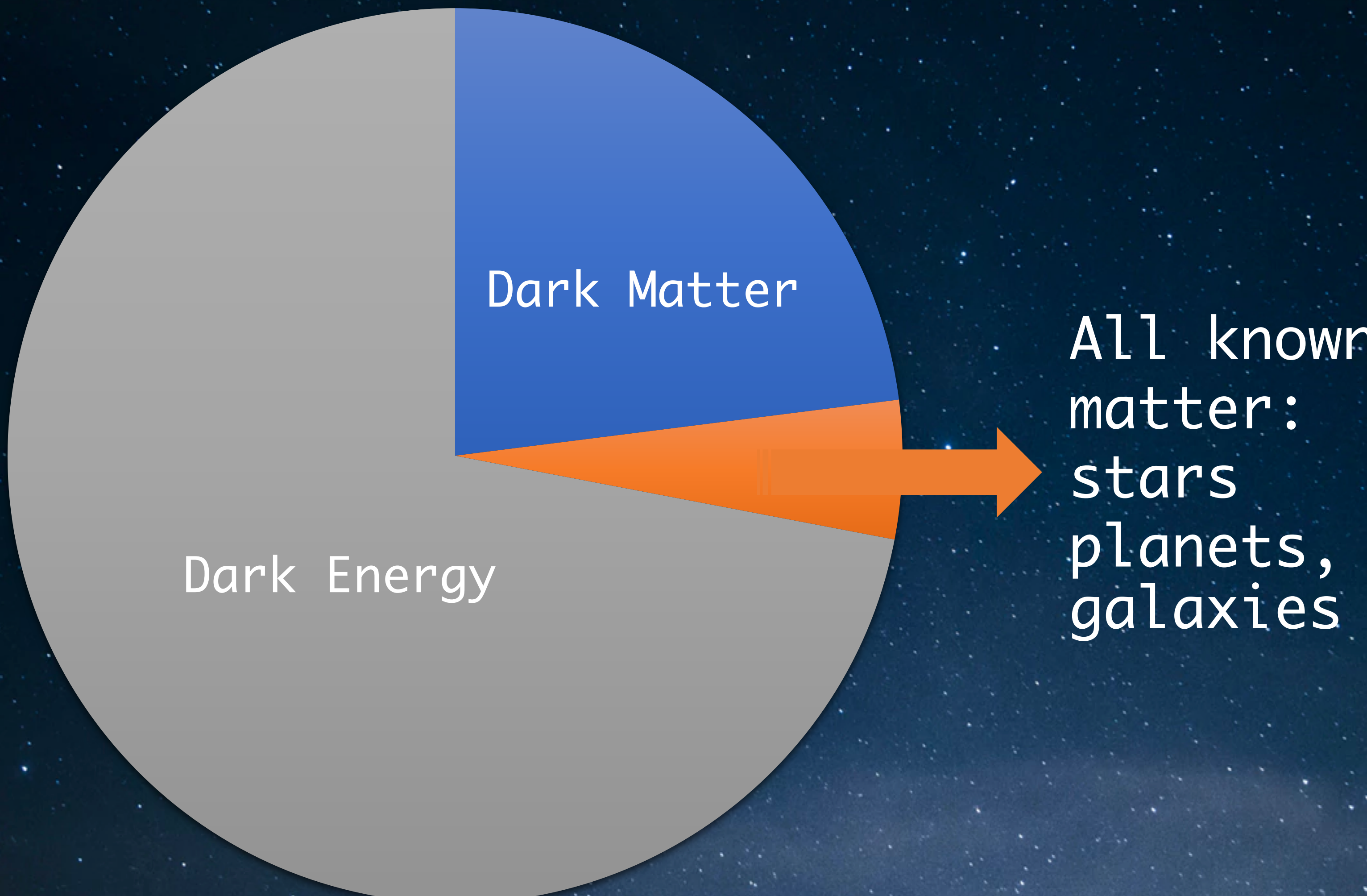
Zoeken naar nieuwe fysica met de hulp van Machine learning

Sascha Caron (Radboud University and Nikhef)

Bij de Steerkundige Vereeniging Triangulum in Apeldoorn

95 % of the Universe is unknown

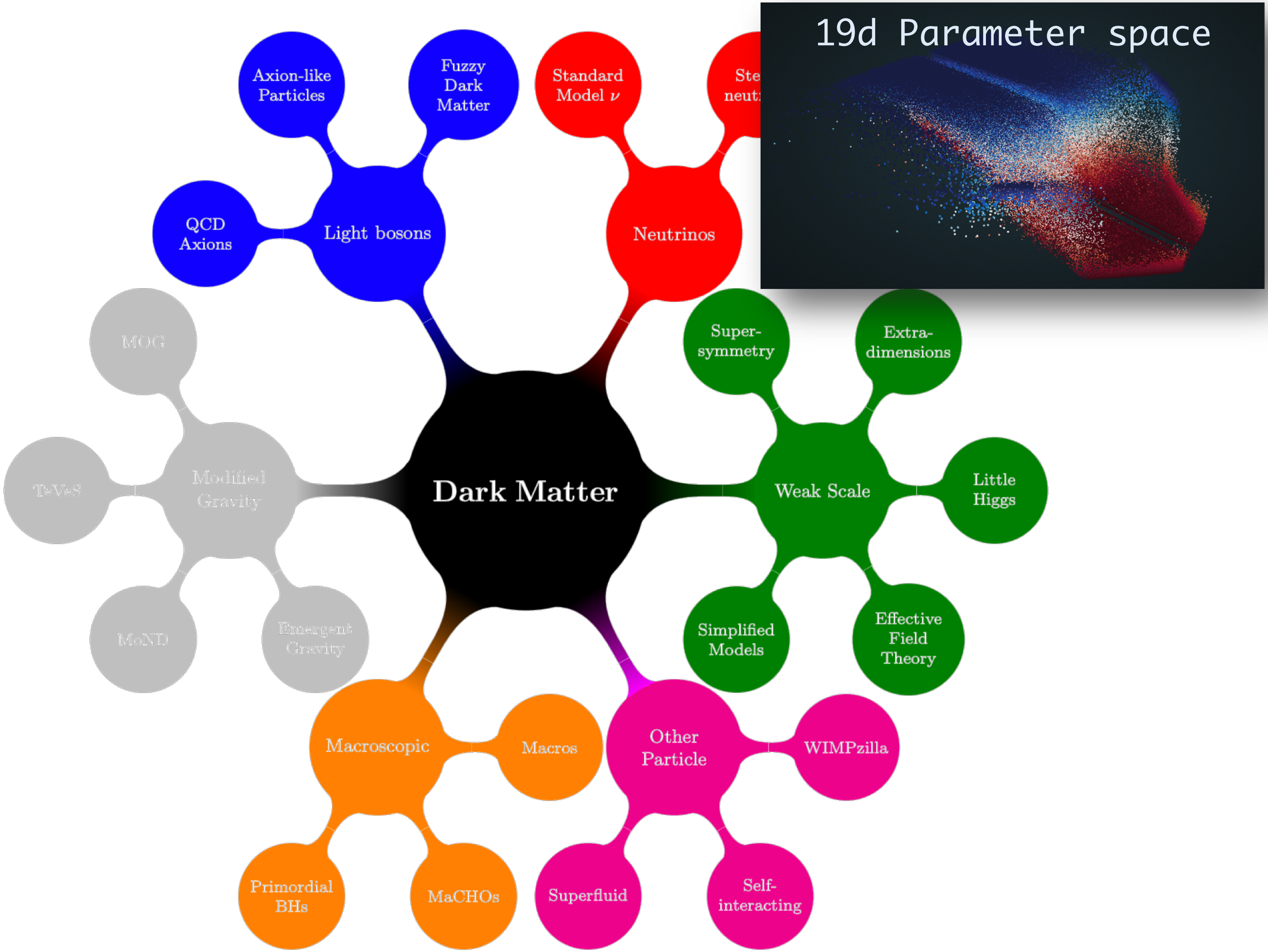


Higgs: You got a new toy, it‘s a playmobil castle with a size between 1-1000 cm. Can you find it ?

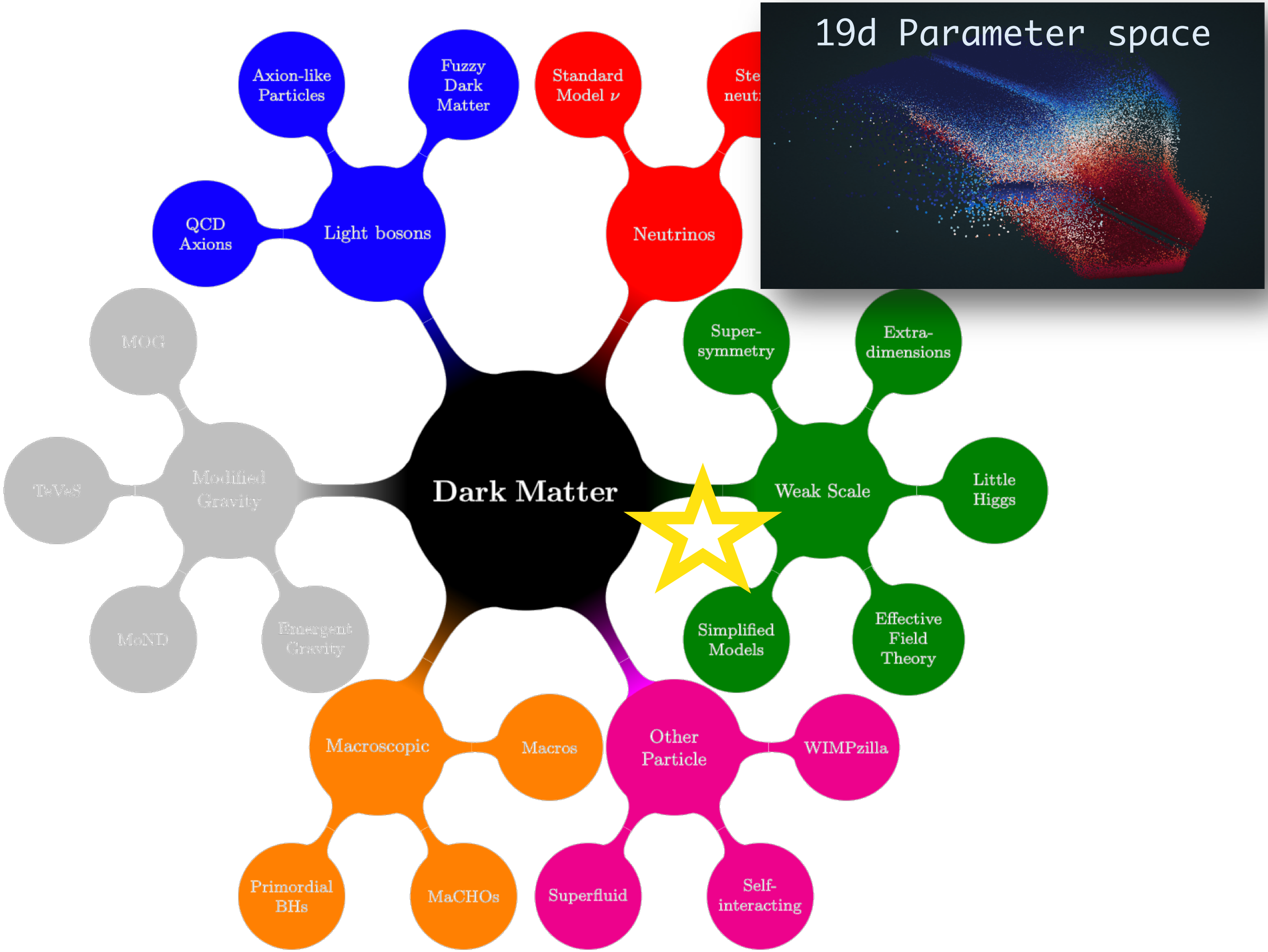
Today: I have a new toy for you, I put it somewhere in your room. The size is 0.1-100 cm. Can you find it ?

What could it be? Dark Matter models

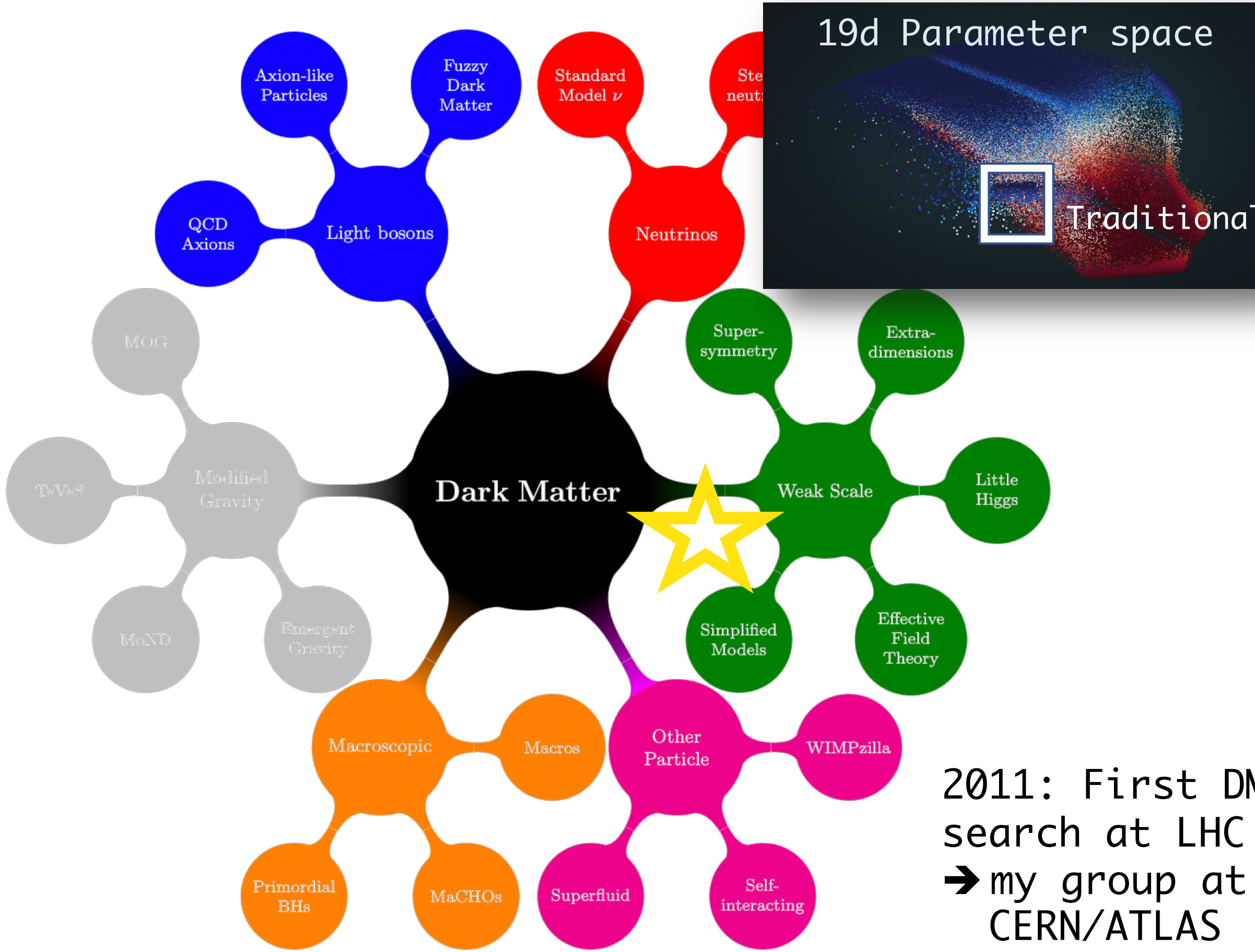
What could it be? Dark Matter models



What could it be? Dark Matter models



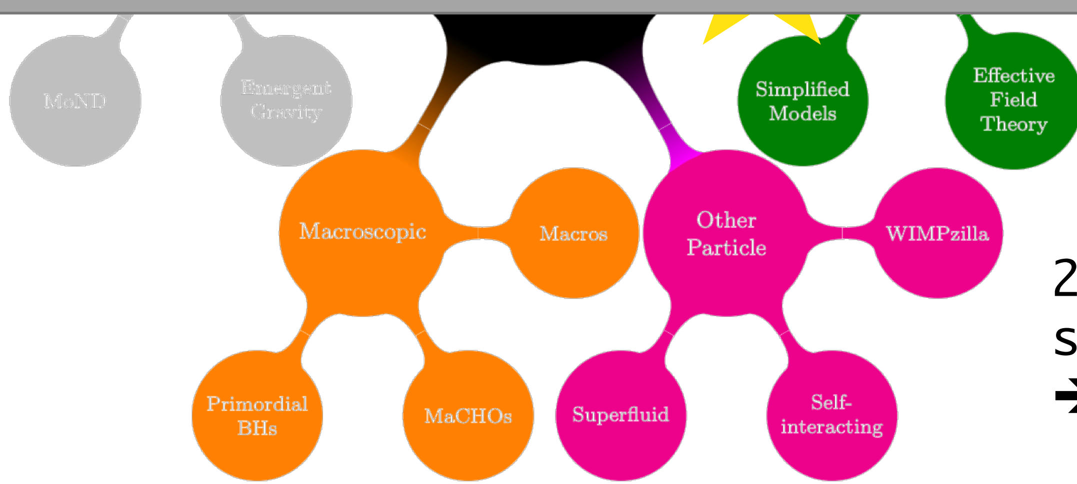
What could it be? Dark Matter models



2011: First DM search at LHC
➔ my group at CERN/ATLAS

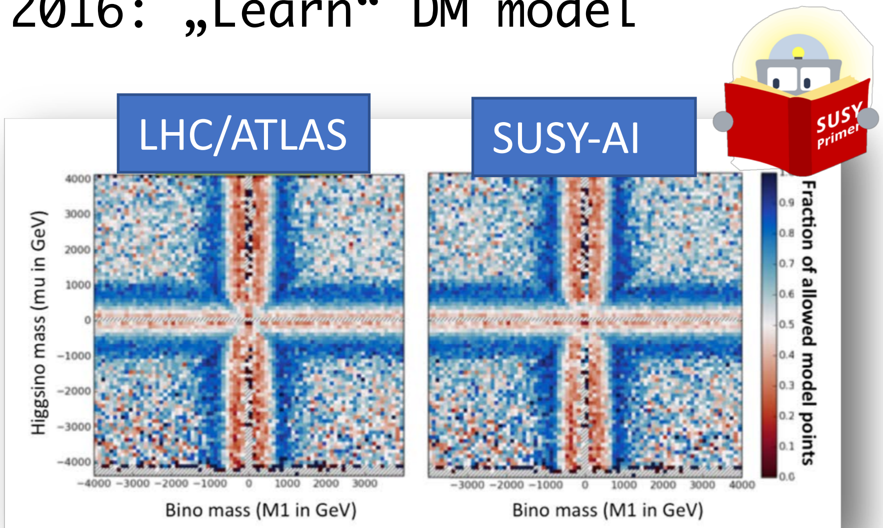
What could it be? Dark Matter models

2018:Exhaustive Automated Search Database of Models



2011: First DM search at LHC
➔ my group at CERN/ATLAS

2016: „Learn“ DM model



Classify/Predict DM from 19 parameters

2017: Scan gamma rays for DM with Deep Networks

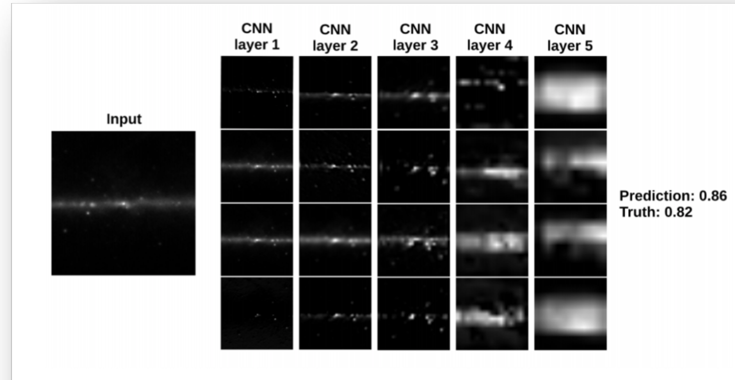
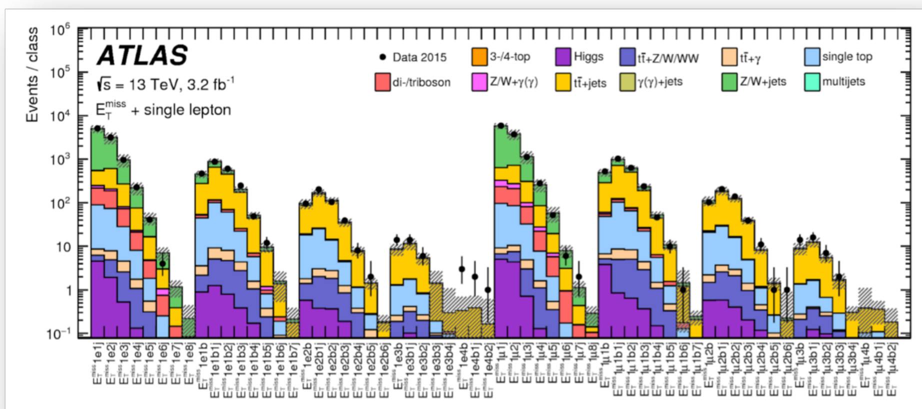


Figure 7: Activations of the different convolutional layers on a simulation. Each column

