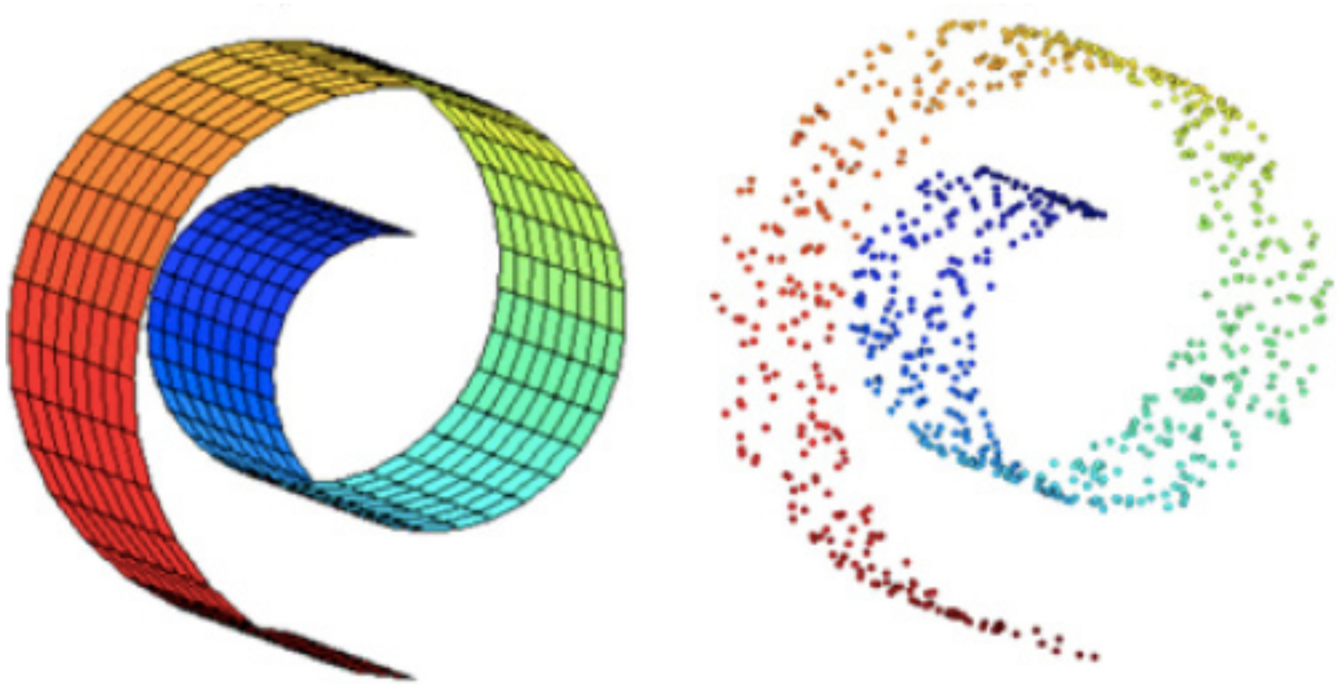
- Field stars may come from disrupted open clusters
- If each OC had distinctive chemistry, we can use abundances as a fingerprint of the parent cluster



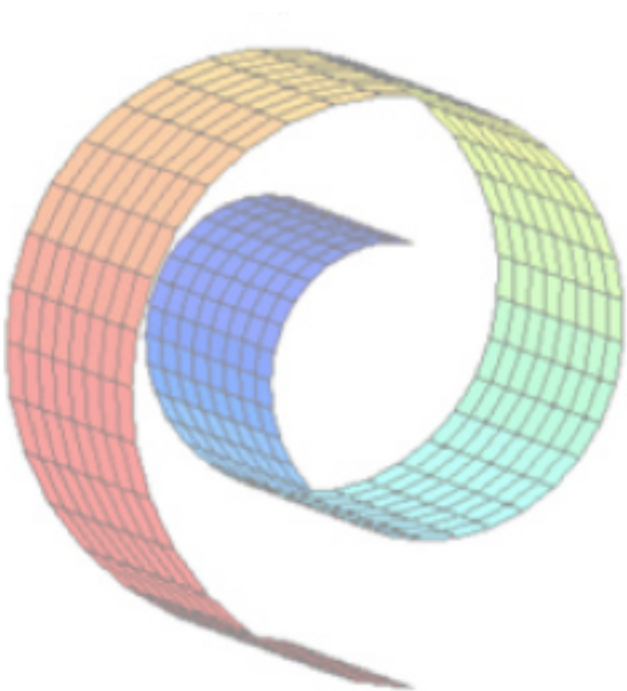
Parallel coordinate display of 12 abundances of 330 main sequence stars from GAIA-ESO (Gilmore et al. 2012). Stars selected in real open cluster fields (courtesy A. Bragaglia), parent open cluster color-coded; can we recover them?

12D -> 2D

- If we could represent this 12D space on a plane...
- PCA and keep the first two components? Nah...



- If we could represent this 12D space on a plane...
- PCA and keep the first two components? Nah...



t-distributed stochastic
neighbor embedding

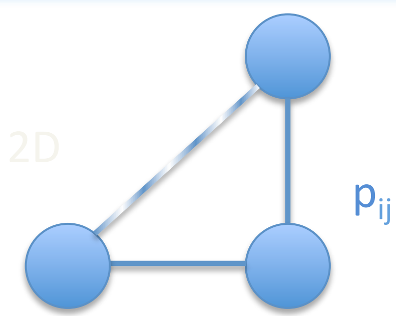
t-SNE

- Local (short) distances matter more than long distances
- Place points in the plane to minimize a loss function that keeps nearby points nearby

$$p_{j|i} = \frac{\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|\mathbf{x}_i - \mathbf{x}_k\|^2 / 2\sigma_i^2)}$$

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}$$

Similarity in high dimension

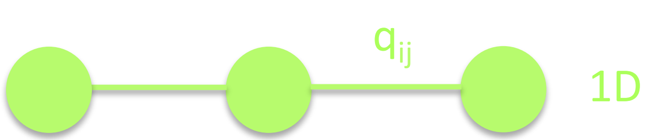


$$q_{ij} = \frac{(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2)^{-1}}{\sum_{k \neq l}(1 + \|\mathbf{y}_k - \mathbf{y}_l\|^2)^{-1}}$$