Scan Data with Million templates

2018: Automatize searches for anomalies

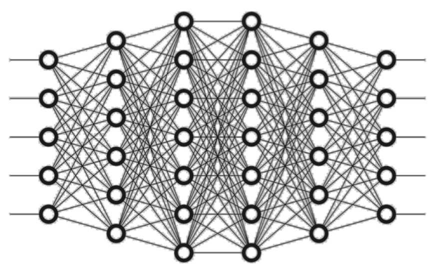


Collider Data: Scanning > 30.000 data selections

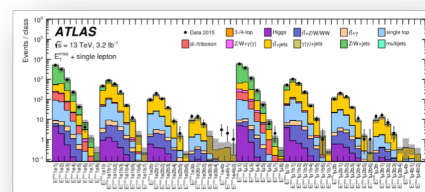
Traditional pipeline:

Model → Simulation → >days → Experiment 10

My goal:

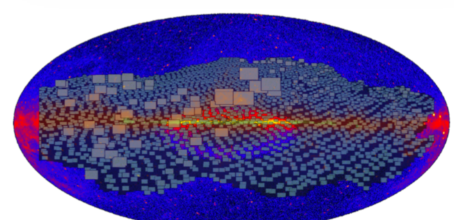


< 1 msec



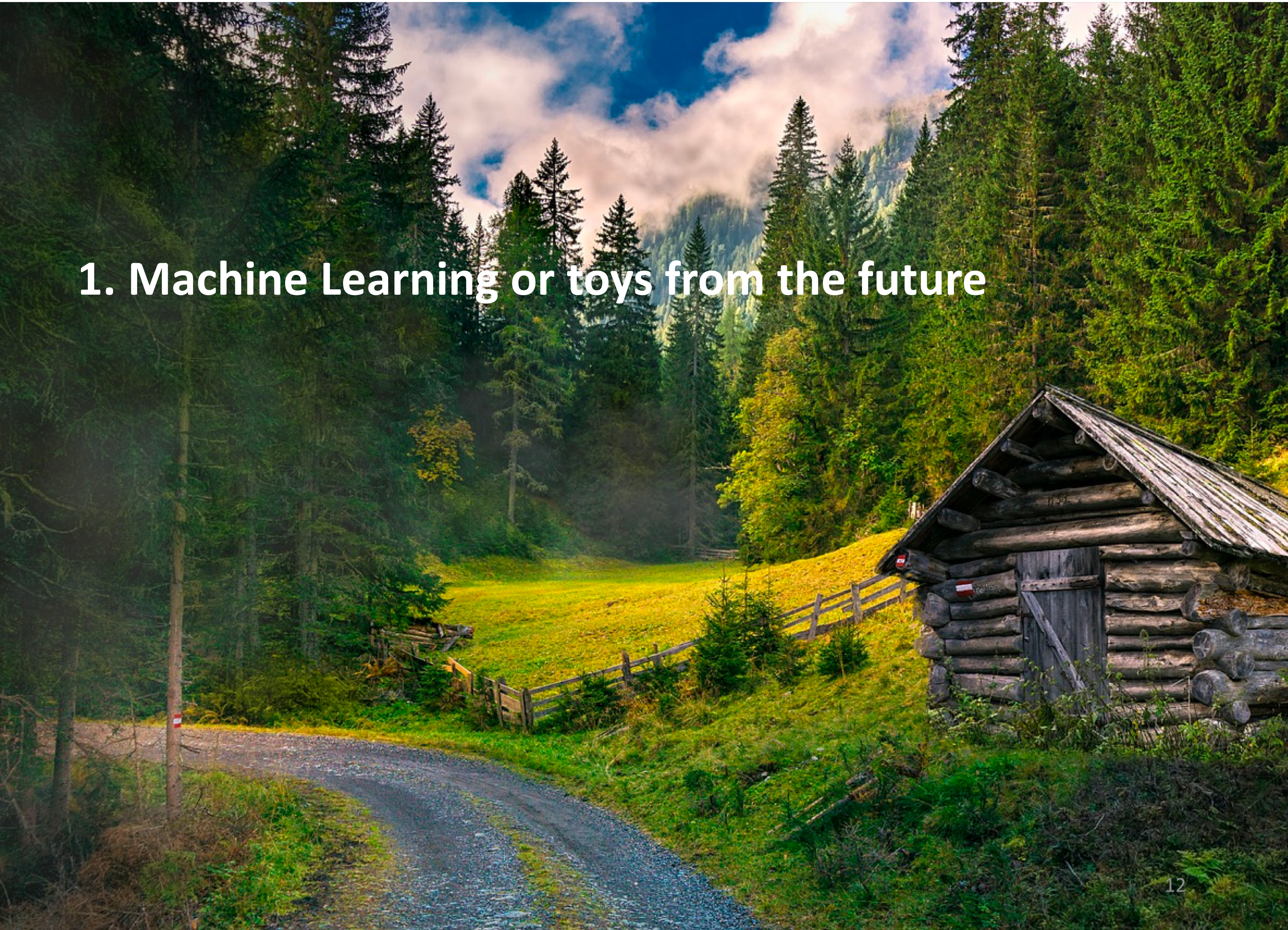
>100000

LHC-AI features



Astro

1. Machine Learning or toys from the future



What is Machine Learning ?

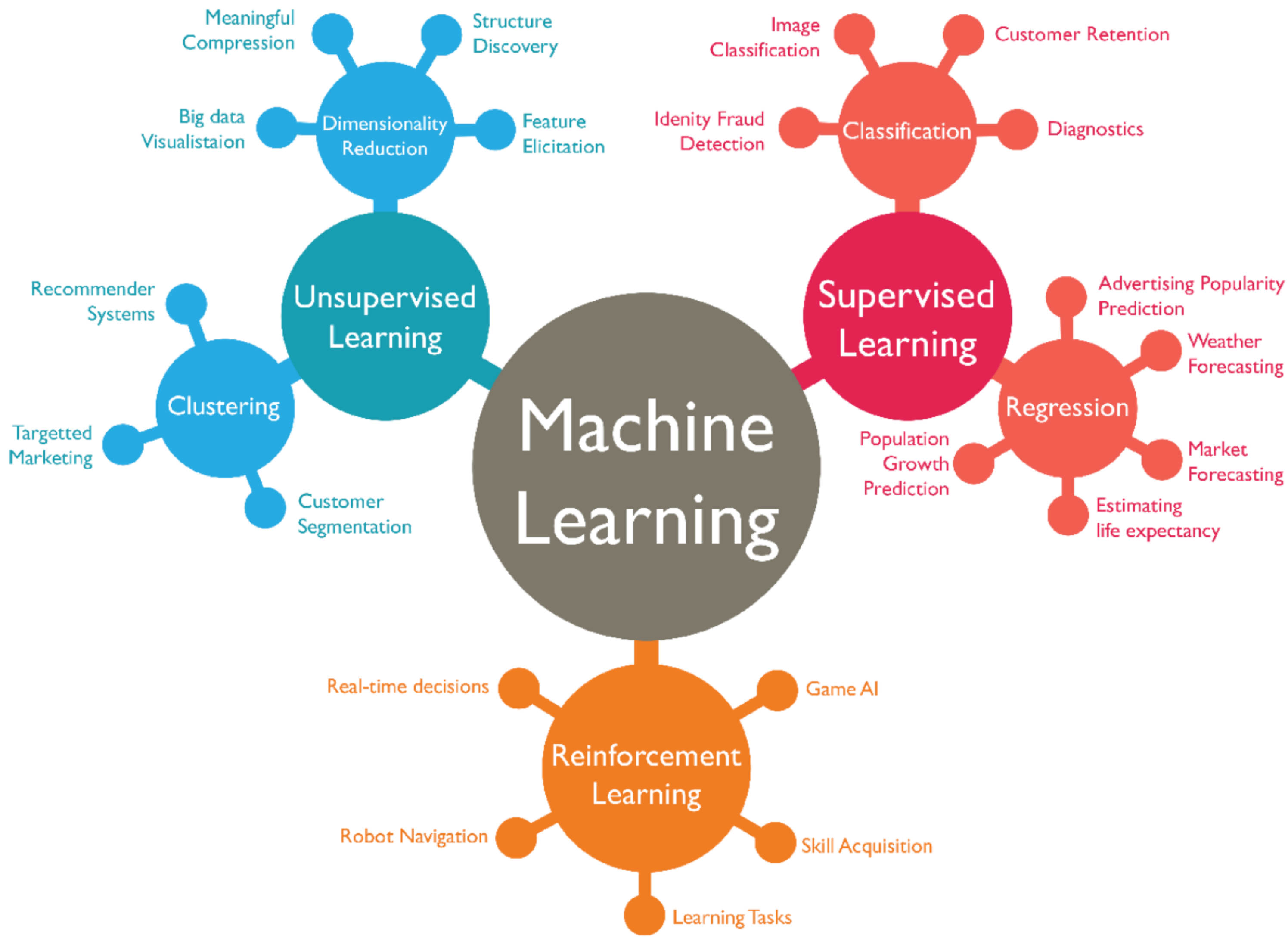


Image via Abdul Rahid

1a) regression

$\mathbf{f}(\vec{x})$s now a 1d function of a 1d variable x

Estimate function using polynominals

Trained function:

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \dots.$$

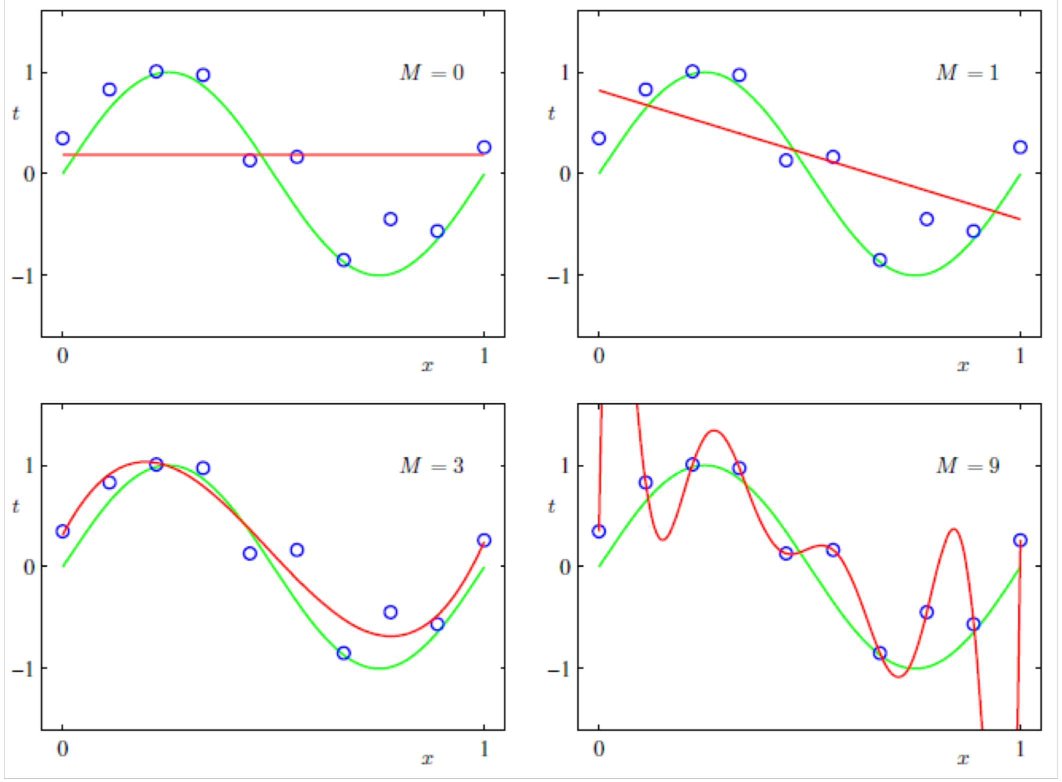
(linear model in the unknown parameter $\vec{w}$).

Problem is to determine "best" model parameters
This is done by defining an "error function" or loss function which is then "minimized".

$$E = \sum_{n=1}^{N} (y(x_n, \mathbf{w}) - f(x_n))^2$$

➔ Easy to solve

However: Which order of the polynomial ?



Example 1dim regression

$$E = \sum_{n=1}^{N} (y(x_n, \mathbf{w}) - f(x_n))^2$$

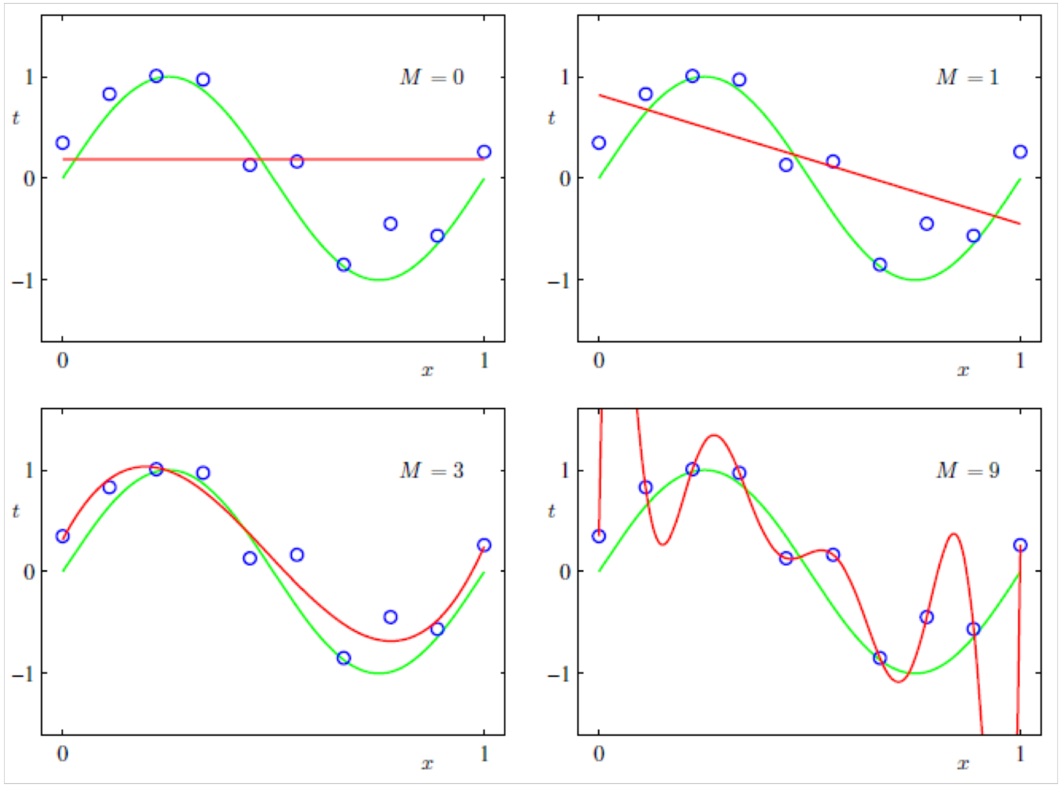
Trained function:

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots .$$

(linear model in the unknown parameter $\vec{w}$).

➔ Higher order polynominals
Naturally fit better, but
they do not agree with the true curve.
➔Overfitting

Can be seen in data by calculating the error function
of an <u>independent</u> data set
➔**Test data error function would be large !**
➔Dataset typically split in a "**training set**" and a "**testing set**"



Neural Network - regression

- Suppose we have a trivial model:

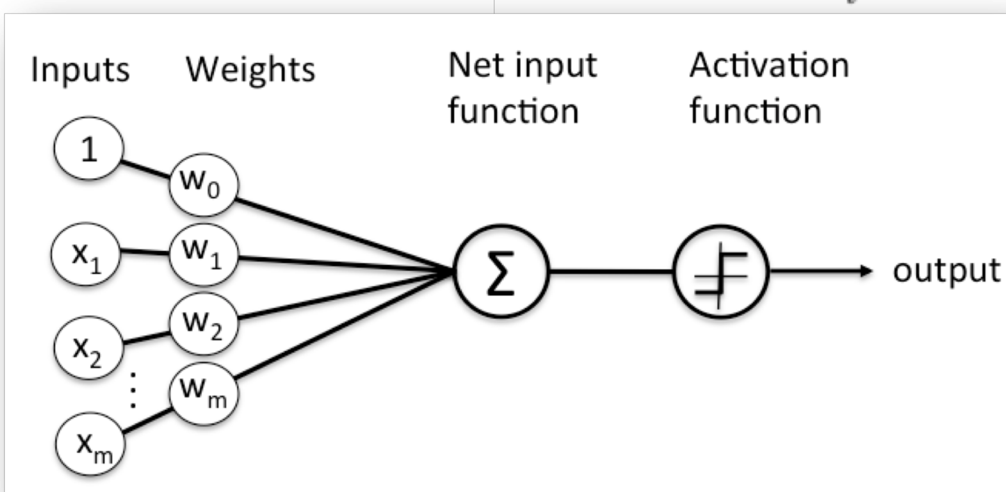
$$y(x, \mathbf{w}) = f(\sum_i w_i x)$$

and f is a non-linear activation function.

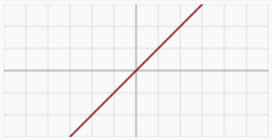
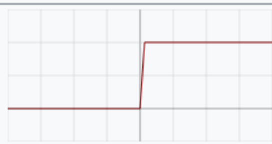
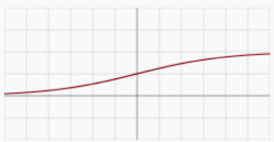
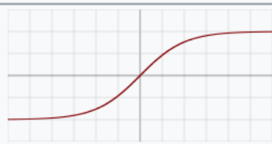
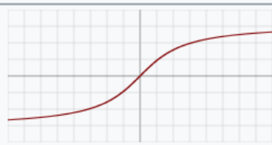
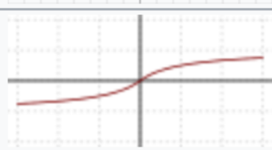
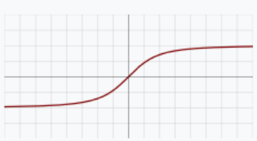
Let's make the input a vector (more input variables):

$$y(\vec{x}, \mathbf{w}) = f(\sum_i w_i x_i)$$

+ bias vector (here neglected, see NN lectures)



Activation functions examples

Identity		$f(x) = x$ Does not work for NN -> No non-linearities	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a. Sigmoid or Soft step)		$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$ [1]	$f'(x) = f(x)(1 - f(x))$
TanH		$f(x) = \tanh(x) = \frac{(e^{x} - e^{-x})}{(e^{x} + e^{-x})}$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2 + 1}$
Softsign [7][8]		$f(x) = \frac{x}{1 + \lvert x \rvert}$	$f'(x) = \frac{1}{(1 + \lvert x \rvert)^2}$
Inverse square root unit (ISRU)[9]		$f(x) = \frac{x}{\sqrt{1 + \alpha x^2}}$	$f'(x) = \left(\frac{1}{\sqrt{1 + \alpha x^2}}\right)^3$
Rectified linear unit (ReLU)[10]		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$

Neural Networks

- Let us now make the basis function itself nonlinear combinations of its inputs

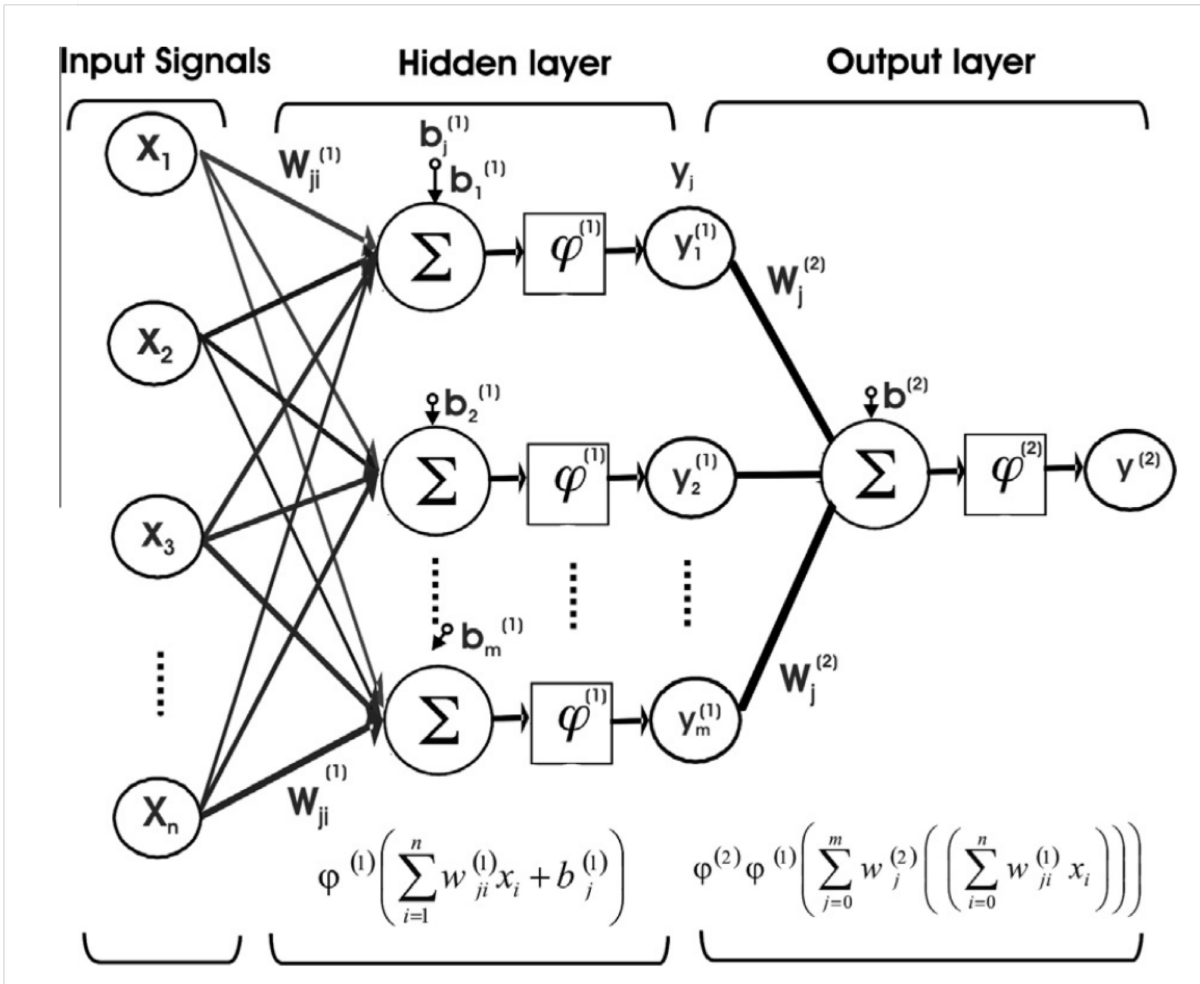
$$
\begin{aligned}
y^{(2)} &= \varphi^{(2)}\left(\sum_{j=1}^{m}\left(w_j^{(2)} y_j^{(1)} + b^{(2)}\right)\right) \\
&= \varphi^{(2)}\left(\sum_{j=1}^{m} w_j^{(2)} \varphi^{(1)}\left(\sum_{i=1}^{n} w_{ji}^{(1)} x_i + b_j^{(1)}\right) + b^{(2)}\right)
\end{aligned}
$$

And phi is a non-linear activation function, b is called bias

(bias allows to "shift" the activation function

Neur

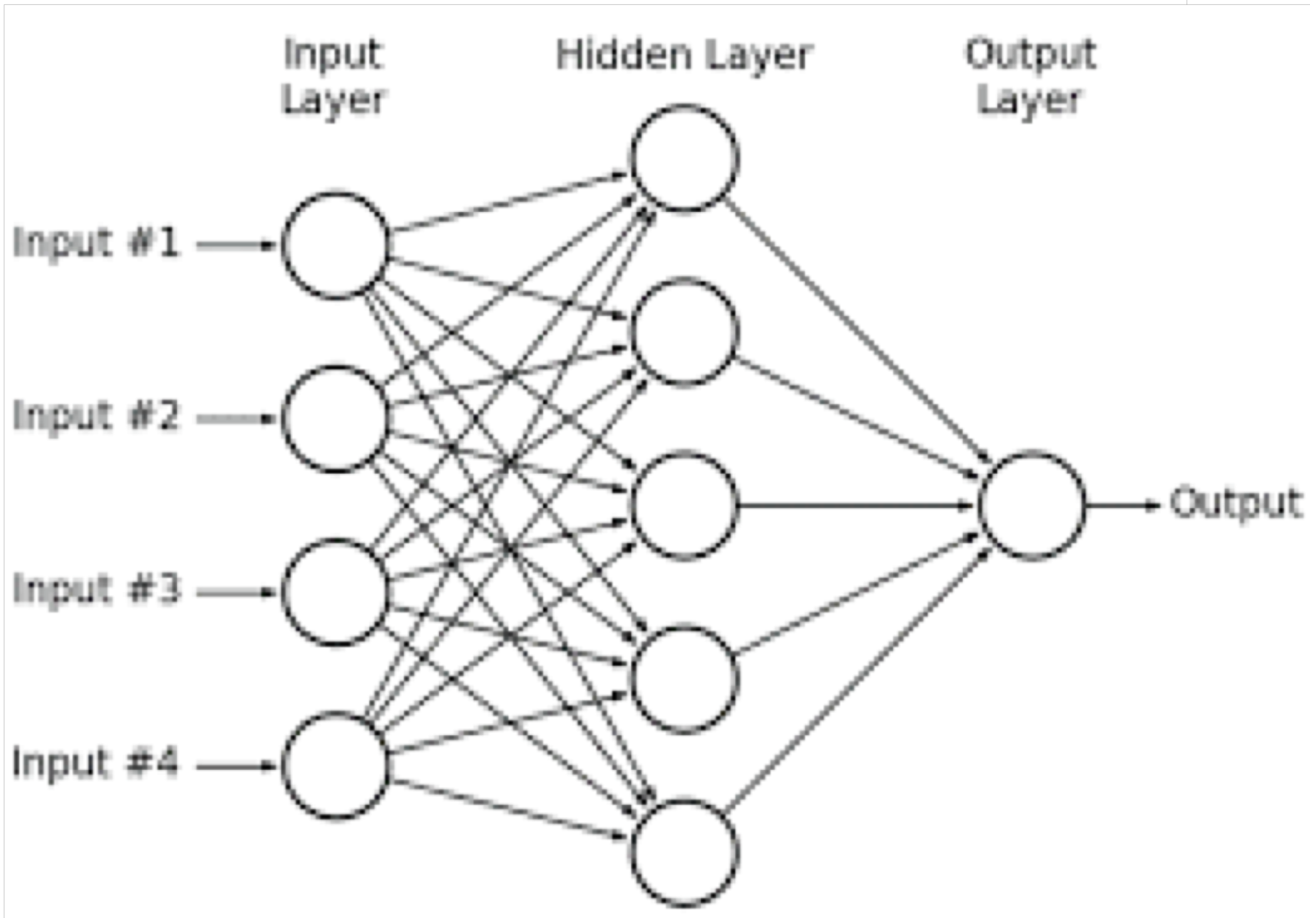
$$
\begin{aligned}
y^{(2)} &= \varphi^{(2)}\left(\sum_{j=1}^{m}\left(w_j^{(2)} y_j^{(1)} + b^{(2)}\right)\right) \\
&= \varphi^{(2)}\left(\sum_{j=1}^{m} w_j^{(2)} \varphi^{(1)}\left(\sum_{i=1}^{n} w_{ji}^{(1)} x_i + b_j^{(1)}\right) + b^{(2)}\right)
\end{aligned}
$$



y

Neur

$$
\begin{aligned}
y^{(2)} &= \varphi^{(2)}\left(\sum_{j=1}^{m}\left(w_j^{(2)} y_j^{(1)} + b^{(2)}\right)\right) \\
&= \varphi^{(2)}\left(\sum_{j=1}^{m} w_j^{(2)} \varphi^{(1)}\left(\sum_{i=1}^{n} w_{ji}^{(1)} x_i + b_j^{(1)}\right) + b^{(2)}\right)
\end{aligned}
$$



This is a 3 layer (1 hidden layer) feedforward (multilayer) perceptron

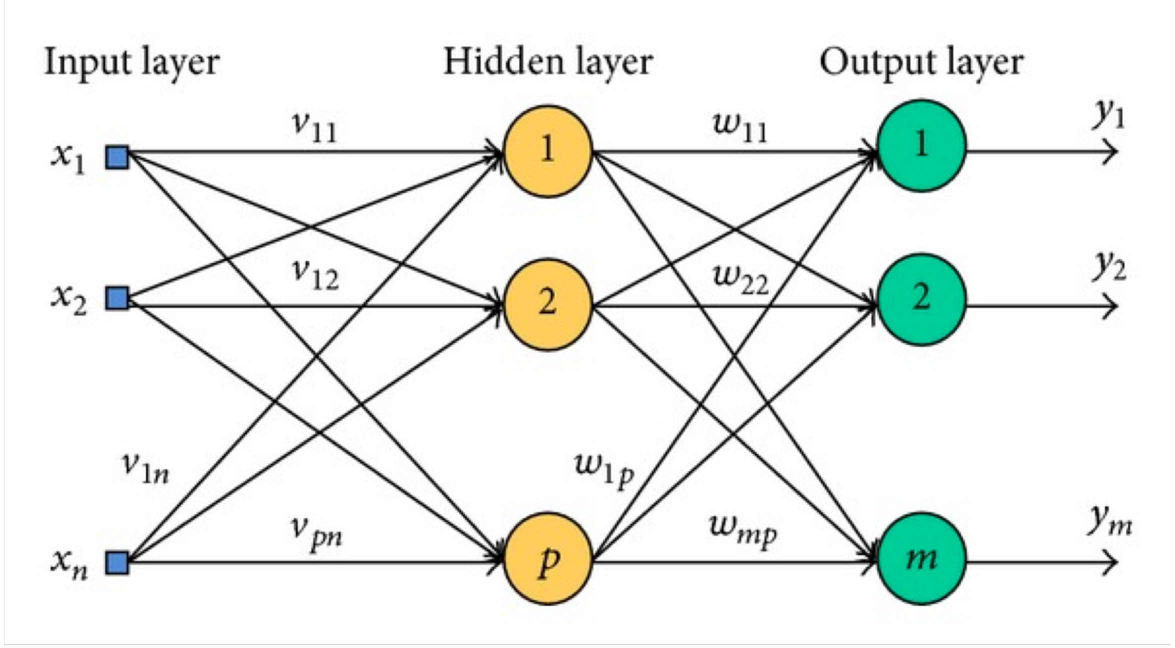
→This is the “simplest network”

“Training”:

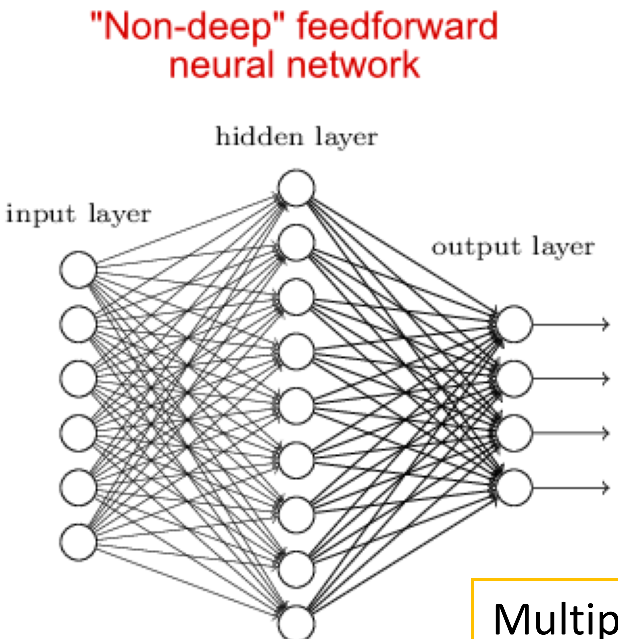
Finding the set of weights which minimize the error function

Neural Networks

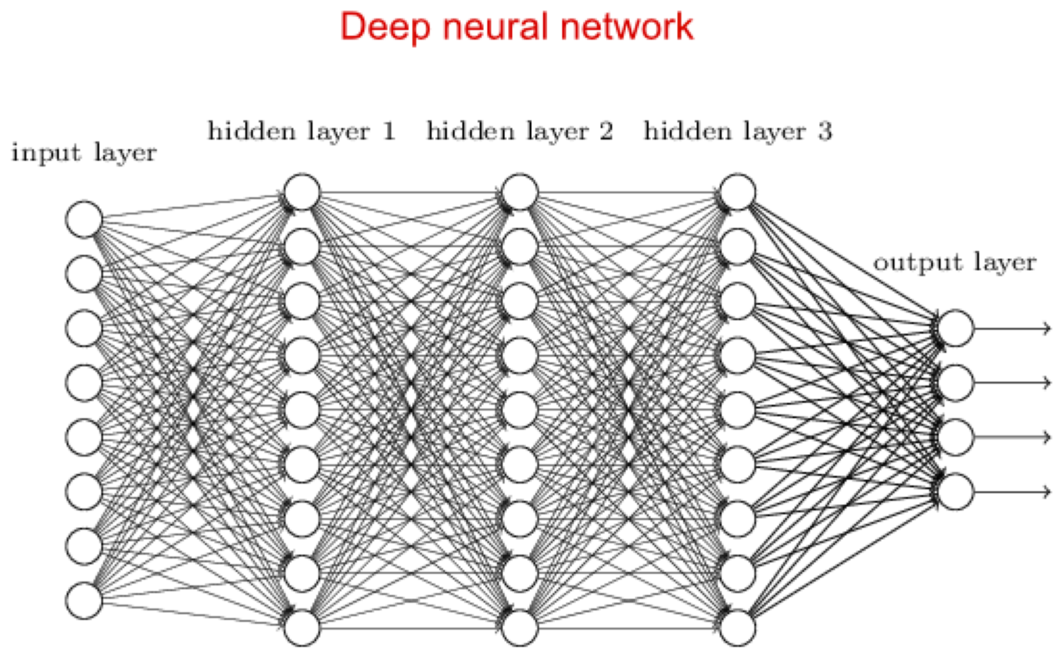
- Of course we can have multiple output nodes



… or multiple hidden layers…



Multiple Output Nodes !



We often use 5-10 layers

2014 First deep network in HEP (begin 2018 we had 50 on arxiv)

Searching for Exotic Particles in High-Energy Physics with Deep Learning

P. Baldi,[1] P. Sadowski,[1] and D. Whiteson[2]

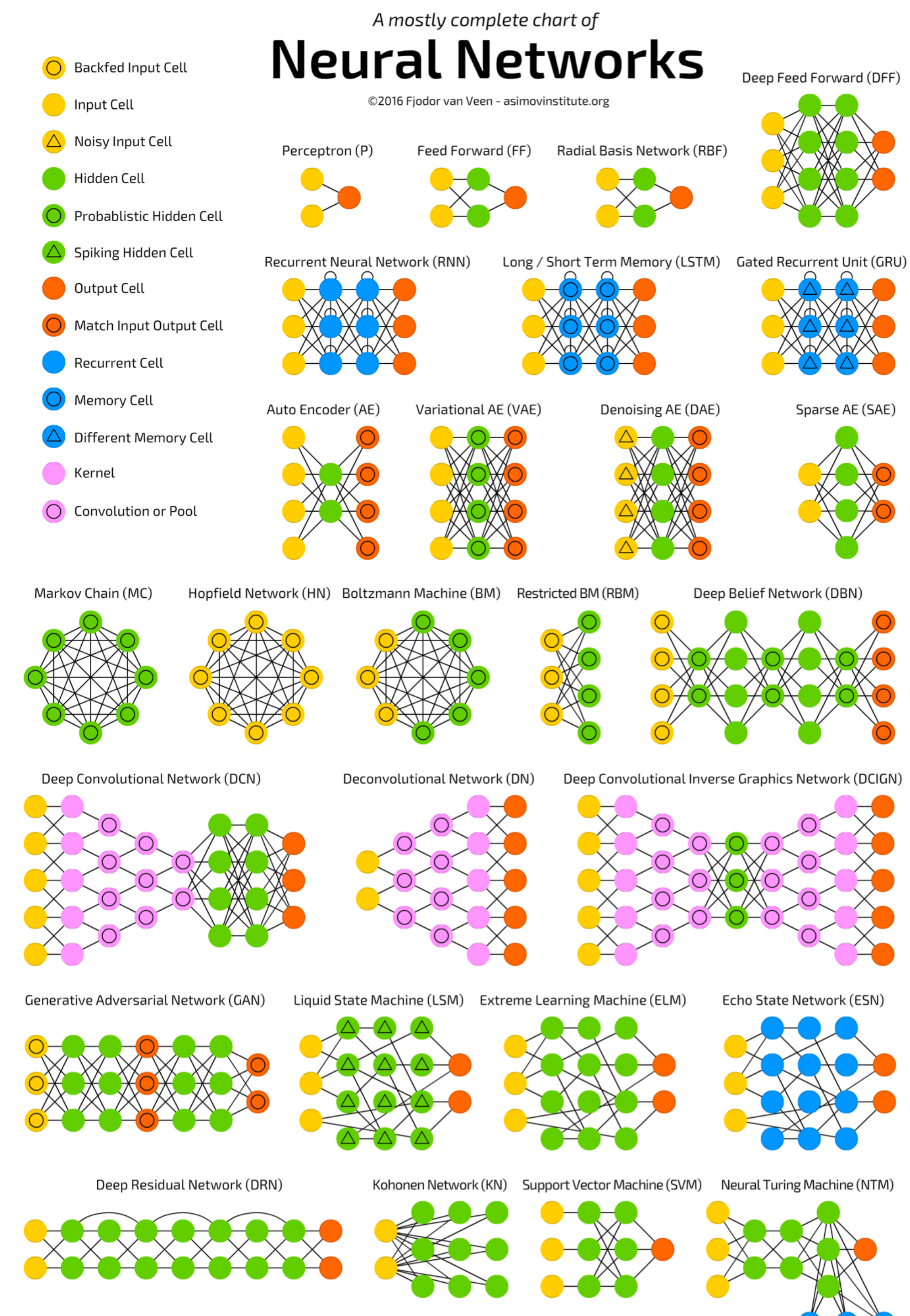
[1] *Dept. of Computer Science, UC Irvine, Irvine, CA 92617**
[2] *Dept. of Physics and Astronomy, UC Irvine, Irvine, CA 92617*†

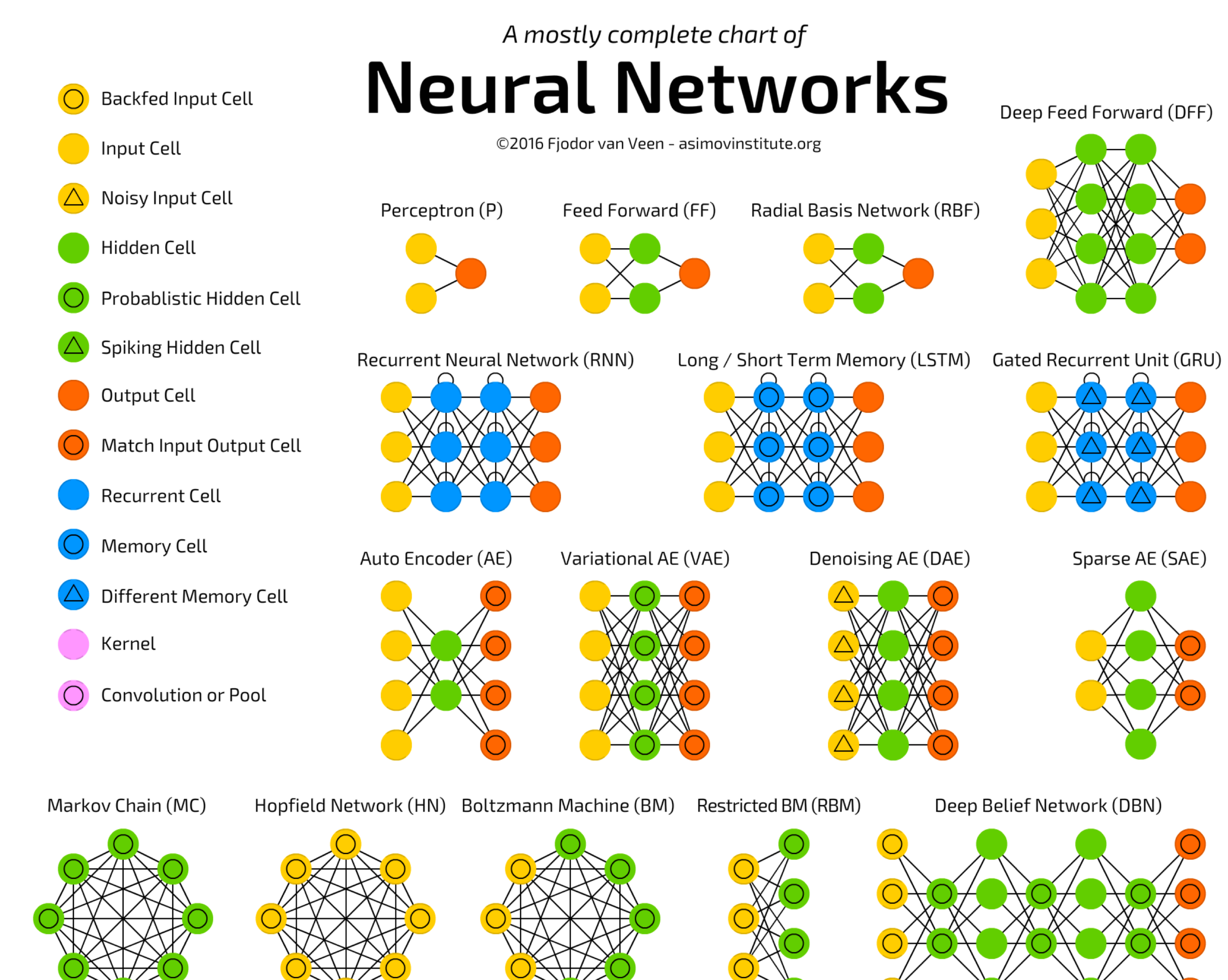
Collisions at high-energy particle colliders are a traditionally fruitful source of exotic particle discoveries. Finding these rare particles requires solving difficult signal-versus-background classification problems, hence machine learning approaches are often used. Standard approaches have relied on 'shallow' machine learning models that have a limited capacity to learn complex non-linear functions of the inputs, and rely on a pain-staking search through manually constructed non-linear features. Progress on this problem has slowed, as a variety of techniques have shown equivalent performance. Recent advances in the field of deep learning make it possible to learn more complex functions and better discriminate between signal and background classes. Using benchmark datasets, we show that deep learning methods need no manually constructed inputs and yet improve the classification metric by as much as 8% over the best current approaches. This demonstrates that deep learning approaches can improve the power of collider searches for exotic particles.