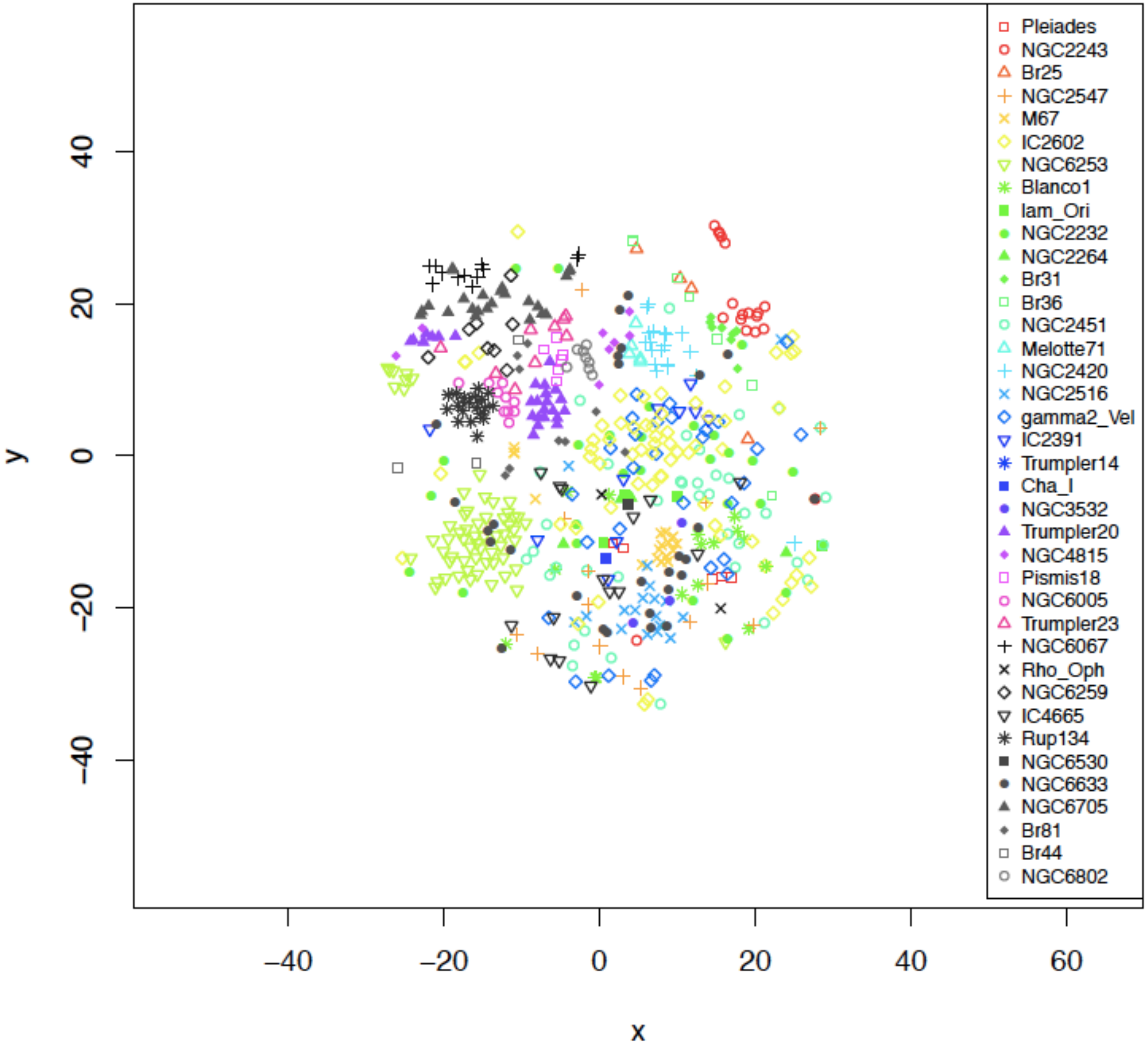
Similarity in low dimension

$$KL(P||Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

Small p_{ij} (faraway points) do not affect the loss



Dimensionality reduction



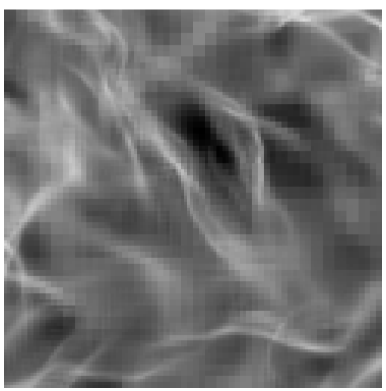
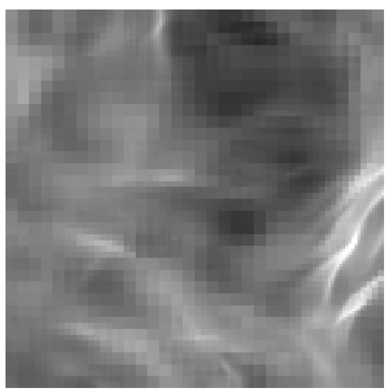
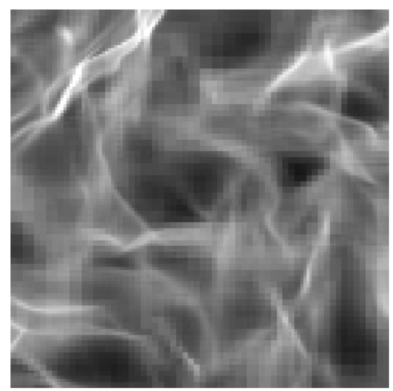
12 dimensional space represented on the plane with t-SNE

Learning the spectral index of turbulence of simulated molecular clouds from projected density maps

- Turbulence in molecular clouds modulates star formation, physics still not fully understood (Elmegreen & Scalo 2004, Hennebelle & Falgarone 2012)
- Velocity power spectrum of turbulence can be measured directly through e.g. line-of-sight velocity (Koch 2019)

Question

- Can we measure the turbulence index of simulated turbulent gas from density maps?
- In particular discriminate between Kolmogorov $P_v(k) = k^{-11/3}$ and Burgers $P_v(k) = k^{-4}$ spectra

Simulations

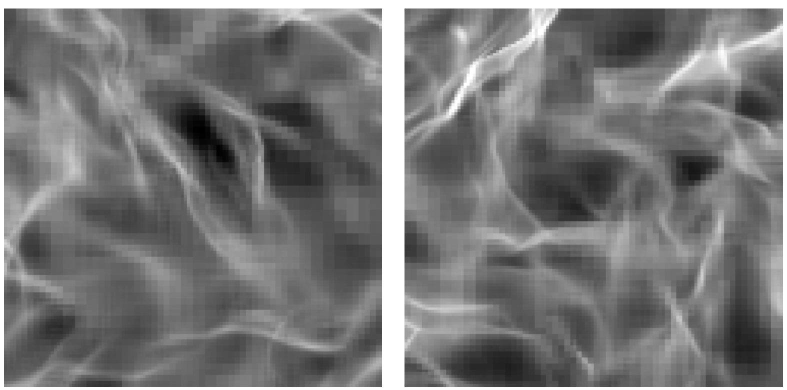
- 1000 simulations of turbulent gas with RAMSES2 [Teyssier 2002] AMR code
- 10x10x10 pc box, initially uniform density gas (6.77×10^{-22}g/cm^3), total mass of $10^4 M_{sun}$.
- Gas kept isothermal at temperature T=10K
- Injected a divergence free, turbulent, supersonic (Mach 1.41) velocity field with spectrum index n=11/3 or 4
- Evolved for 0.5 Myr, solving Euler's equation with a Lax-Friedrichs Riemann Solver, periodic boundaries without self-gravity and magnetic fields

Train/test/holdout split

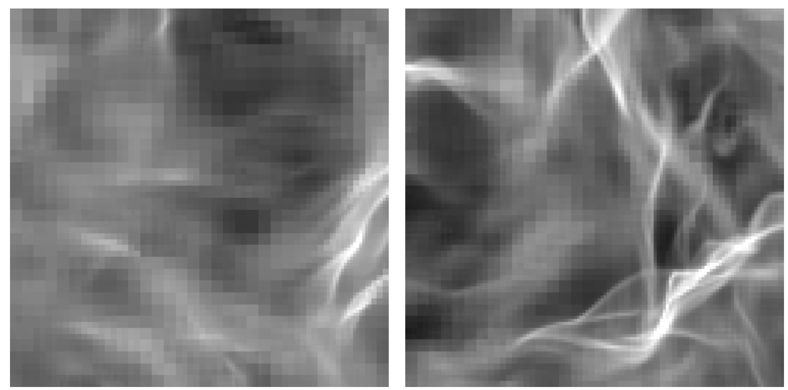
- 500 sims w. Kolmogorov index, 500 w. Burgers
- 400+400 build the train set -> 3 projections (x,y,z) X 4 flip/flop X 4-way cut = 38400 training images
- 50+50 in the test set = 4800 test images
- 50+50 never looked at (holdout set) = 4800 images

Images

- 250x250 pixels, grayscale; each image corresponds to ¼ of the box, seen in projection along an axis (x,y,z)
- Luminosity encodes log column density



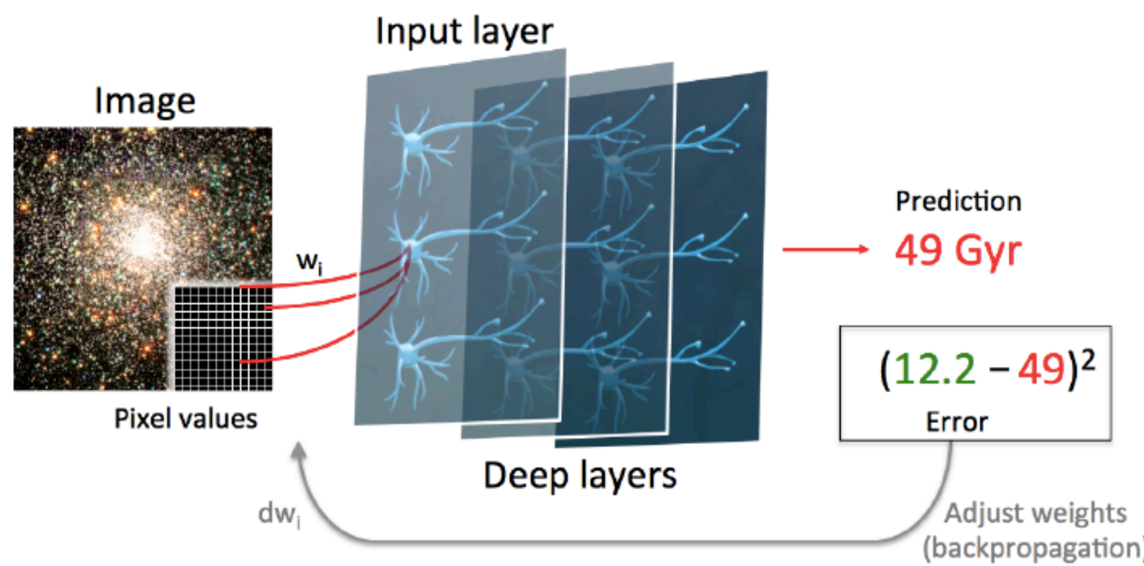
Kolmogorov



Burgers

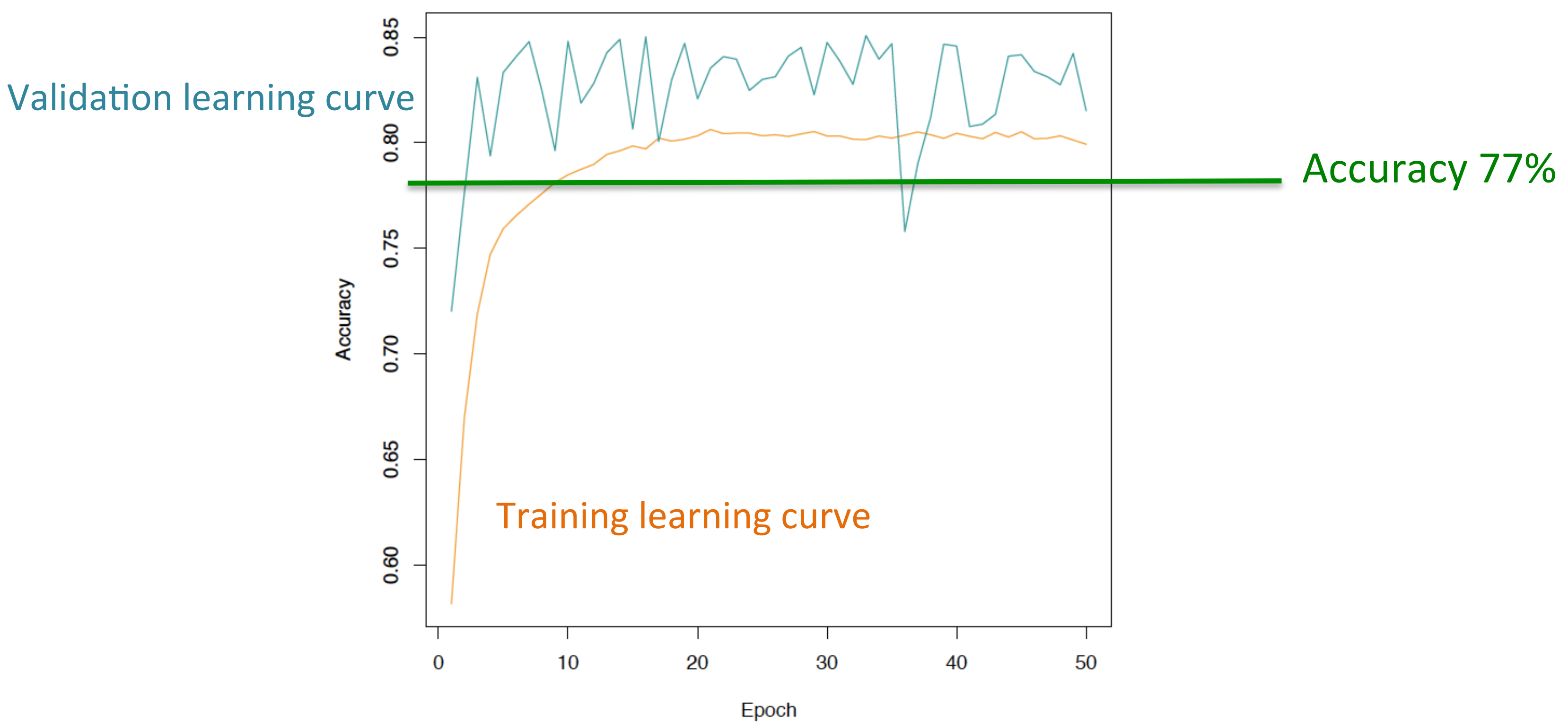
Deep learning setup

- Keras on top of Tensorflow on workstation with a Titan V GPU
- Four convolutional layers (with max pooling) + three dense layers, RELU act.
- Dropout regularization