	AUC		
Technique	Low-level	High-level	Complete
BDT	0.73 (0.01)	0.78 (0.01)	0.81 (0.01)
NN	0.733 (0.007)	0.777 (0.001)	0.816 (0.004)
DN	0.880 (0.001)	0.800 (< 0.001)	0.885 (0.002)

	Discovery significance		
Technique	Low-level	High-level	Complete
NN	2.5σ	3.1σ	3.7σ
DN	4.9σ	3.6σ	5.0σ

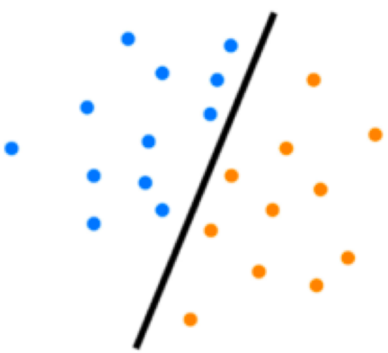
Important:
Input only 4 vectors !!!!
No knowledge about physics !!!!

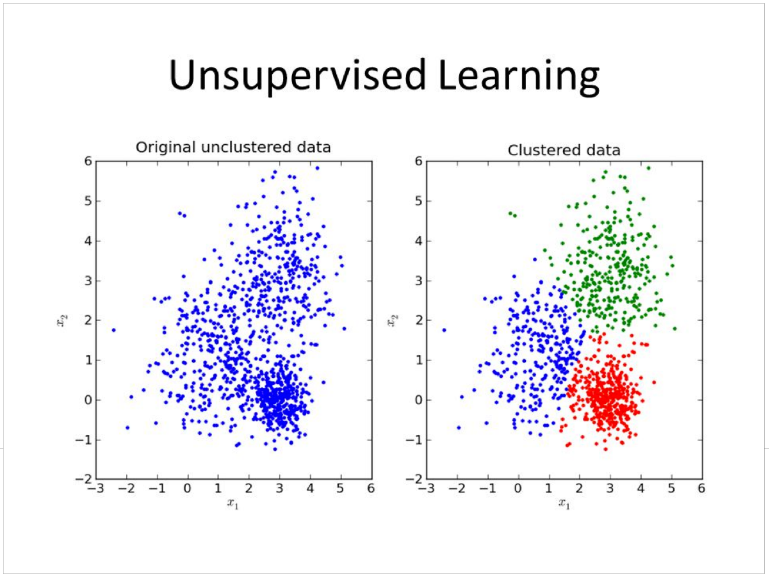

A mostly complete chart of
Neural Networks
©2016 Fjodor van Veen - asimovinstitute.org
Backfed Input Cell
Input Cell
Noisy Input Cell
Hidden Cell
Probablistic Hidden Cell
Spiking Hidden Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Different Memory Cell
Kernel
Convolution or Pool
Perceptron (P)
Feed Forward (FF)
Radial Basis Network (RBF)
Deep Feed Forward (DFF)
Recurrent Neural Network (RNN)
Long / Short Term Memory (LSTM)
Gated Recurrent Unit (GRU)
Auto Encoder (AE)
Variational AE (VAE)
Denoising AE (DAE)
Sparse AE (SAE)
Markov Chain (MC)
Hopfield Network (HN)
Boltzmann Machine (BM)
Restricted BM (RBM)
Deep Belief Network (DBN)
Deep Convolutional Network (DCN)
Deconvolutional Network (DN)
Deep Convolutional Inverse Graphics Network (DCIGN)
Generative Adversarial Network (GAN)
Liquid State Machine (LSM)
Extreme Learning Machine (ELM)
Echo State Network (ESN)
Deep Residual Network (DRN)
Kohonen Network (KN)
Support Vector Machine (SVM)
Neural Turing Machine (NTM)


A mostly complete chart of
Neural Networks
©2016 Fjodor van Veen - asimovinstitute.org
Backfed Input Cell
Input Cell
Noisy Input Cell
Hidden Cell
Probablistic Hidden Cell
Spiking Hidden Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Different Memory Cell
Kernel
Convolution or Pool
Perceptron (P)
Feed Forward (FF)
Radial Basis Network (RBF)
Deep Feed Forward (DFF)
Recurrent Neural Network (RNN)
Long / Short Term Memory (LSTM)
Gated Recurrent Unit (GRU)
Auto Encoder (AE)
Variational AE (VAE)
Denoising AE (DAE)
Sparse AE (SAE)
Markov Chain (MC)
Hopfield Network (HN)
Boltzmann Machine (BM)
Restricted BM (RBM)
Deep Belief Network (DBN)

Training methods

Supervised	Unsupervised	Reinforcement
Learning known patterns	Learning unknown patterns	Generating data Learning patterns

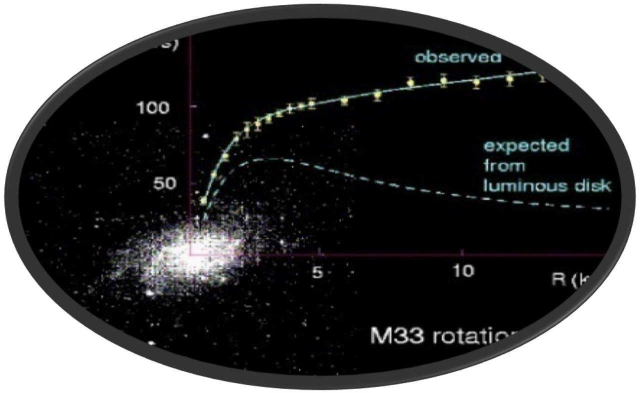




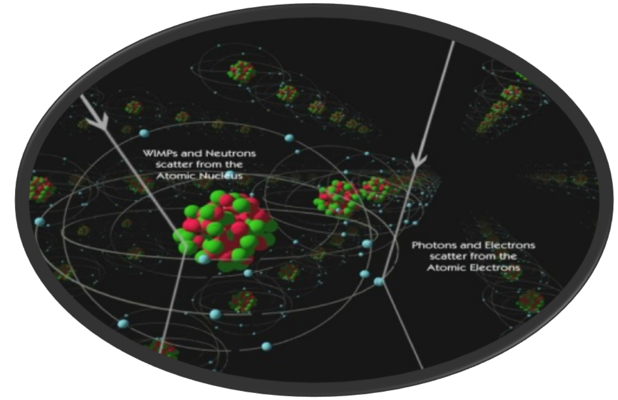
… there is also e.g. semi-supervised using labeled + unlabeled data

2. Astroparticle DM searches with Machines

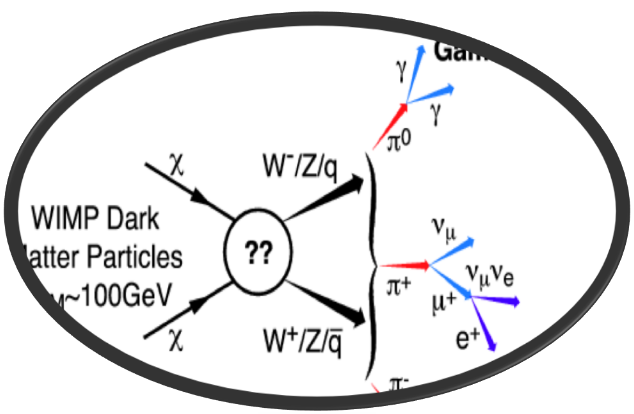
Dark Matter data gathering pillars



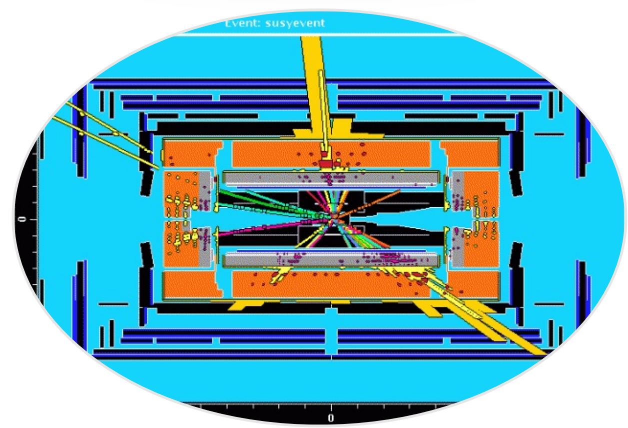
Gravitational interactions



Direct Detection



Indirect Detection



Production

Deep Convolutional Networks

Actually Alpha-go used a deep convolutional network…
What is this ?

We recently used deep convolutional networks to analyse gamma ray images for

Dark Matter

https://arxiv.org/abs/1708.06706

Convolutional Networks

- Convolutional networks have convolution layers based on “filters”, a filter (a **matrix**) maps “a group of numbers” to “a number” reducing the data → CONV layers
- There are also layers which only do a downsampling (lower the dimensionality) POOL or “fully connected layers” to process the final numbers…

important paper:LeCun, Yann. "LeNet-5, convolutional neural networks". Retrieved 16 November 2013.

Filters (Matrix)

- Unity

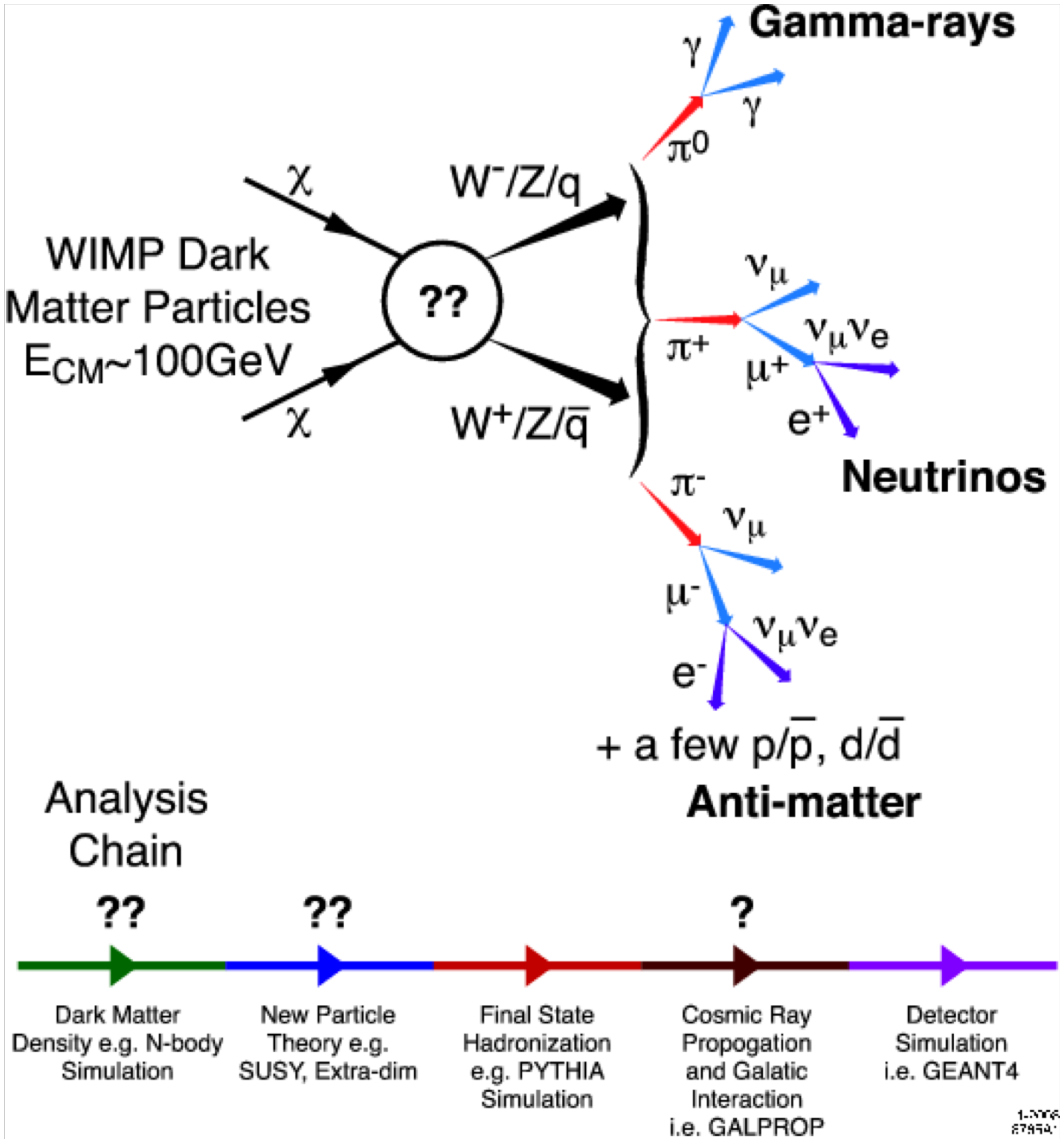
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

"Edge detector":

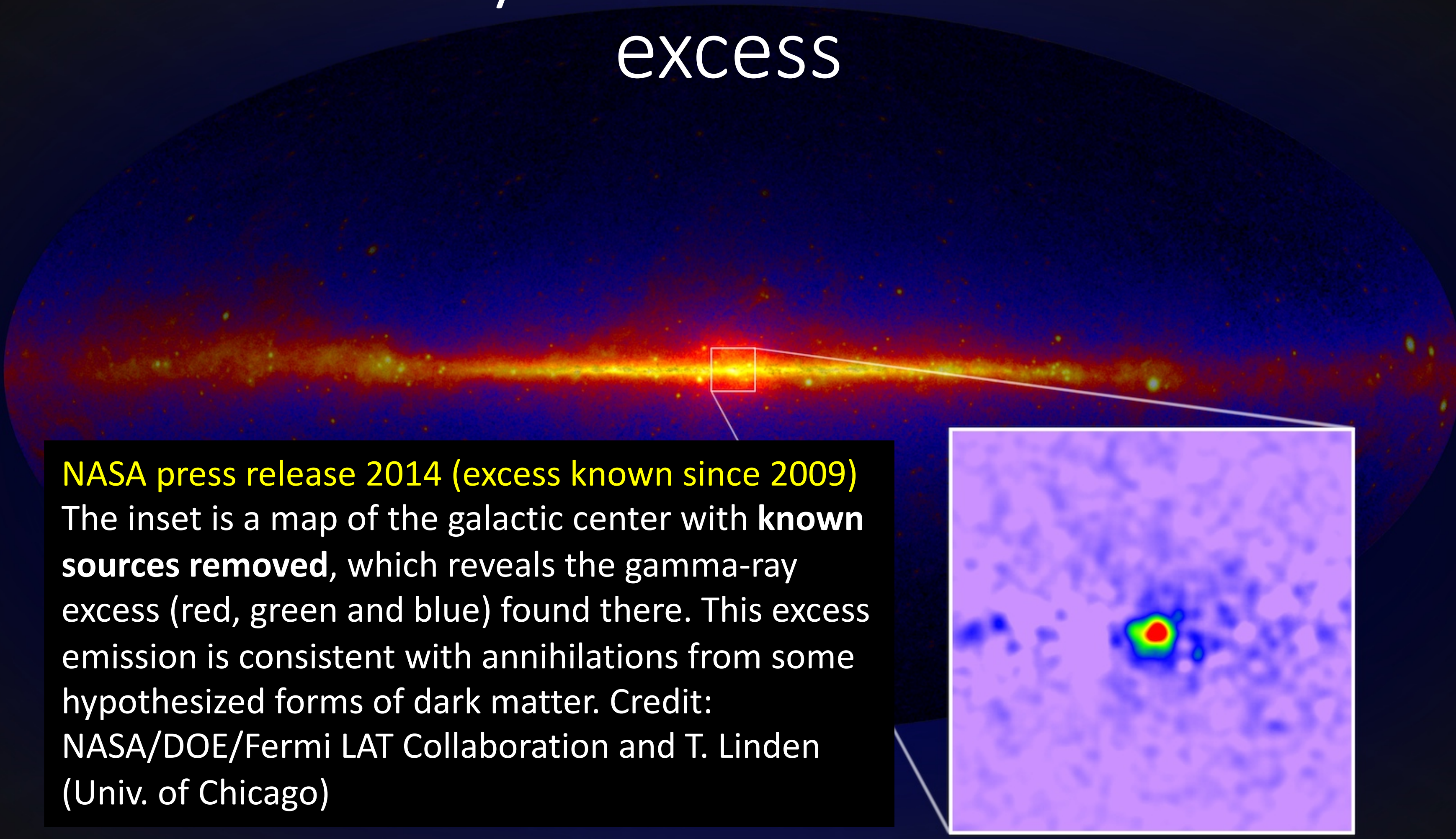
$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

"Convolutional" Network use "invariances" (rotation, translation) in data (e.g. images)

WIMP Astrophysics



Gamma rays & the Galactic Center excess



NASA press release 2014 (excess known since 2009)
The inset is a map of the galactic center with **known sources removed**, which reveals the gamma-ray excess (red, green and blue) found there. This excess emission is consistent with annihilations from some hypothesized forms of dark matter. Credit: NASA/DOE/Fermi LAT Collaboration and T. Linden (Univ. of Chicago)

Official paper in 2015

Fermi-LAT Observations of High-Energy Gamma-Ray Emission Toward the Galactic Center
Fermi-LAT Collaboration (M. Ajello (Clemson U.) *et al.*). Nov 9, 2015. 29 pp.
e-Print: **arXiv:1511.02938 [astro-ph.HE]** | PDF

Our convolutional network (convnet)

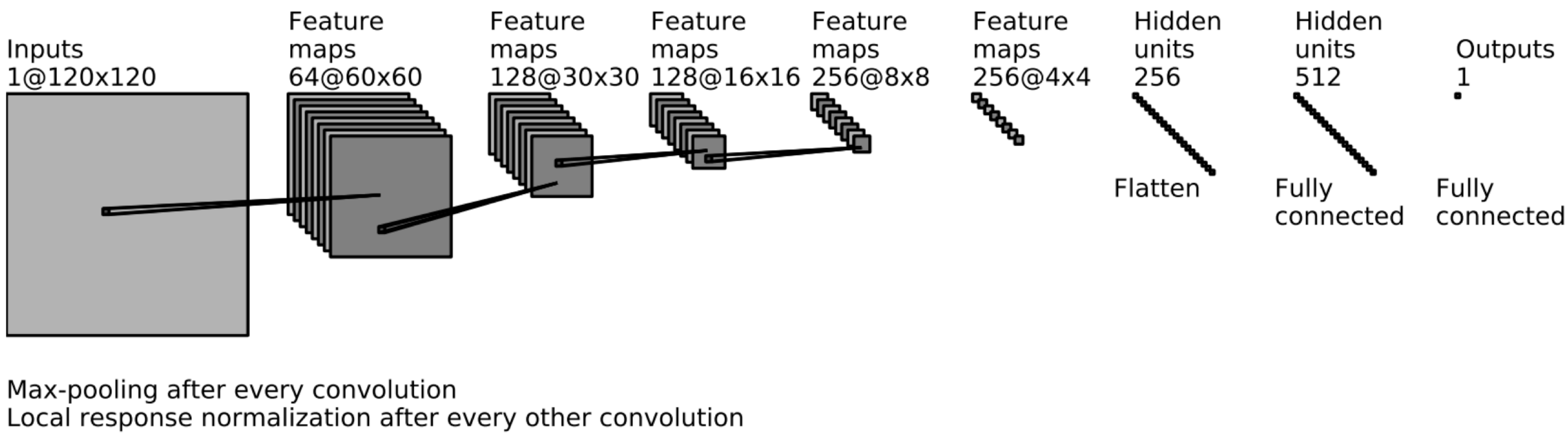


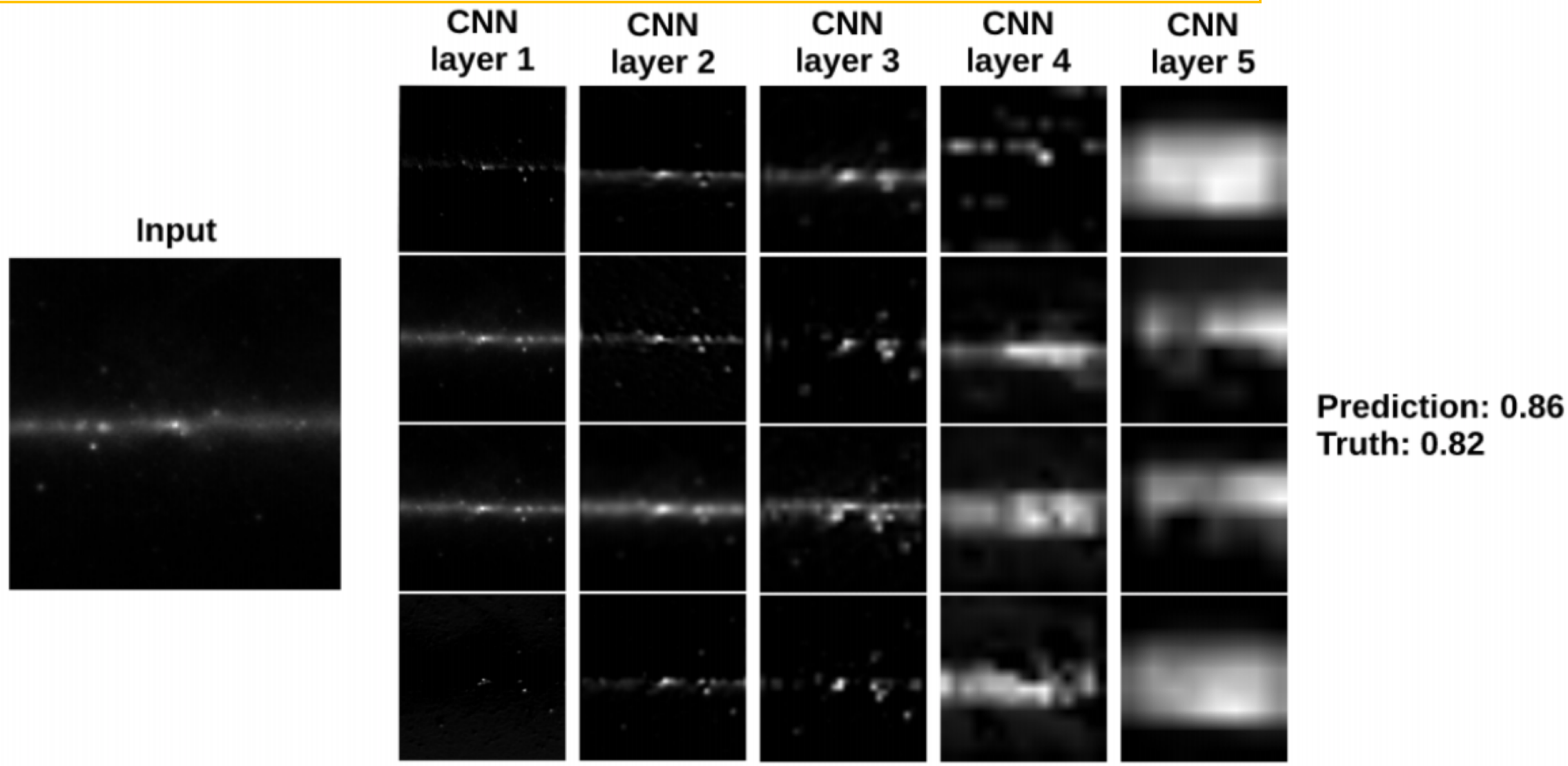
Figure 6: Visualization of the convolutional neural network. The network consists of an input layer, 5 convolutional + pooling layers, 2 fully connected layers and finally an output layer.

Hierarchical/compositional structure → smaller to larger structures
(reason: visible system is hierarchical as well…)

In comparison: GoogleLeNet has like 30 layers…

Isotropic or point sources: A Deep Convolutional Network approach

Output of the 5 convolutional layers can be "visualized" per event.



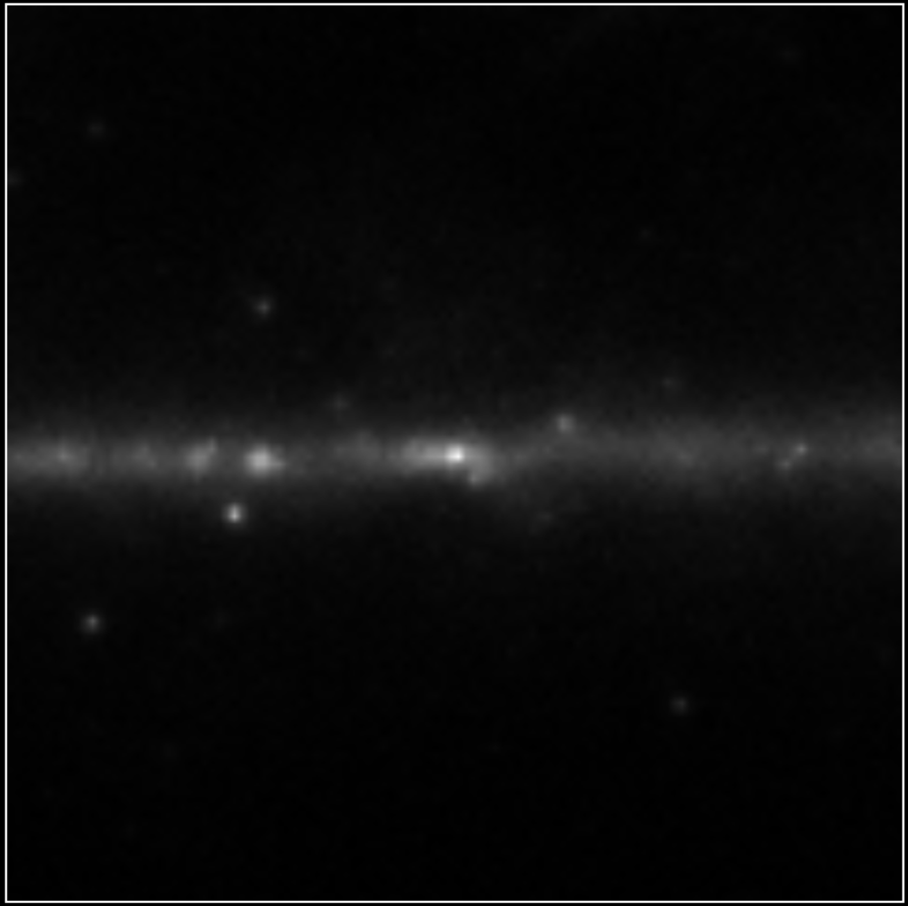
Activations of the network. Only four filters per layers are shown for clarity, between 256 and 65 filters are used for the different layers

arXiv:1708.06706

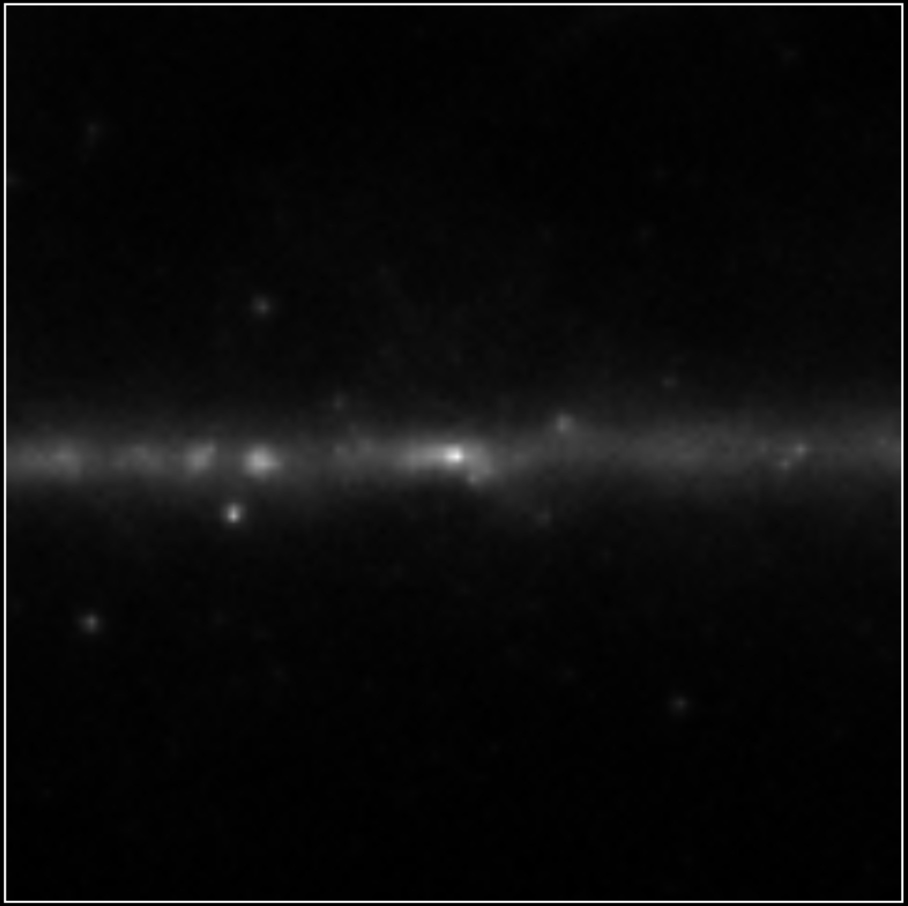
Guess the fraction of point sources

www.mydarkmachine.org

What is this fraction?



This is 0.5



Your prediction:

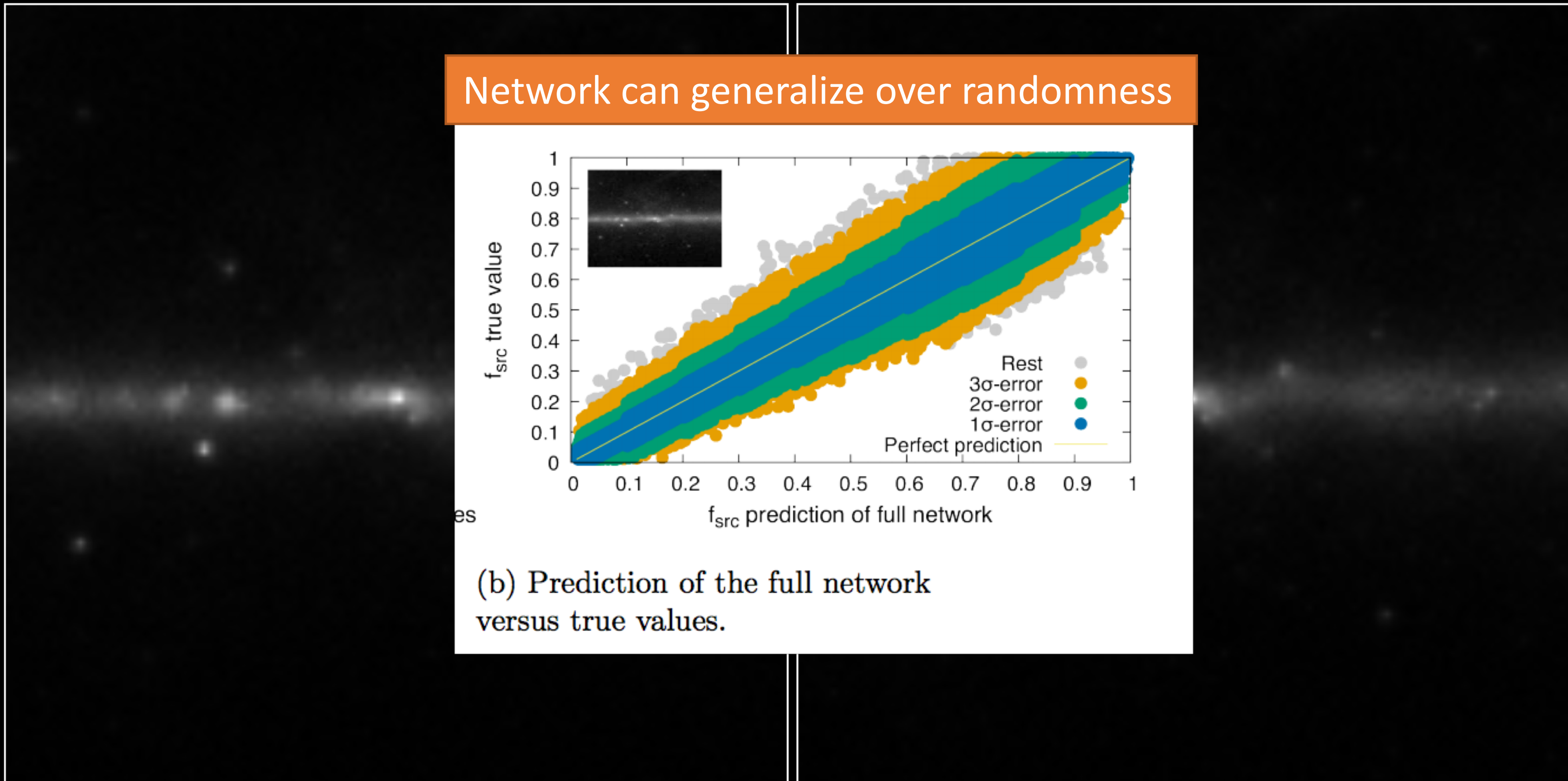
Invert image:

Guess

What is this fraction?

This is 0.5

Network can generalize over randomness



(b) Prediction of the full network versus true values.

Your prediction:
Invert image:

Truth:	0.052
Network:	0.1230
Your guess:	0.5

Who is better? The network

Interpretation here is
frequentists and relies on the model
to be correct (uncertainties
from toy experiments, no p-value yet)

Main message: Parameter determination of the physical model with a Neural Network

- Finally our goal is to **determine the model parameters** from 1 image ("real data")
- We do this by **training the network on "simulation"** ("simulated data")
- We need to **ensure that simulation agrees with data:** *Is the true image in the simulation parameter space of images ? **If not DM parameters maybe wrong !***

In simple words we do a "fit" to the image including all kinds of "unknown correlations" using a deep convolutional network trained on simulations

Next steps

- Categorize objects on the gamma-ray sky

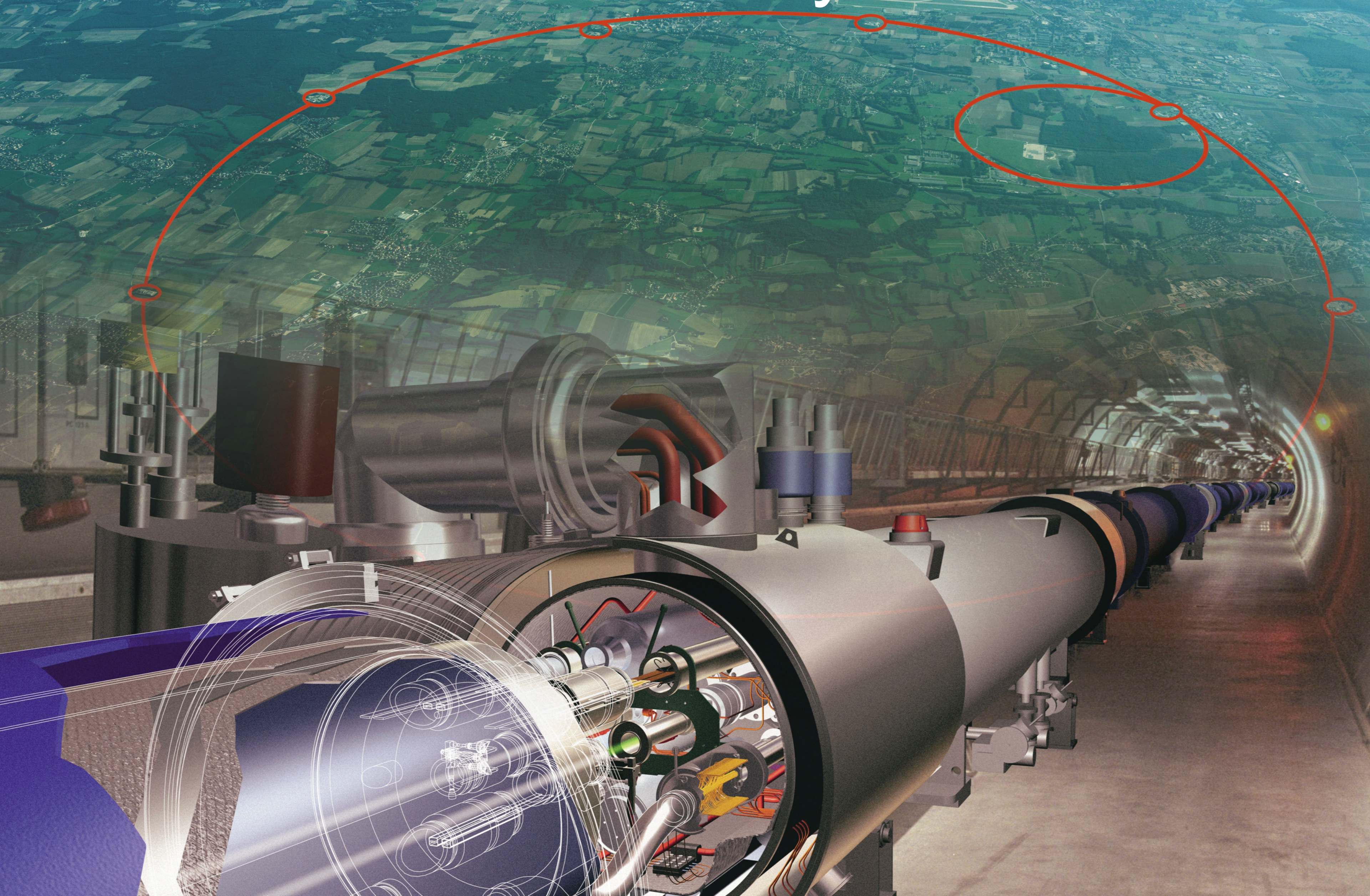
Also point source detection now → see e.g. recent paper called "deepsource" https://arxiv.org/abs/1807.02701

Can we simulate the most elementary interactions of our Universe with machine learning?

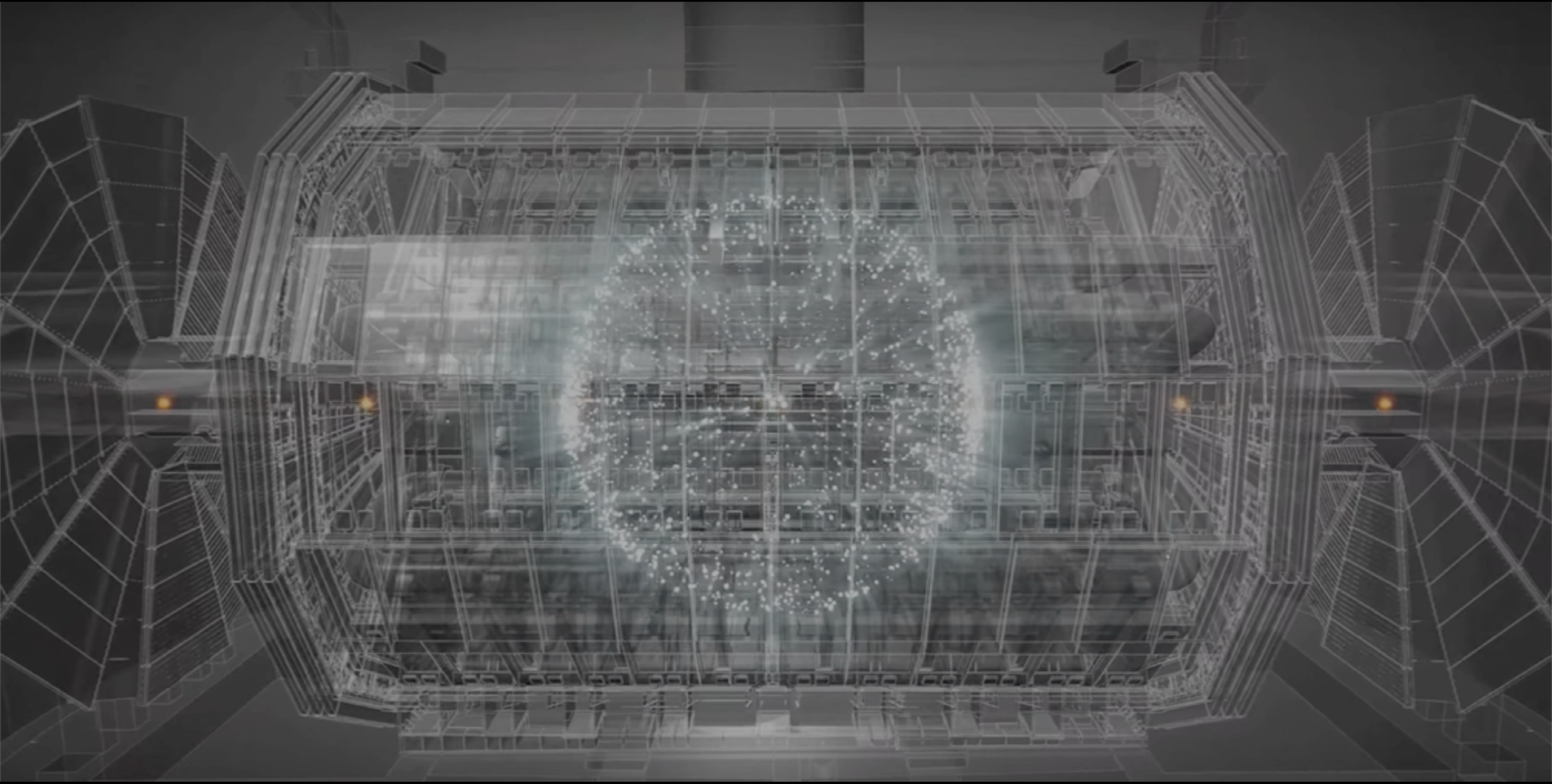
Sydney Otten,[1,2,*] Sascha Caron,[1,3,†] Wieske de Swart,[1] Melissa van Beekveld,[1,‡] Luc Hendriks,[1] Caspar van Leeuwen,[4] Damian Podareanu,[4] and Roberto Ruiz de Austri[5]

[1] *Institute for Mathematics, Astro- and Particle Physics IMAPP Radboud Universiteit, Nijmegen, The Netherlands*
[2] *GRAPPA, University of Amsterdam, The Netherlands*
[3] *Nikhef, Amsterdam, The Netherlands*
[4] *SURFsara, Amsterdam, The Netherlands*
[5] *Instituto de Fisica Corpuscular, IFIC-UV/CSIC University of Valencia, Spain*