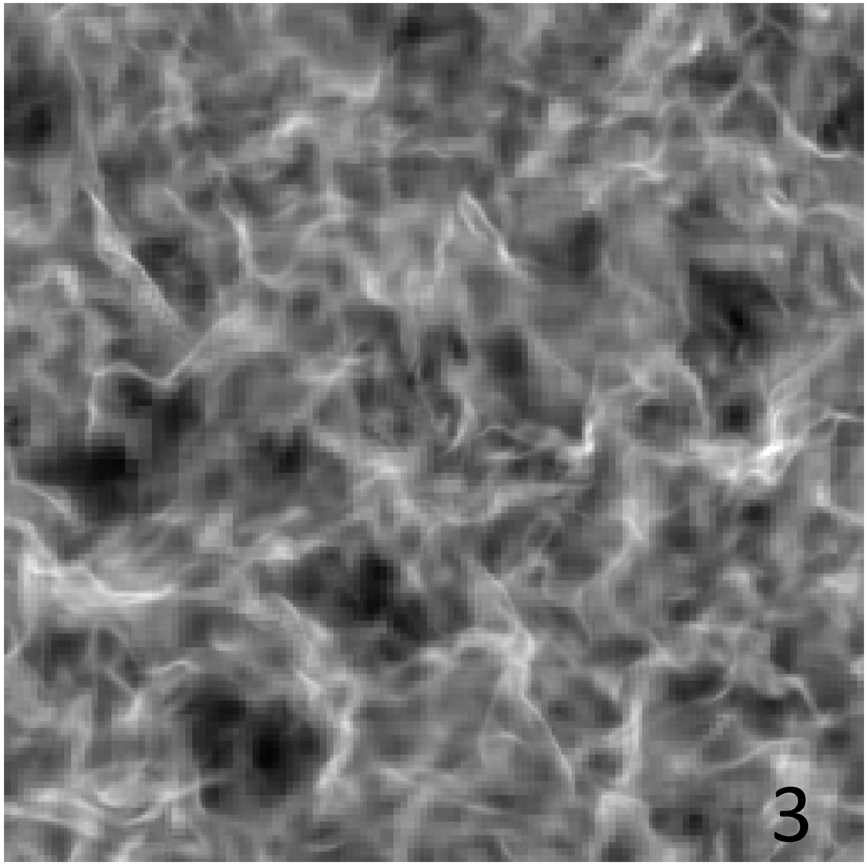


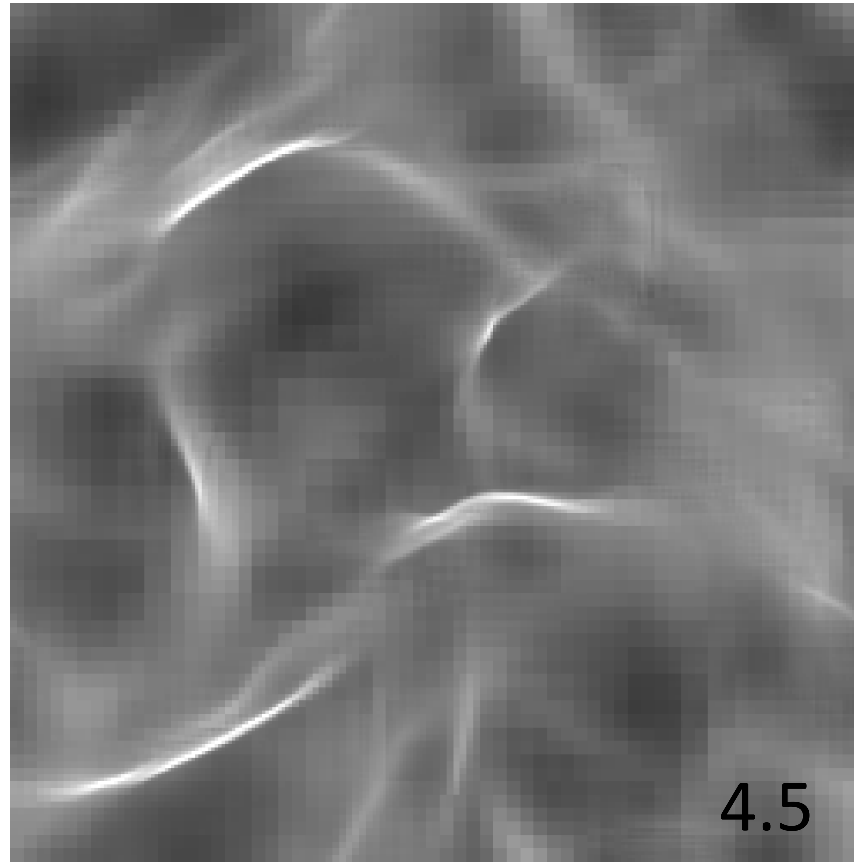
Performance on holdout set

	Predicted Kolmogorov	Predicted Burgers
Kolmogorov	2113	287
Burgers	812	1588



Testing on different indices



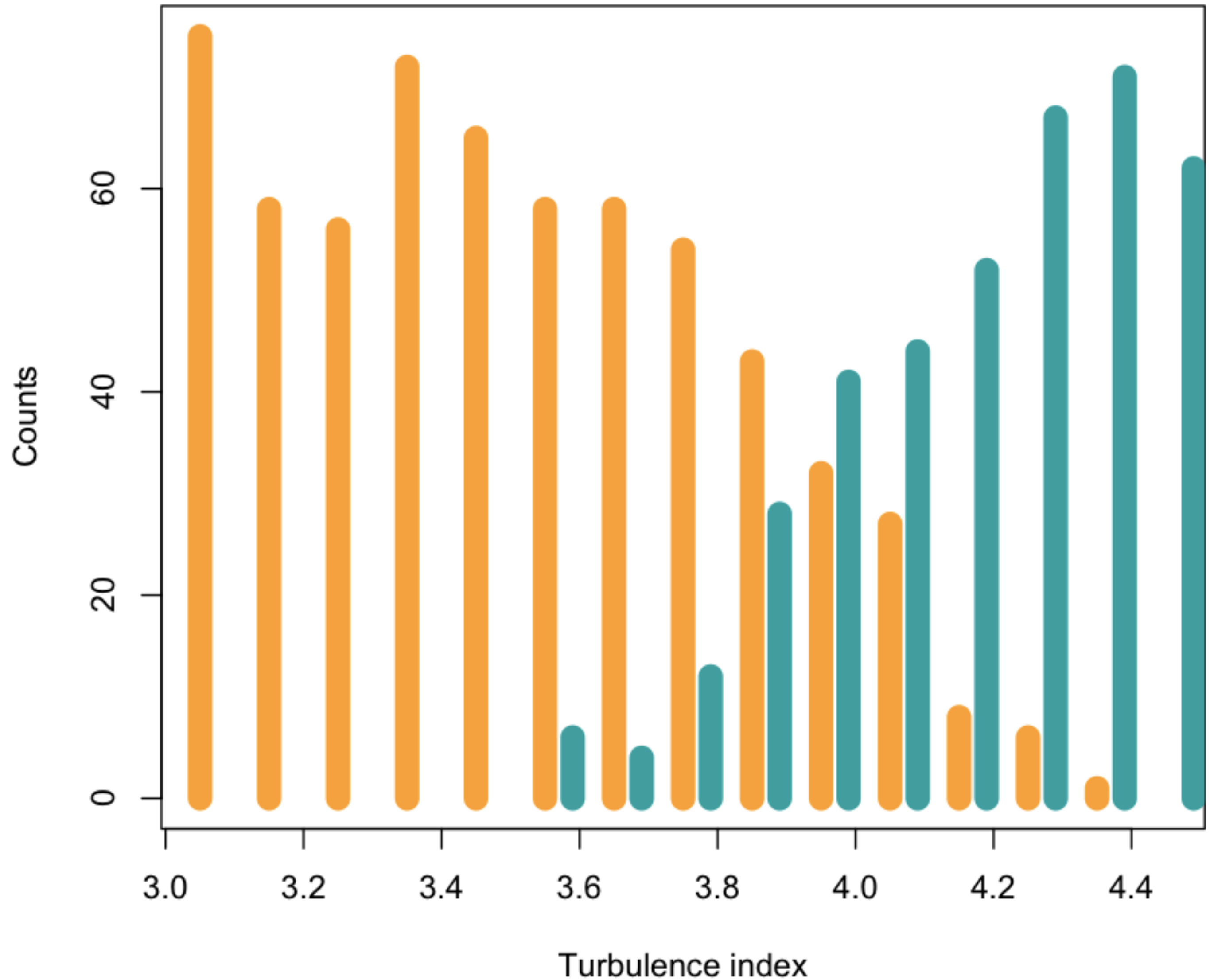


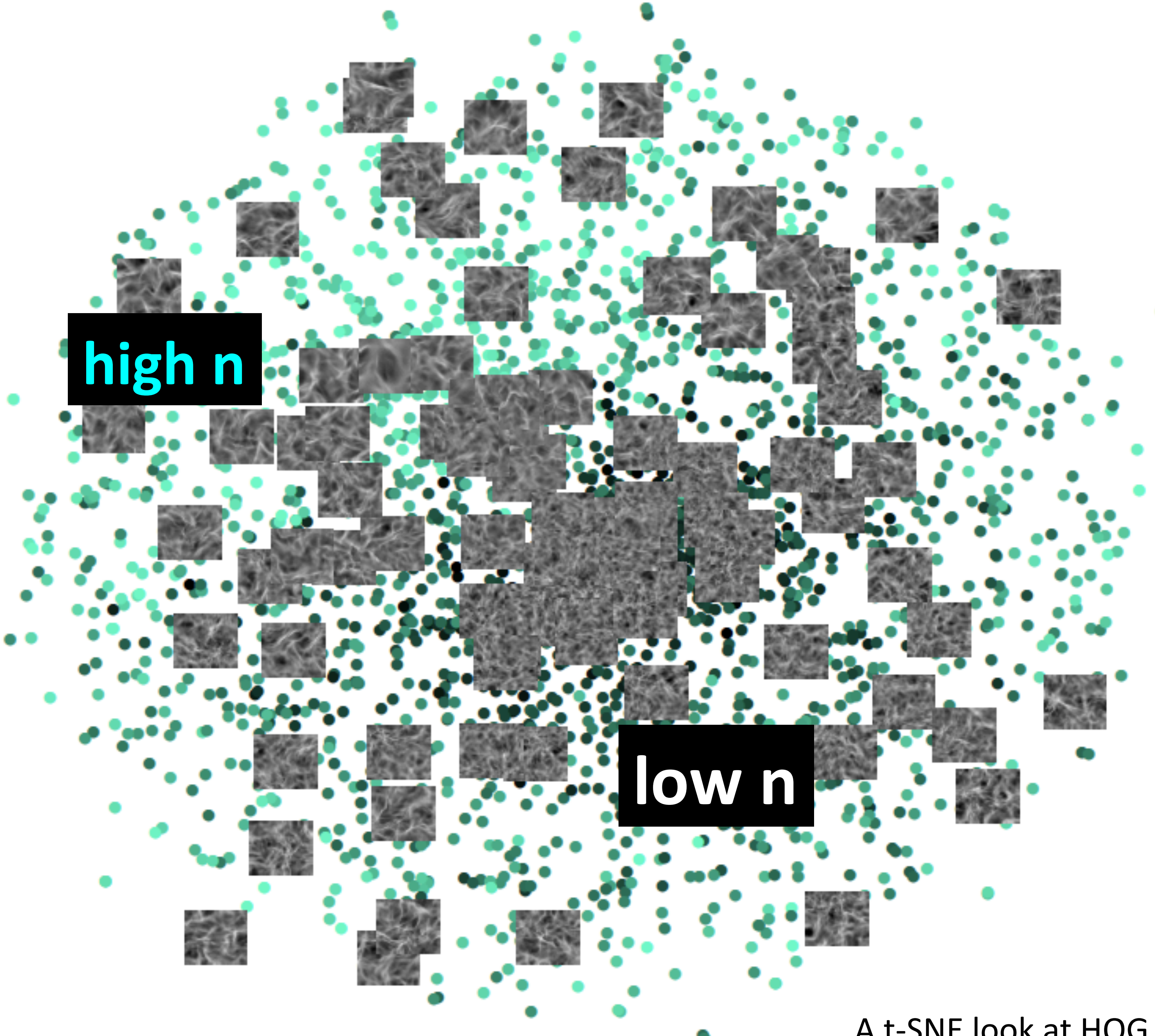
We ran 1000 more simulations with turbulence index that ranges continously from 3 (left) to 4.5 (right). What will the net predict?