The most elementary interactions

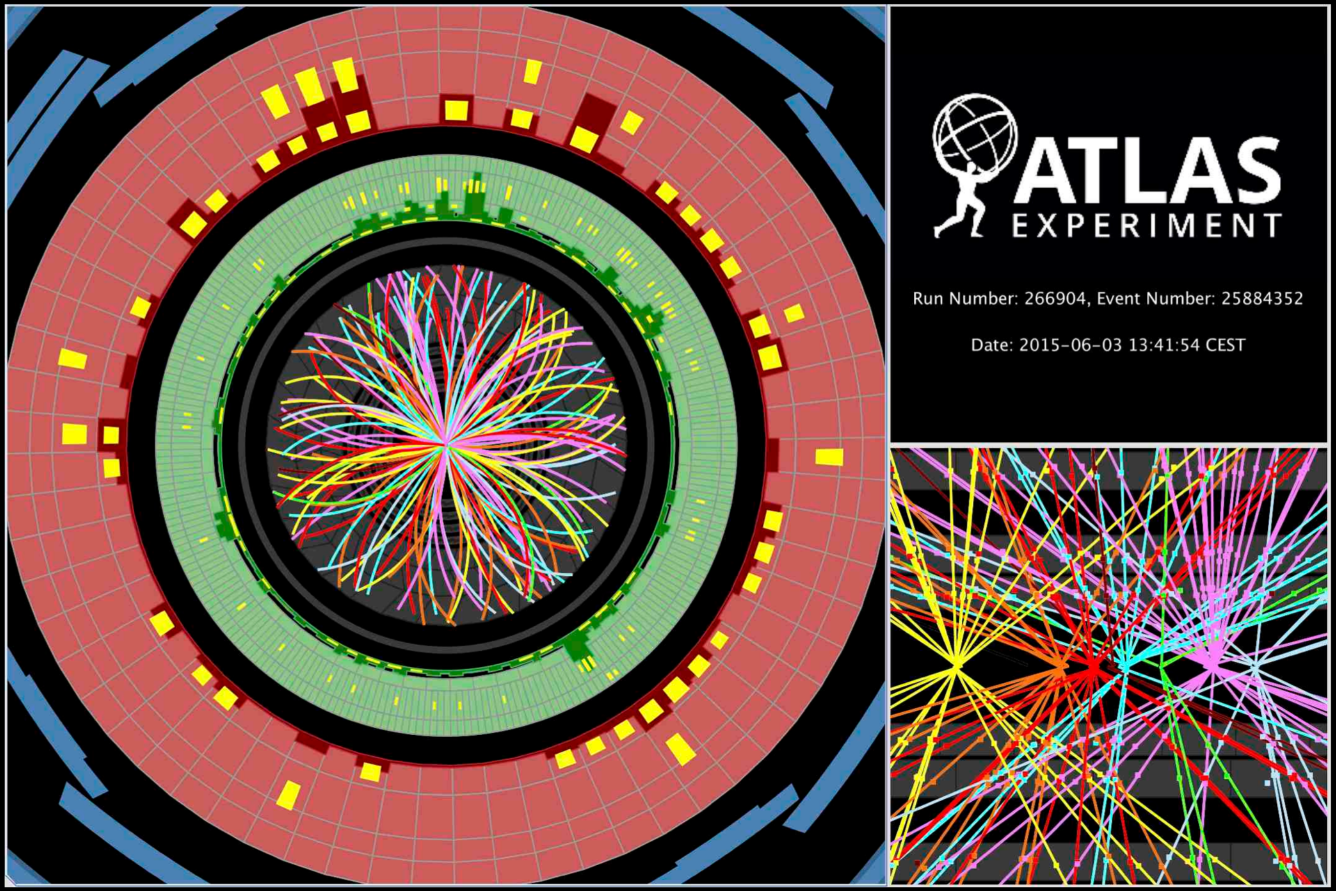


Collisions at the Large Hadron Collider



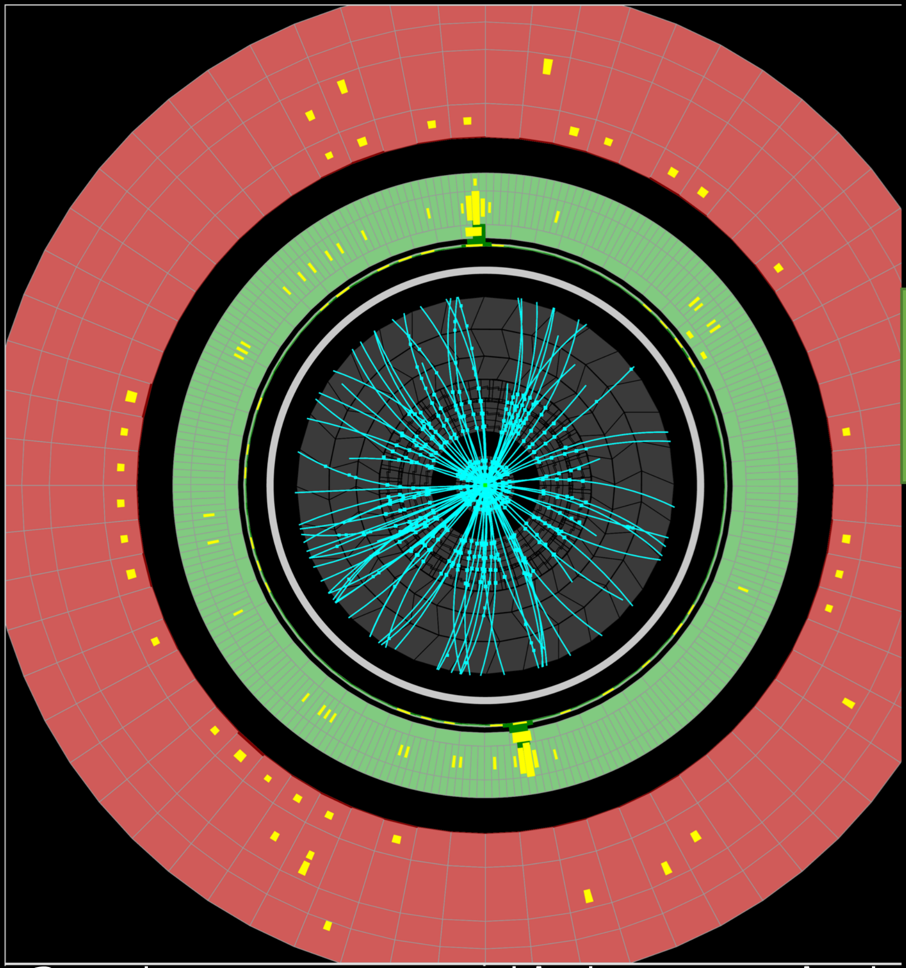
Bunch crossing every 25 ns… many collisions per bunch crossing

Most events look like this…



Event from LHC run-2

1 in >1000 billion events looks like this



Mass of the Higgs is reconstructed with photon energies

Higgs to 2 photon candidate with mass of 125 GeV

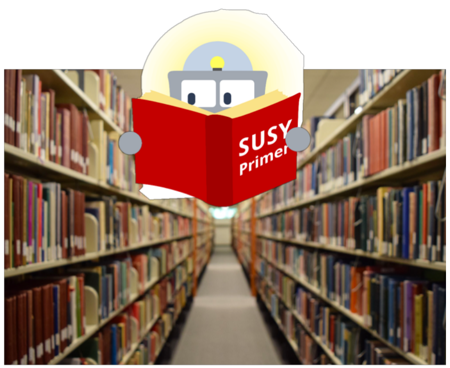
Analysis pipeline:

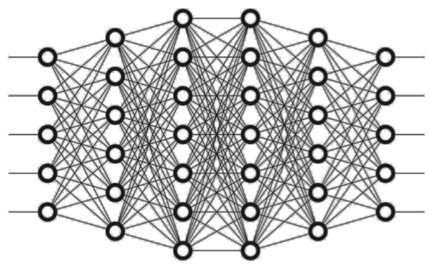
>days

Physical Model → Simulation → Experiment

Our goal:

< 1 msec





ML- tool scan
model-independently
(unsupervised)
for new physics

Accelerating simulations

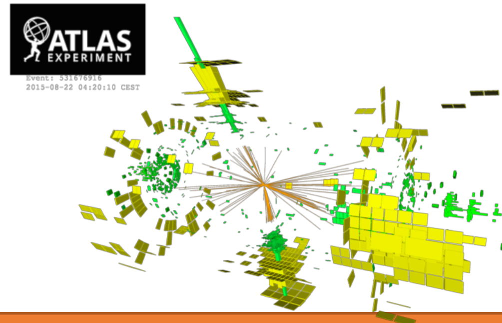
- **SUSY-AI** (www.susy-ai.org) : Good or bad model from 19 parameters
  - ➔ from hours to microseconds
- **DeepXs** : Calculating frequency of new physics events
  - ➔ from 20 min to microseconds

This project with surfsara:

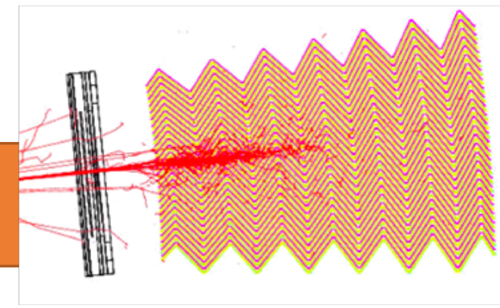
Generate and simulate full events !