Predictions

Predicted Kolmogorov

Predicted Burgers

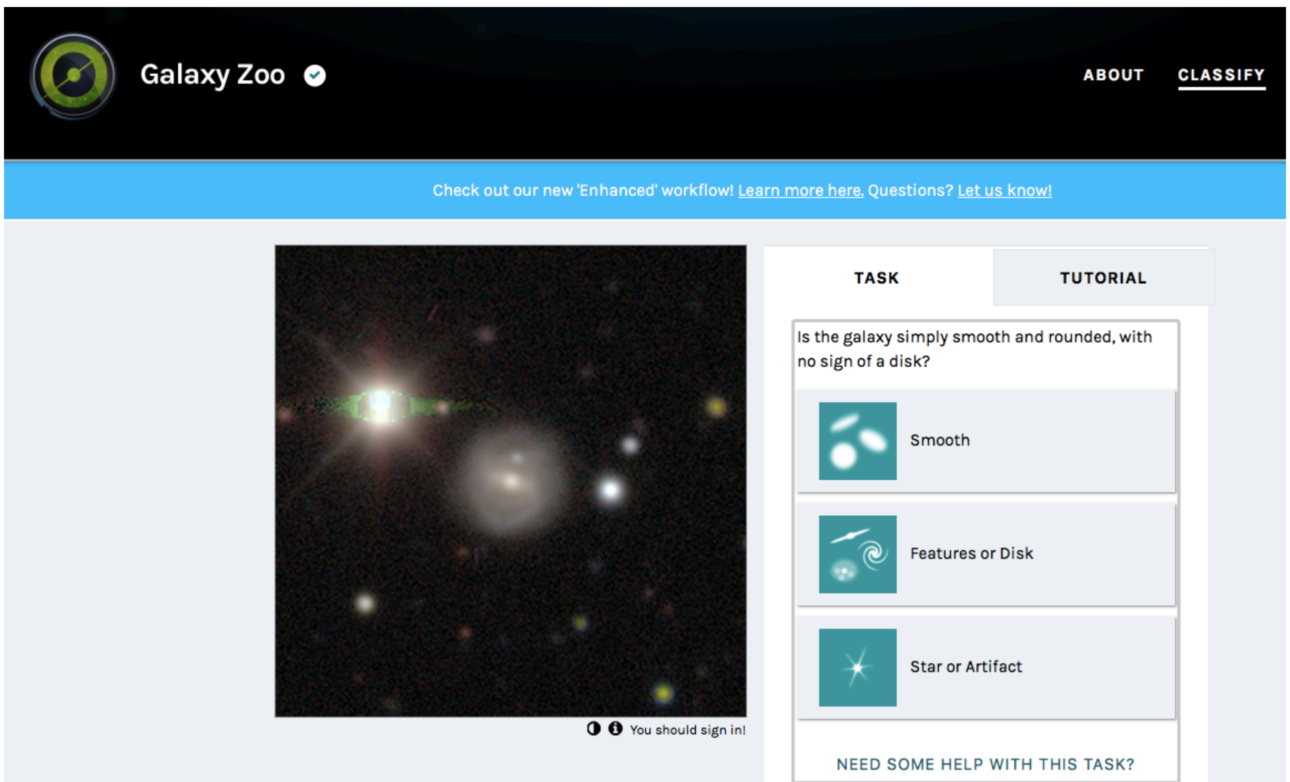




A t-SNE look at HOG features

Back to classification: expensive labels

- Human-labeled data + training → classification



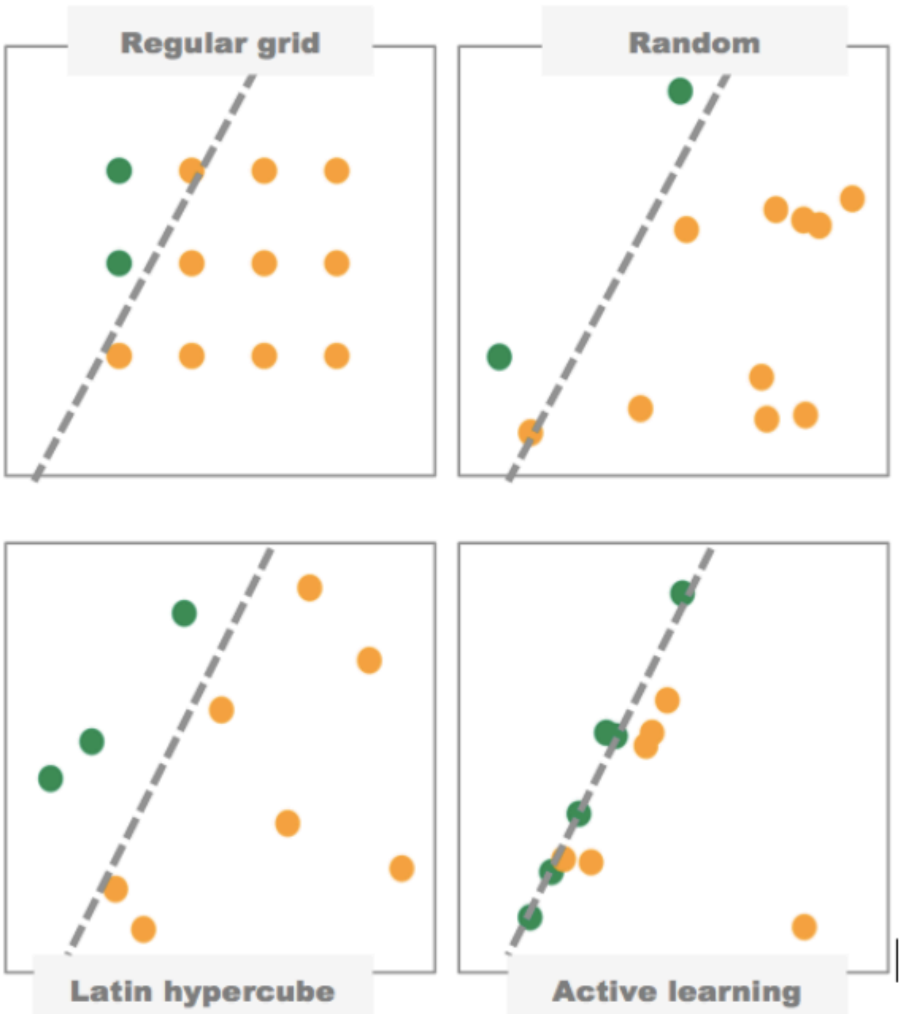
Example: galaxy zoo / zooniverse:

morphological classification of galaxies

Other example: finding jellyfish galaxies in TNG Yun et al. 2019, Yun et al in prep.

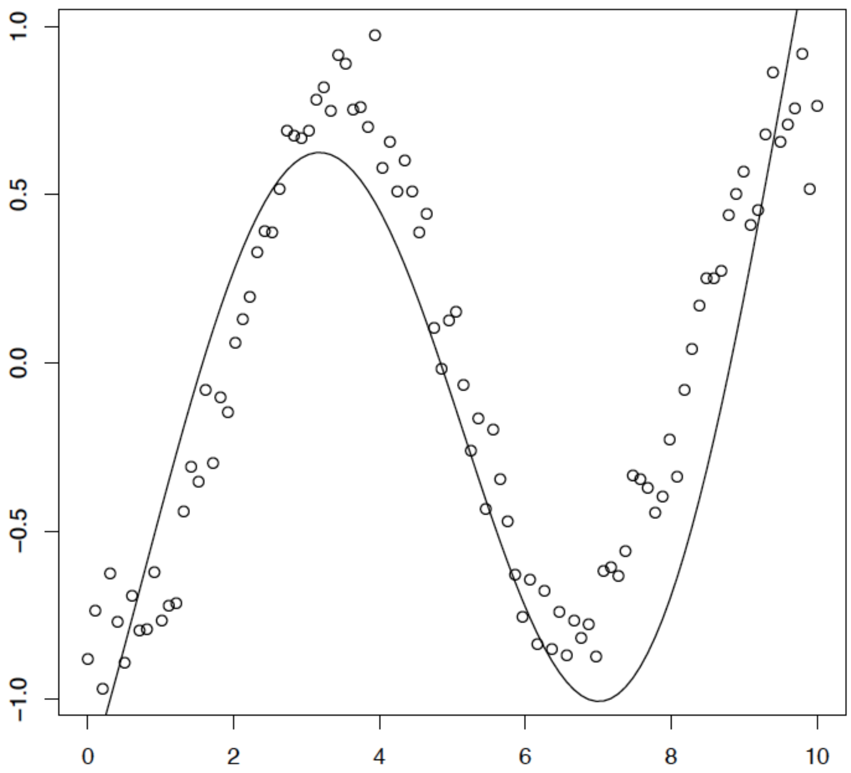
Few labels, costly to label data

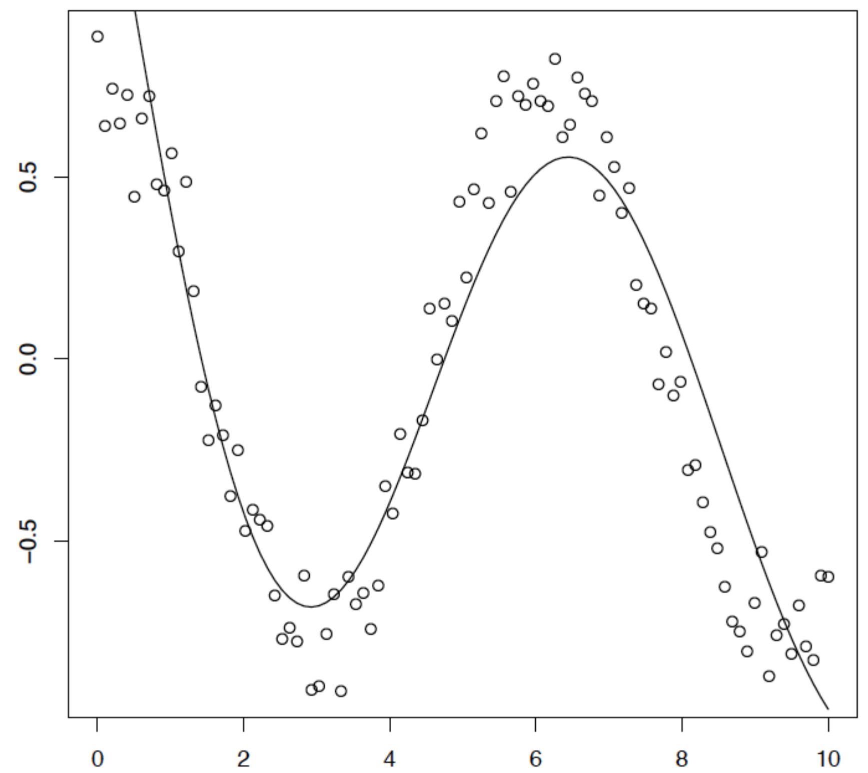
- Active learning: the program chooses which data to learn from
- So you need less hand-labeled data
- Data is labeled where most needed



An active learning toy example

- A particle moves in a 1-D potential + noise
- Initial conditions x_0, x_0', evolved for time t
- Train a ML model to predict $x = x(x_0, x_0', t)$



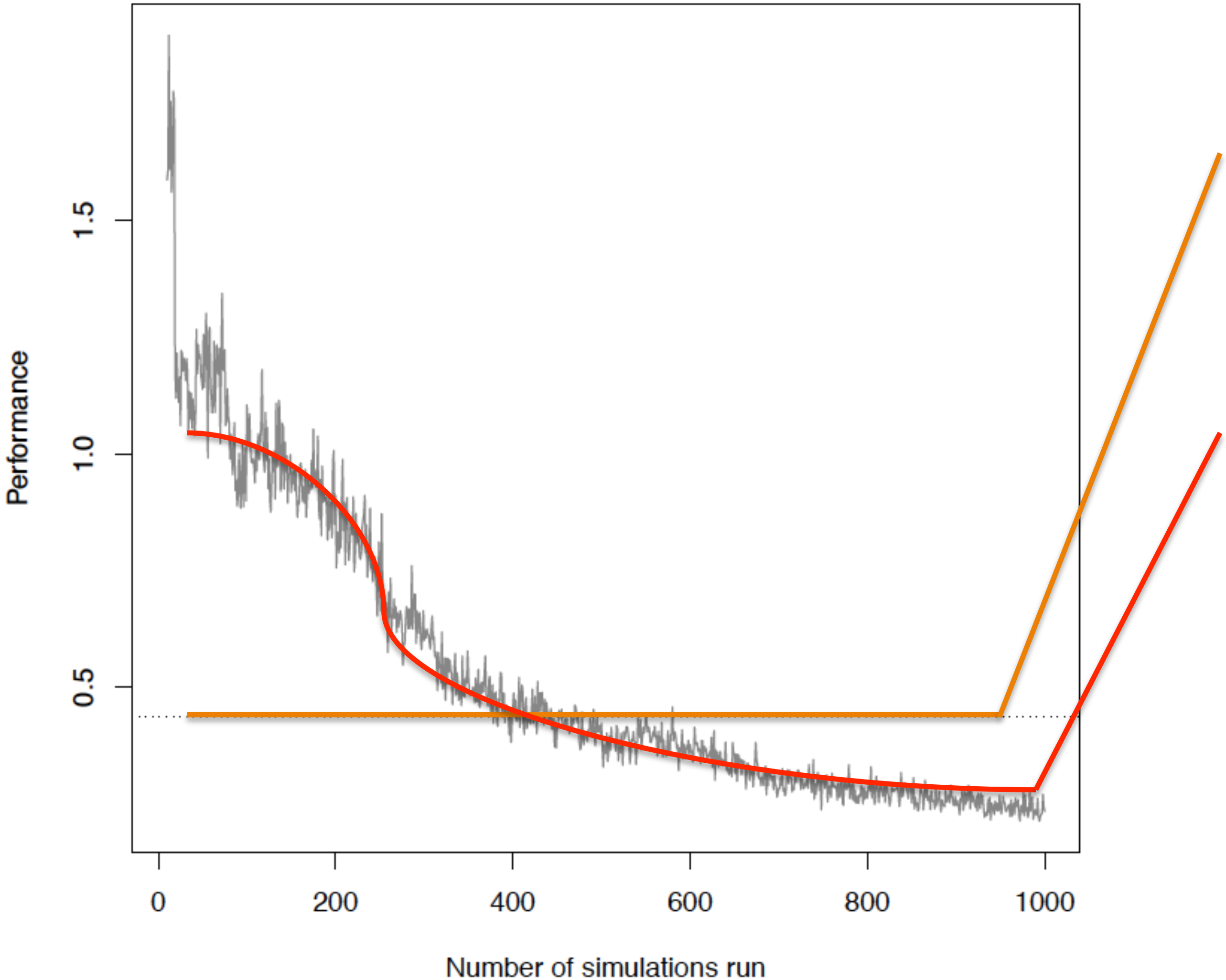


The model learns `physics'

- The model is trained only on couples

  (x_0, x_0', t) ; x and never gets any direct information on the potential
- Why do this? We already have enough students to teach physics to...
- The point is: which model will learn faster (i.e. require less training data):
  - one that picks its own samples (active learner)
  - one that trains on random samples (passive learner)

Active learning VS passive learning



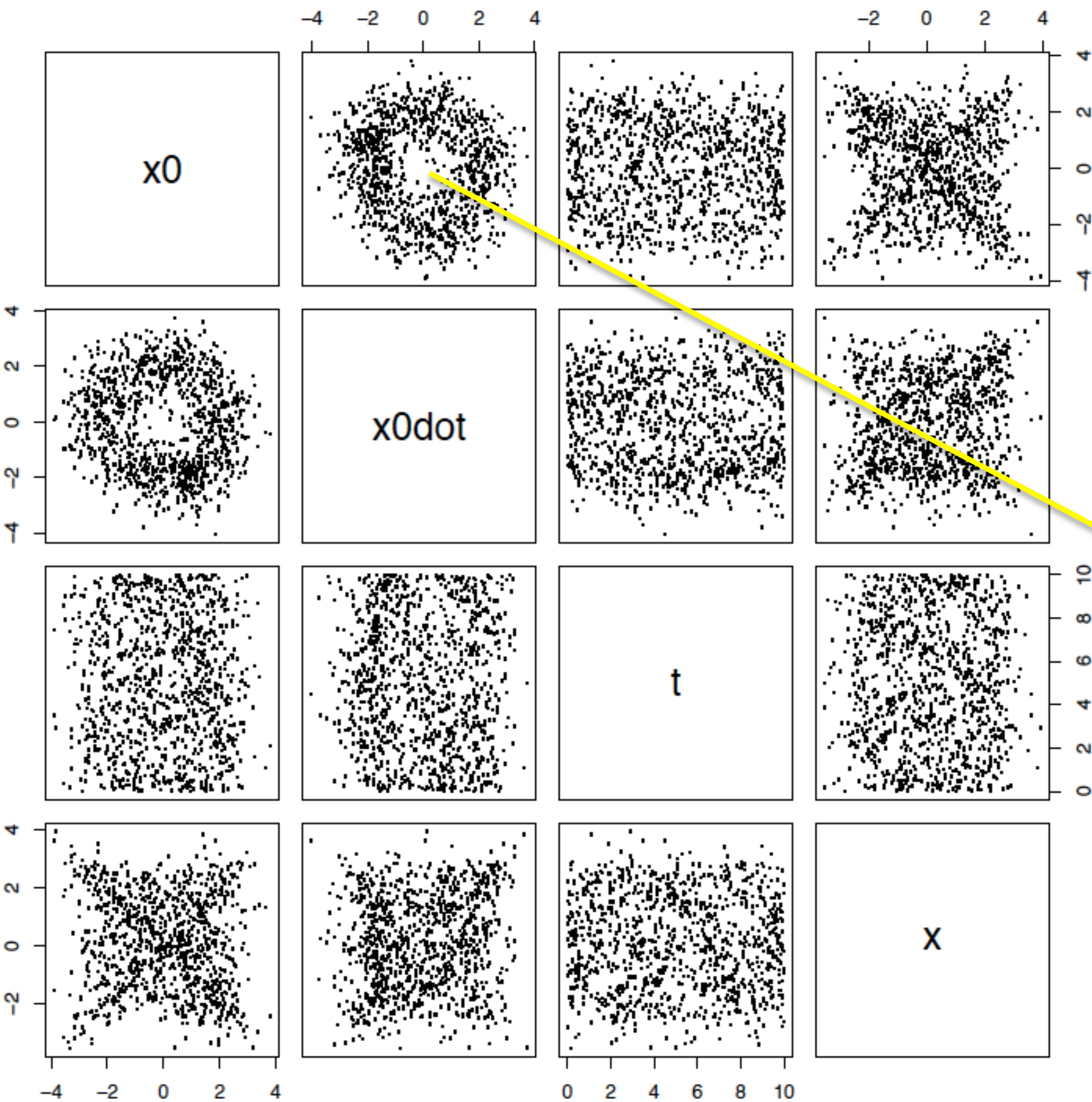
a couple (x_0, x_0', t) ; x is one 'simulation'