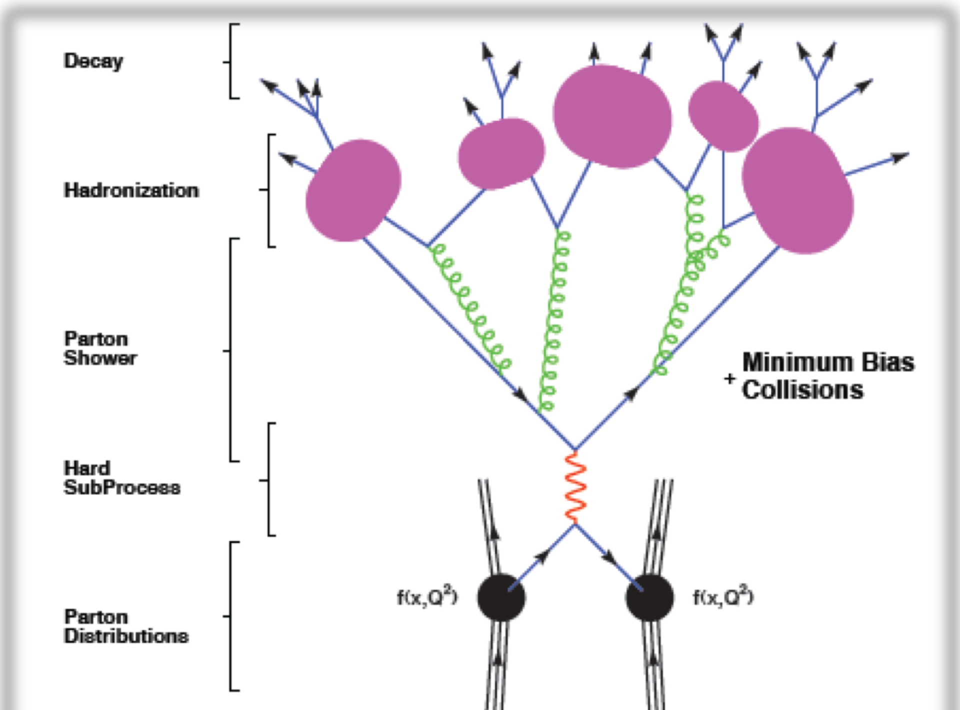
Simulation: Traditional

Energy and angles of reconstructed particles



Detector Simulator

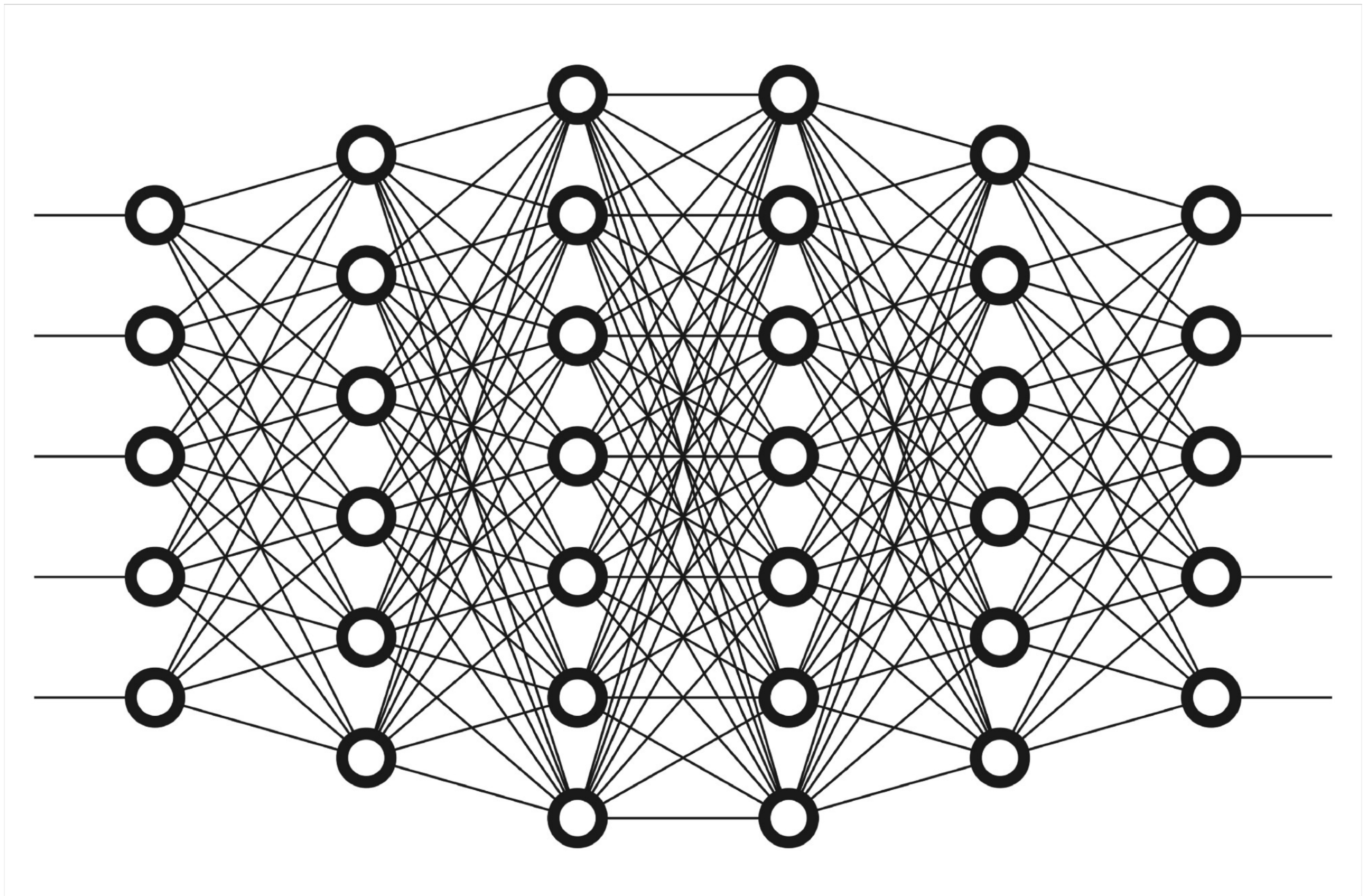




Input:
Random numbers

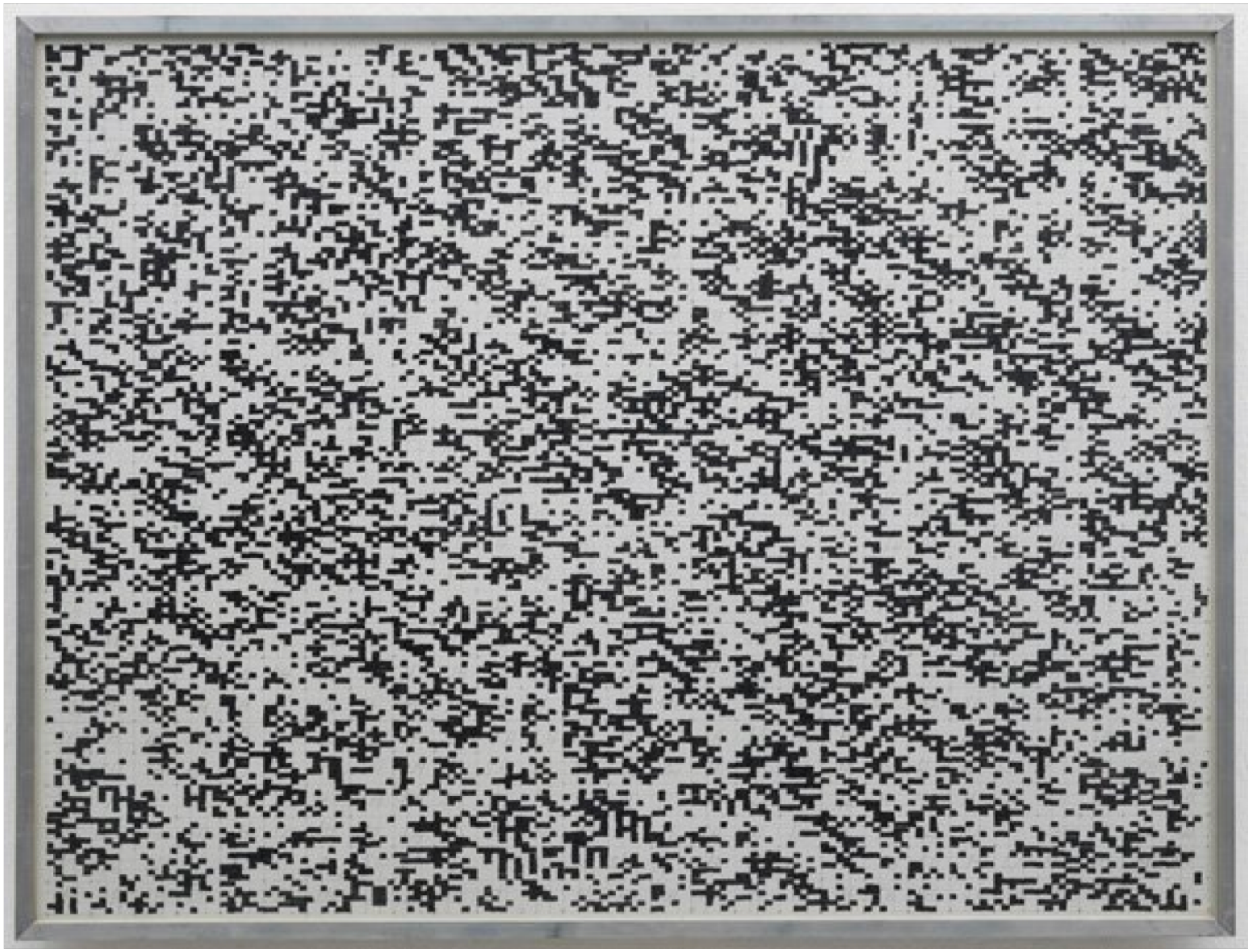
Simulation: Us

Energy and angles of reconstructed particles



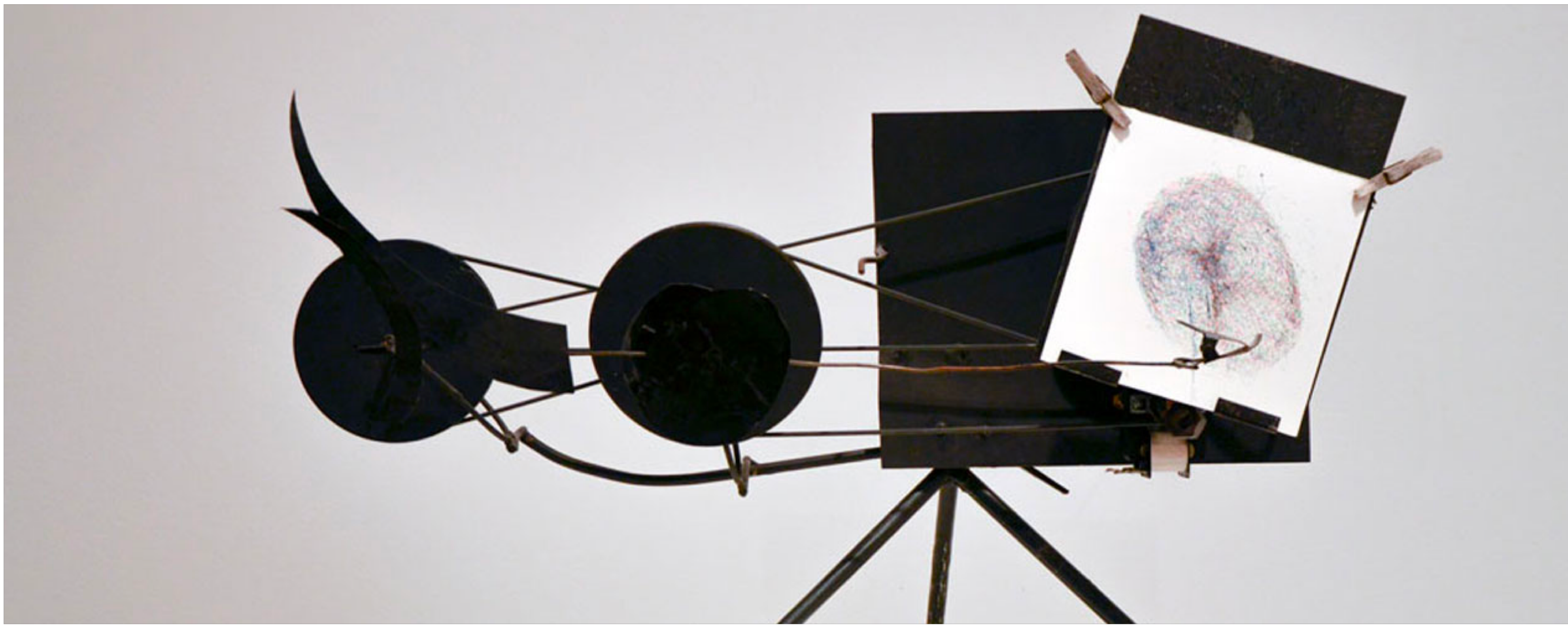
Input:
Random numbers

Random numbers…



Götz, Karl Otto: Statistisch-metrischer
Versuch 4:2:2:1, Entwurf Sommer 1959

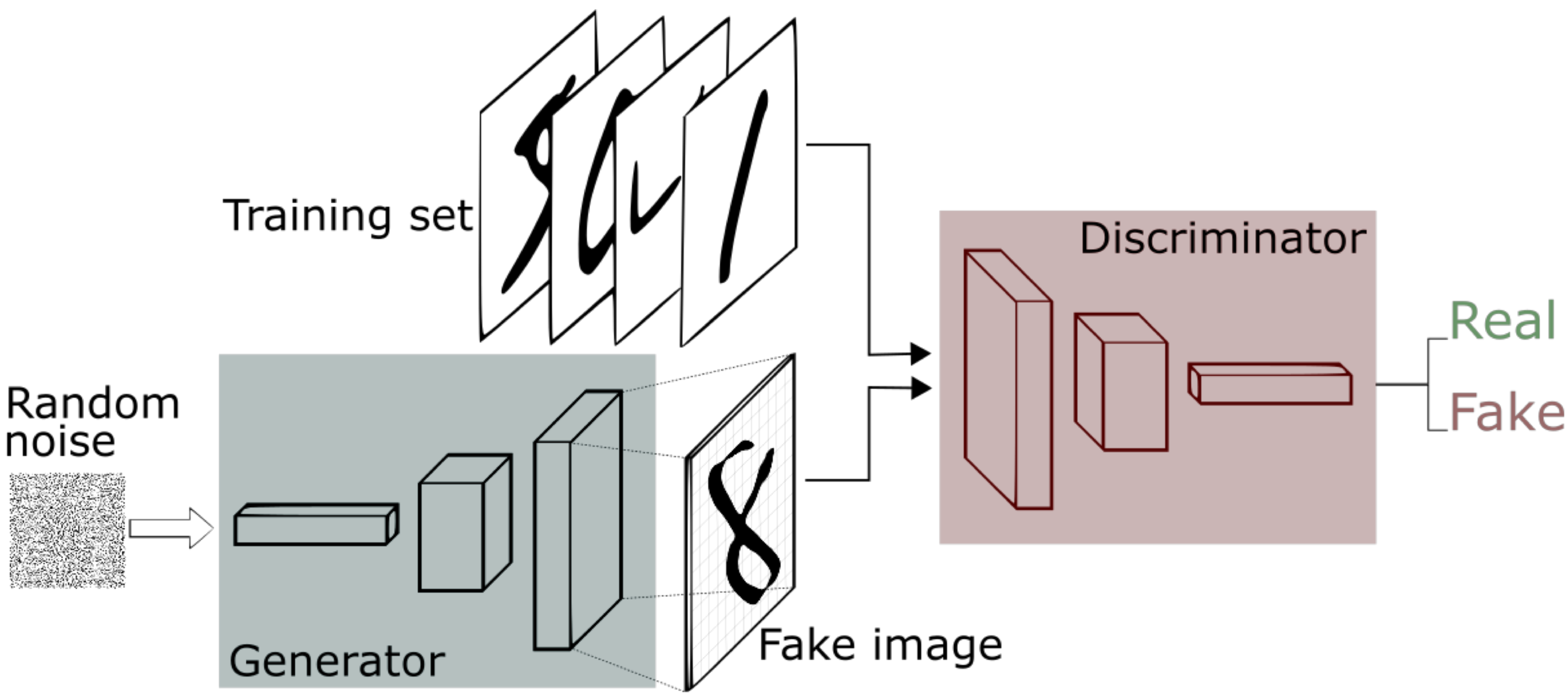
Random input → Art



Tinguely, Meta Matics

Network simulations ?

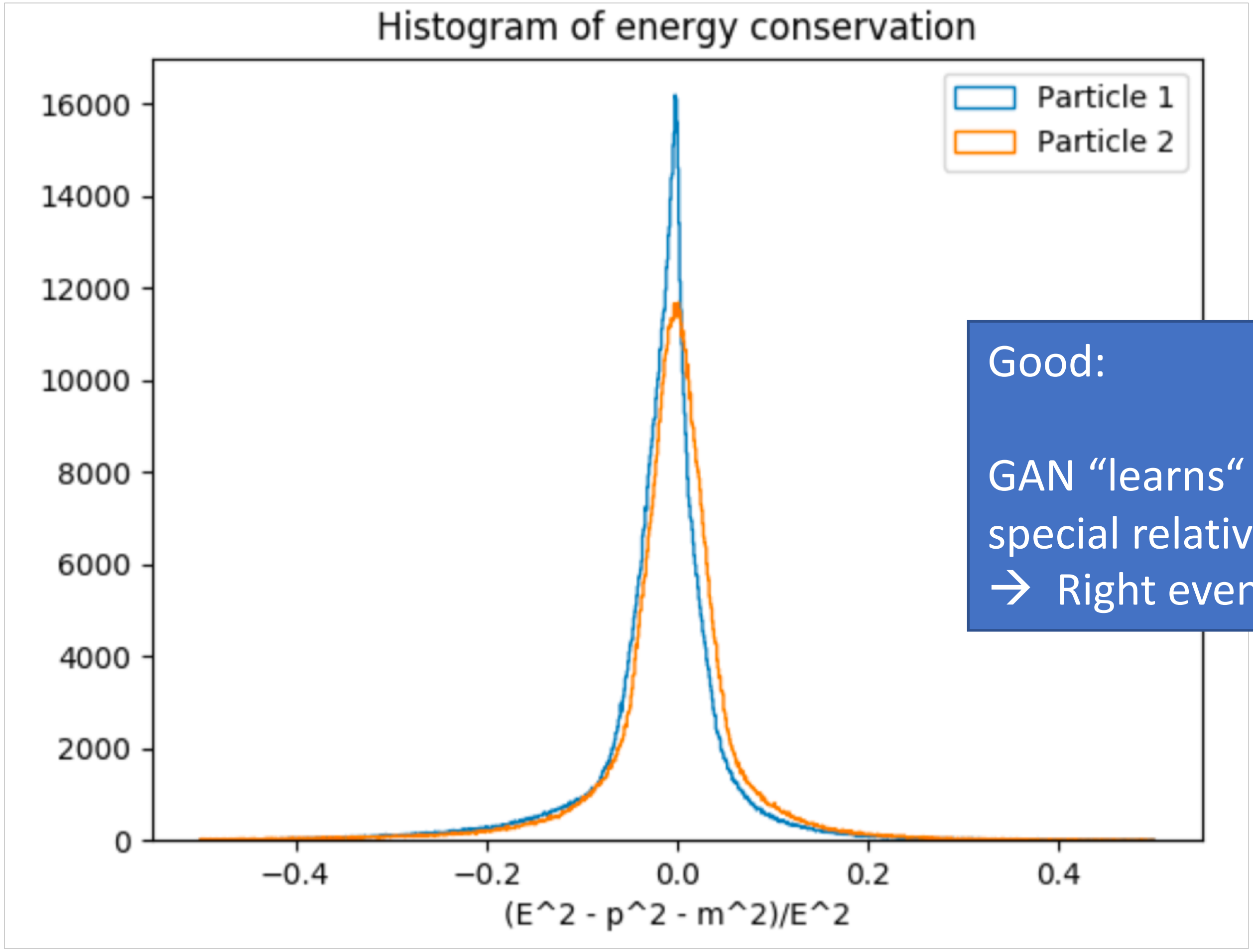
Generative Adversarial Networks state of the art:



1812.04948


source
destination
Coarse styles copied
Middle styles copied

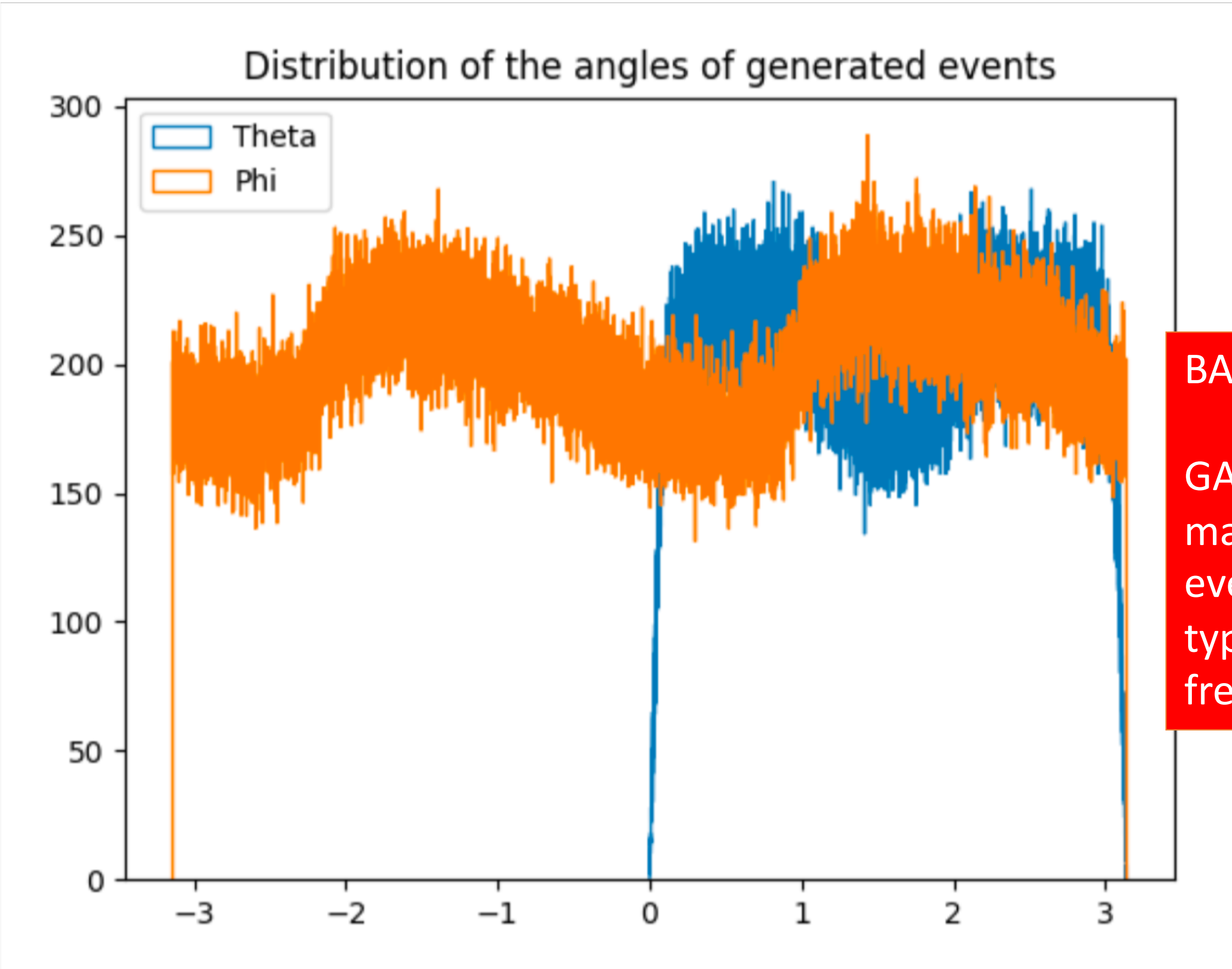
Distributions of Particle Collision “Events“ with GANs



Good:

GAN “learns“ special relativity !
→ Right events

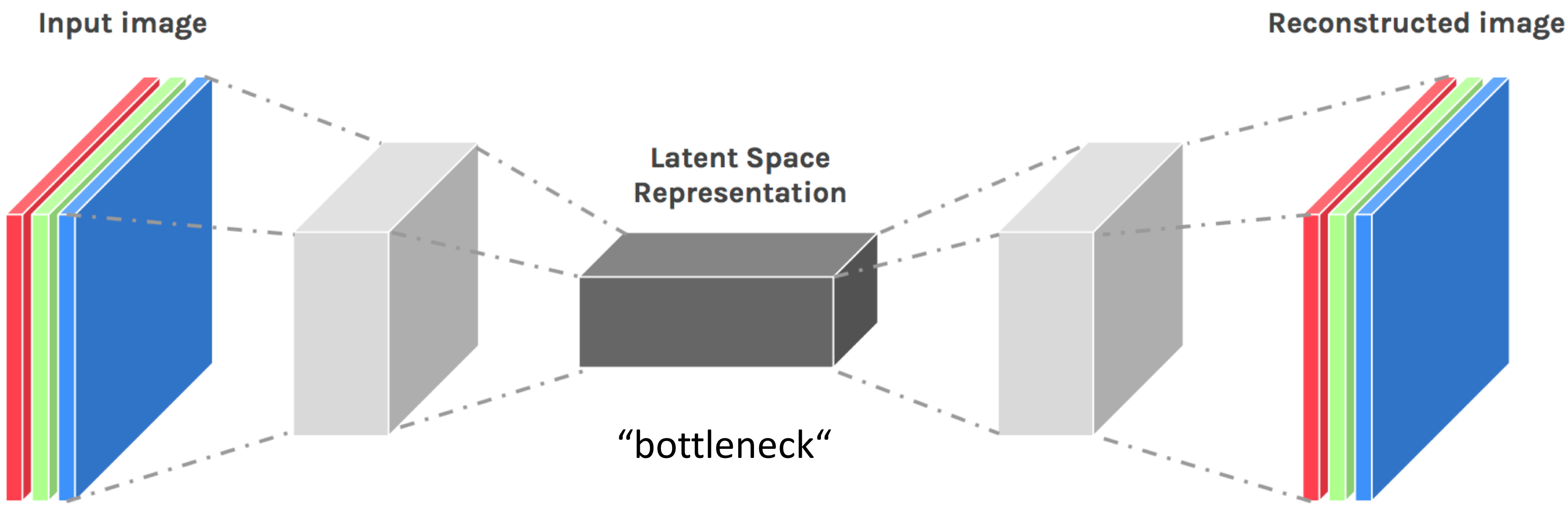
Distributions of Particle Collision "Events" with GANs



BAD:

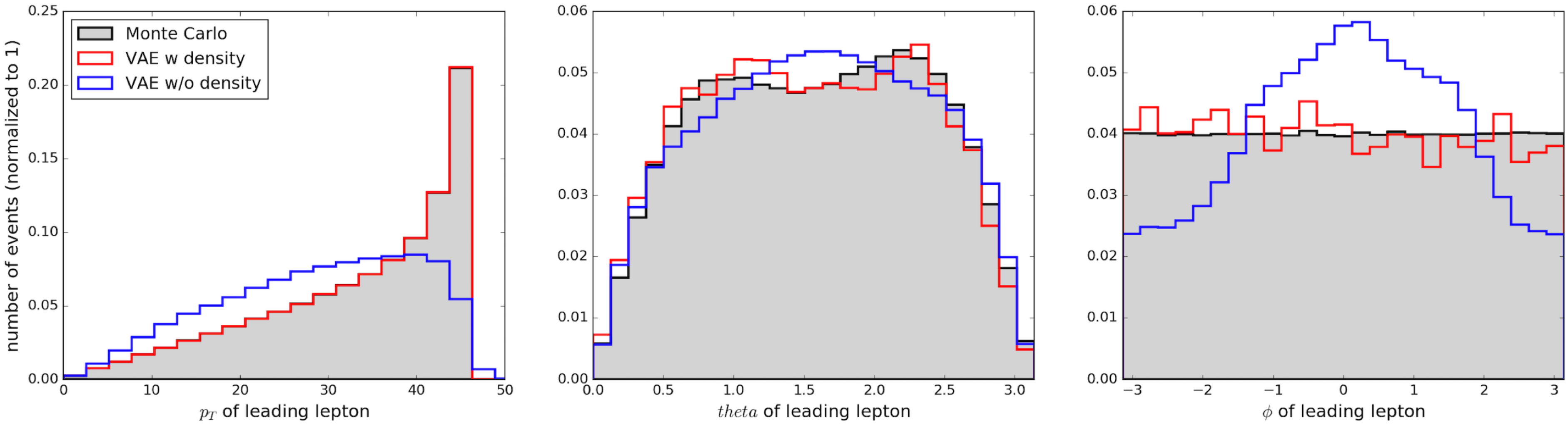
GAN does not make
events of different types with right frequencies !

Autoencoders



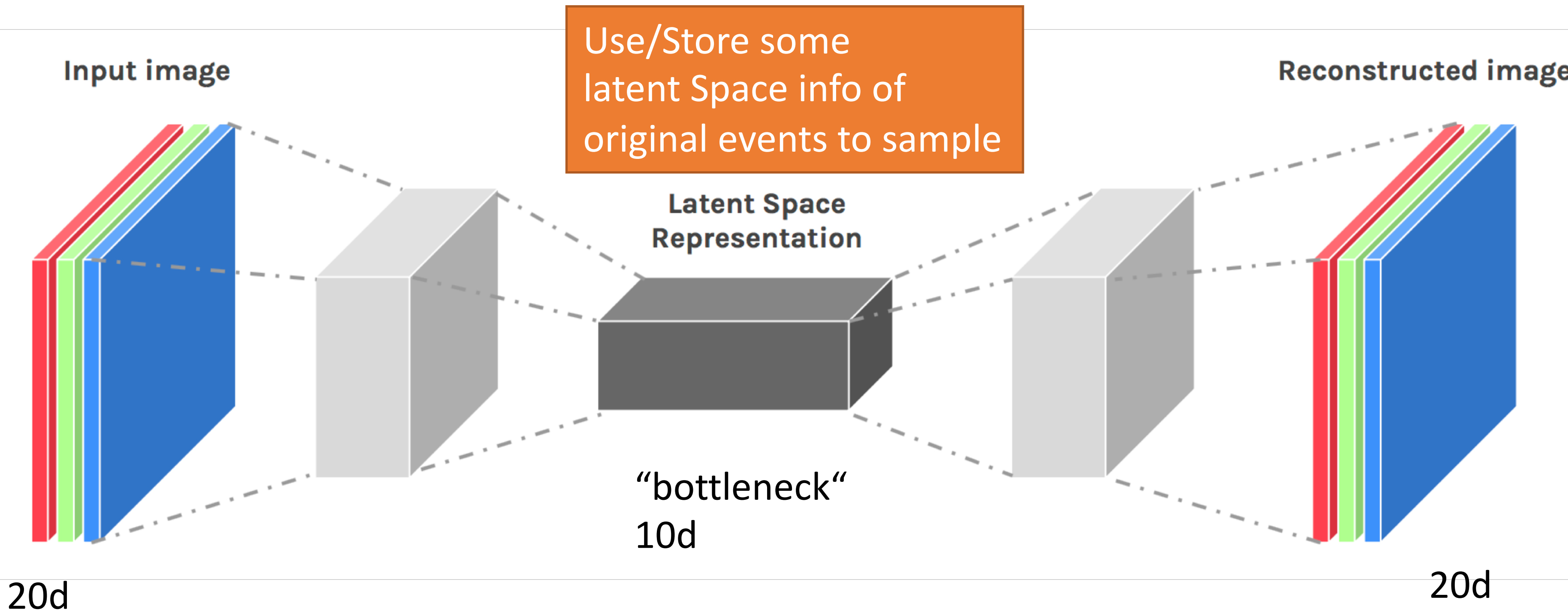
We actually use a better version:
„Dutch“ Autoencoder
(Variational Autoencoder by Dederik Kingma and Max Welling)

Distributions of Particle Collision “Events“ with variational autoencoders



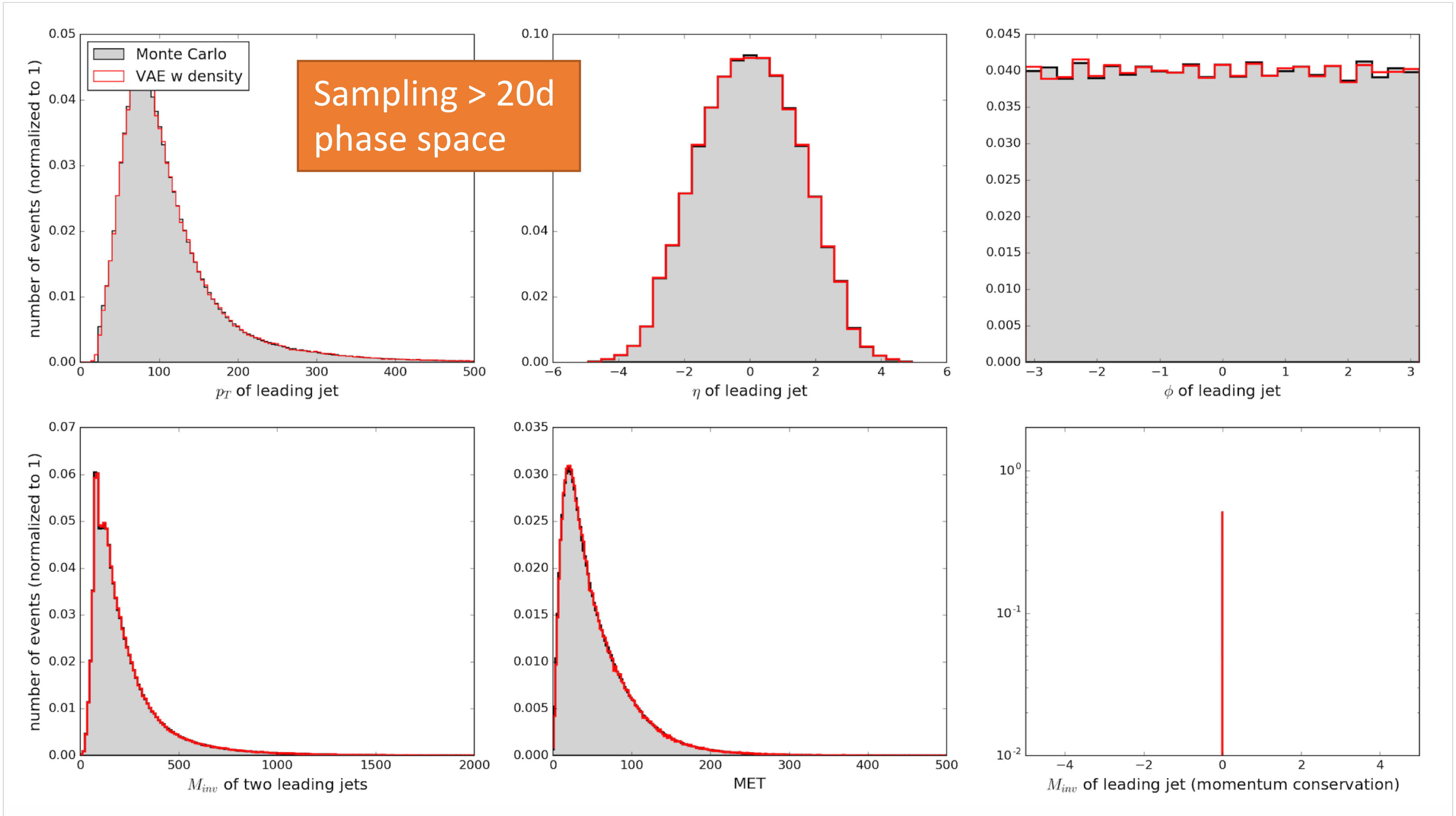
BAD:
Autoencoder typically does not make events of different types with right frequencies !