passive learner receives 1000 couples with (x_0, x_0', t) chosen at random and x calculated

active learner chooses the initial conditions (x_0, x_0', t) on which it has more doubts, calculates x

ACTIVE LEARNER REACHES THE SAME ERROR (MSE) AS A PASSIVE LEARNER WITH HALF AS MANY SIMULATIONS

Active learners run informative simulations



How much do you learn about the dynamics by placing a particle at rest at the bottom of the potential well?

Not much...

So the active learner undersamples this region (hole in the donut)

Active learners do not waste labelling effort on uninformative regions of parameter space

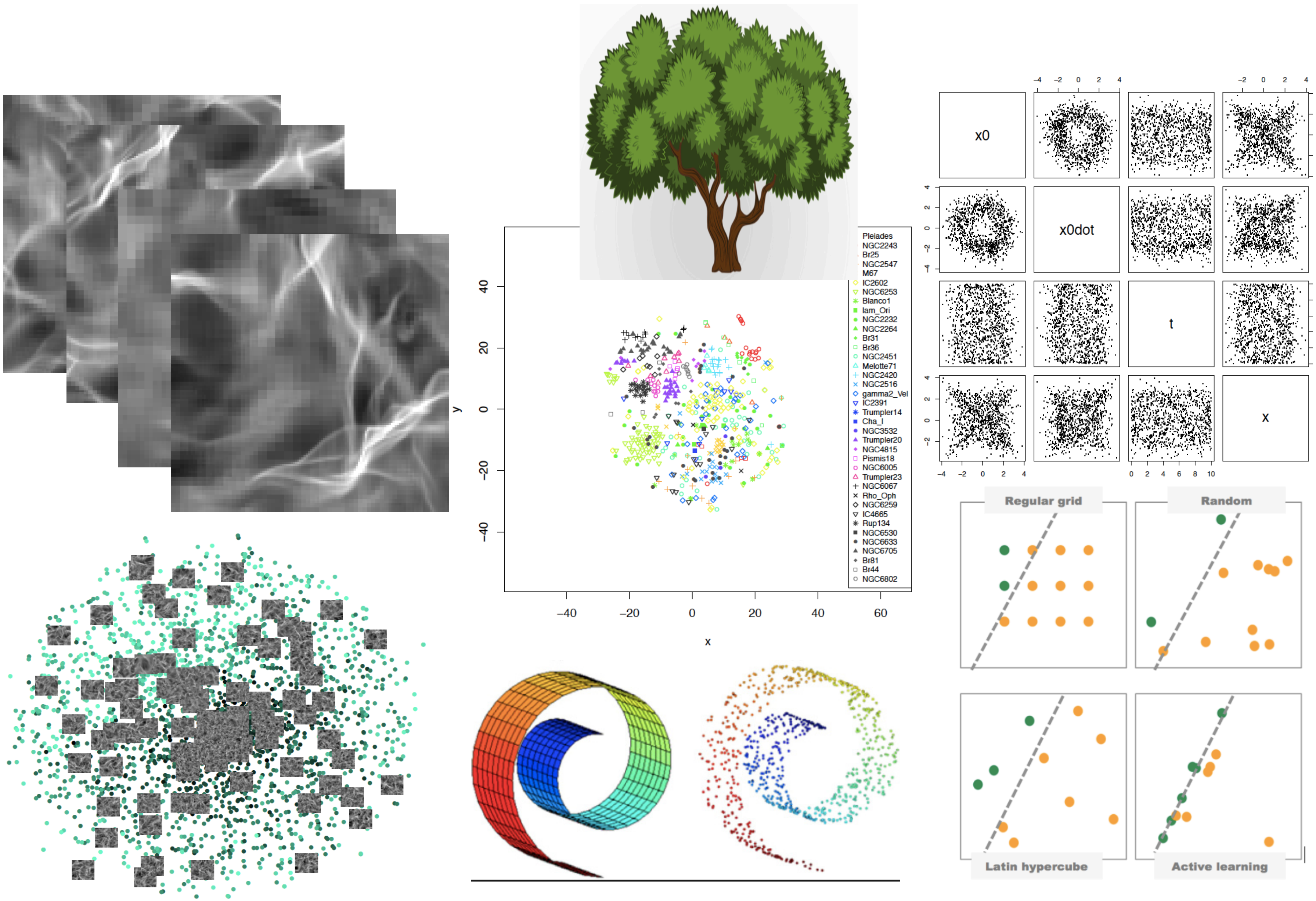
How to accomplish this?

How to pick what to study

Two students prepare for an exam together by solving practice tests.
They do not know the actual solutions, but they know whether their two solutions match.
They study again the topics where the solutions do not match.

- First something at random (e.g. 10 (x_0, x_0', t) ; x couples)
- On this, train two different models
- Generate a few candidate (x_0, x_0', t) at random
- Predict x_1 and x_2 with the two models
- Query the (x_0, x_0', t) for which the two predictions differ most strongly
- **'Query by committee'** – there are other schemes as well

Questions?



Pleiades
NGC2243
Br25
NGC2547
M67
IC2602
NGC6253
Blanco1
lam_Ori
NGC2232
NGC2264
Br31
Br36
NGC2451
Melotte71
NGC2420
NGC2516
gamma2_Vel
IC2391
Trumpler14
Cha_I
NGC3532
Trumpler20
NGC4815
Pismis18
NGC6005
Trumpler23
NGC6067
Rho_Oph
NGC6259
IC4665
Rup134
NGC6530
NGC6633
NGC6705
Br81
Br44
NGC6802
x
y
x0
x0dot
t
x
Regular grid
Random
Latin hypercube
Active learning

Acknowledgment

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 664931