Autoencoder
(+ event info in latent space)



We actually use a better version:
„Dutch“ Autoencoder
(Variational Autoencoder by Dederik Kingma and Max Welling)

Distributions of Particle Collision "Events" with „density" variational autoencoders



Why is this useful ?

Can „store“ events in *lower dimensional* latent space and interpolate between them

→New events (by interpolation and compression/decompression)

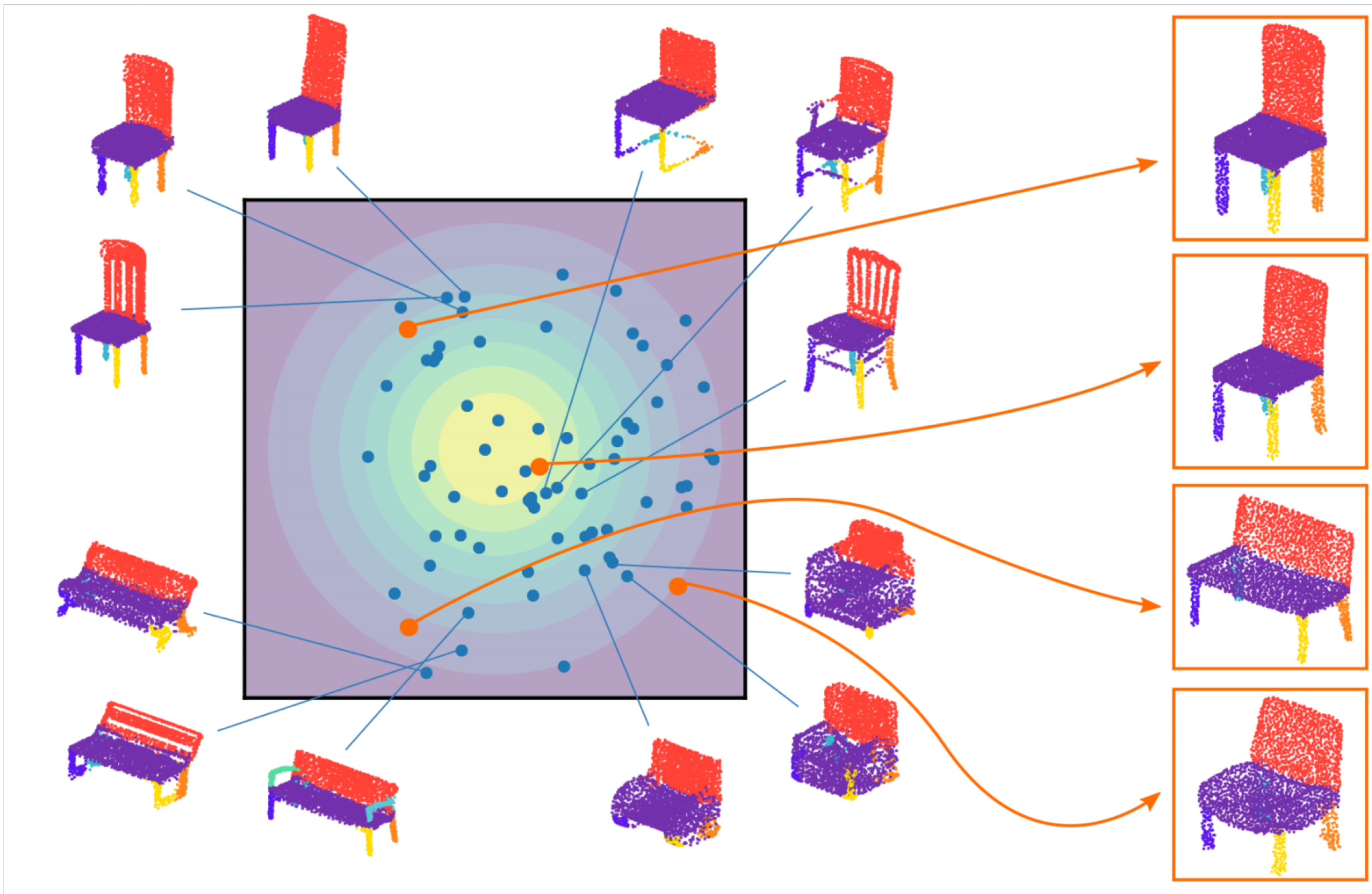
→New concepts (by interpolation)

→New models (by interpolation)

→Better random number sampling

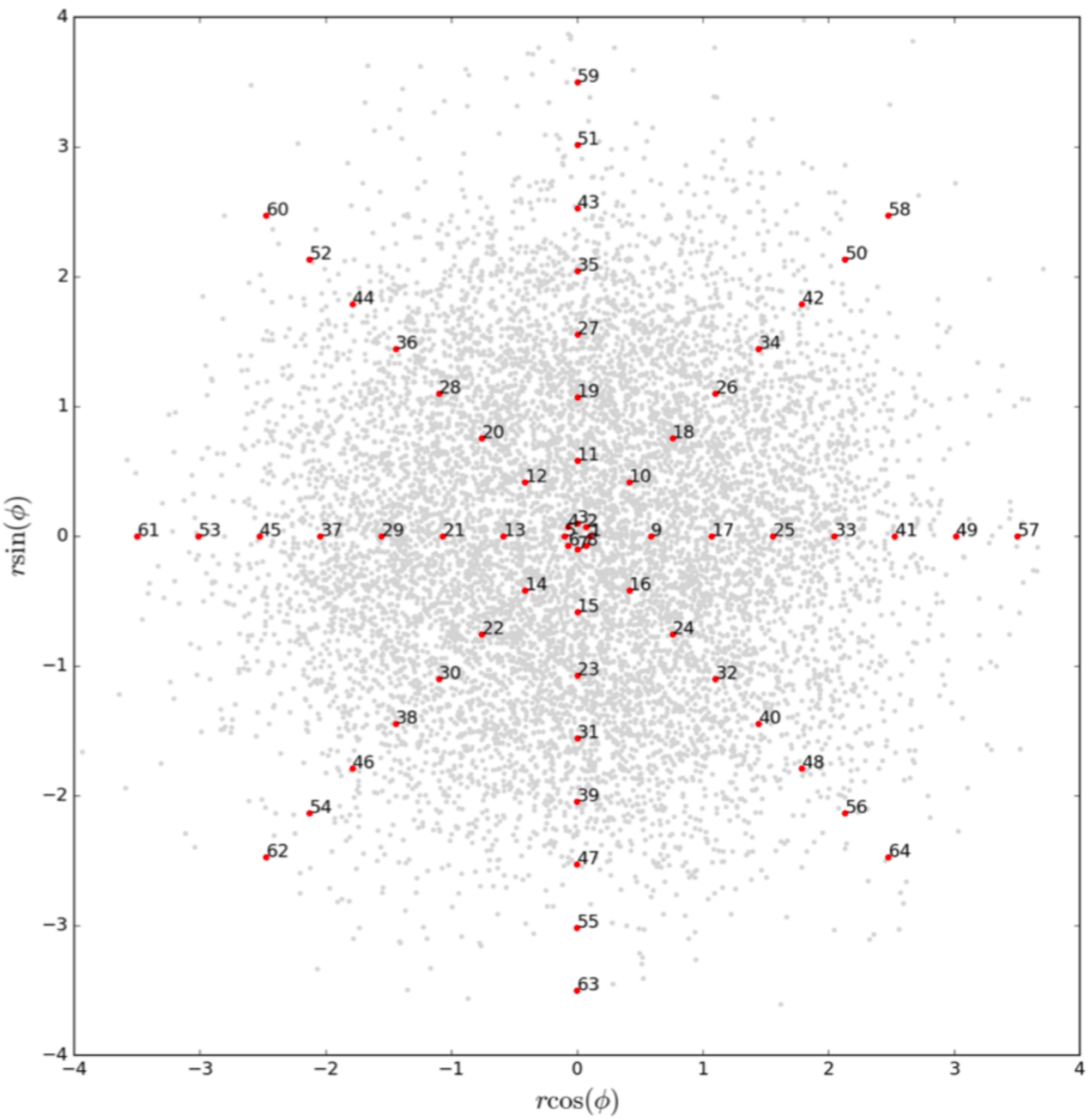
→Ultrafast (Million events per second compared to 1 event per minute …)

Concept of a latent space of sofas and chairs



arxiv 1610.07584

Top top Latent space PCA1 vs PCA2



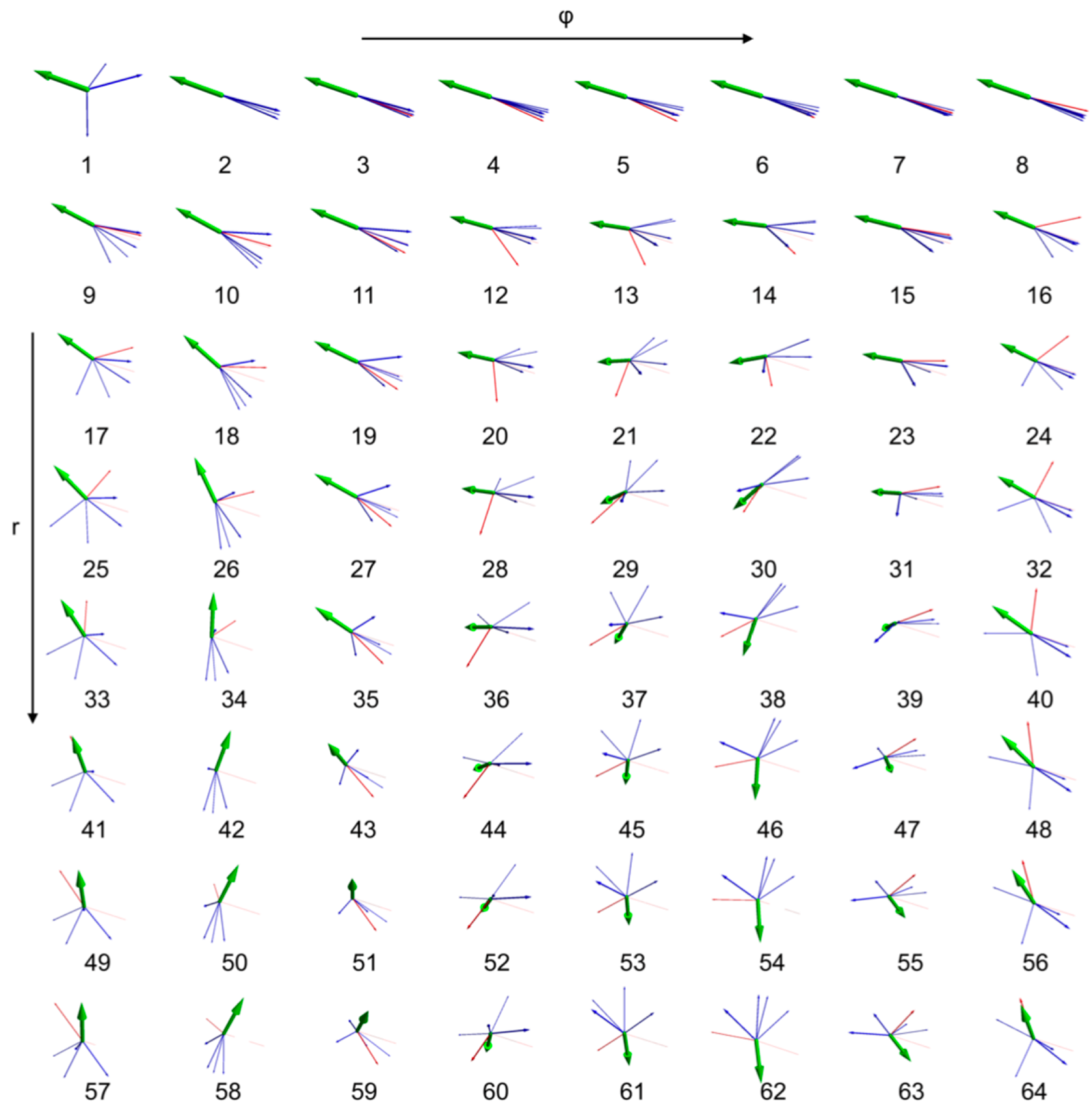
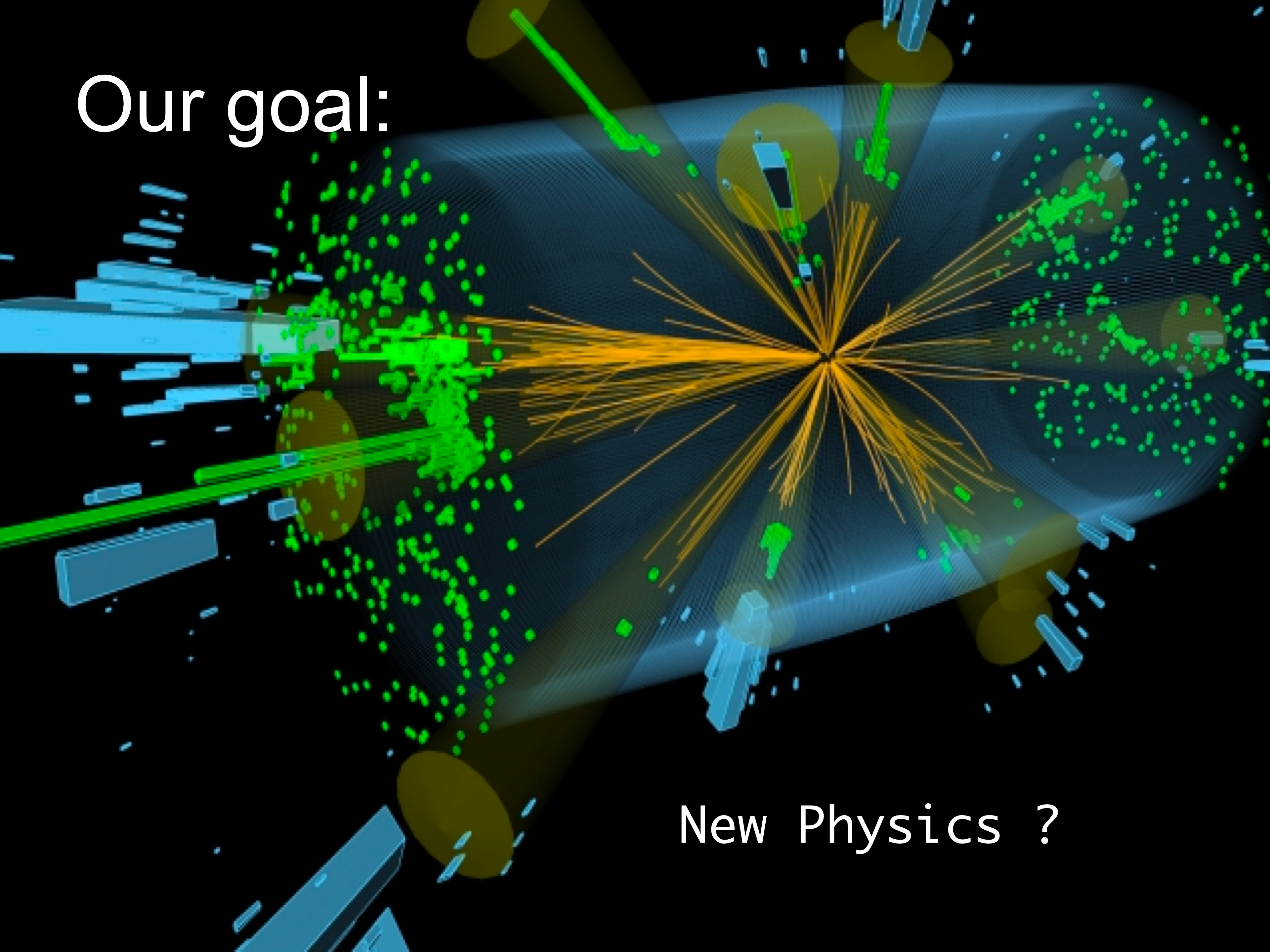
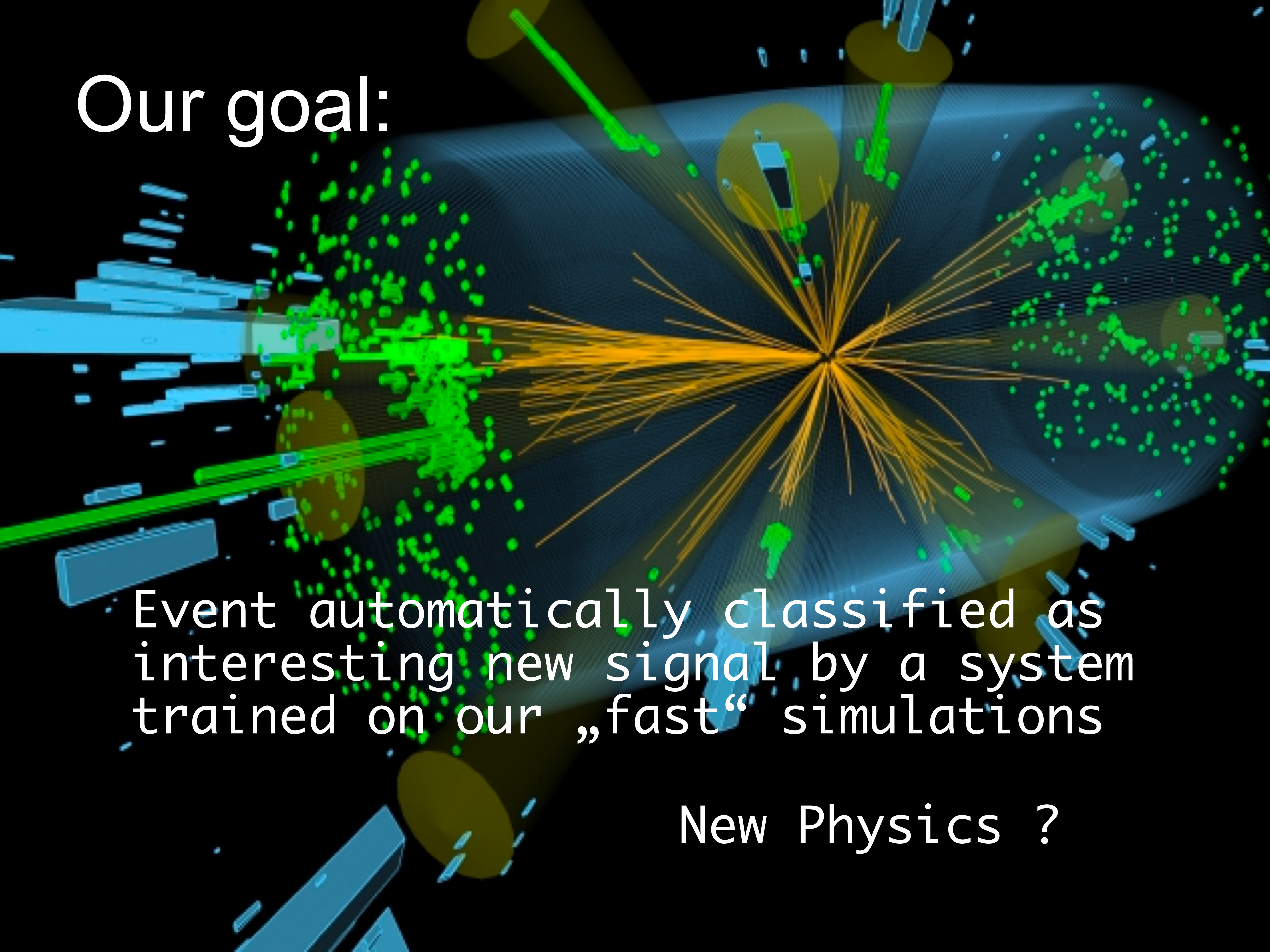


FIG. 7: Visualization of the first two components of a principal component analysis of encoded Monte Carlo events in latent space. This shows an 8×8 grid of event displays following the red dots in Figure 6. These 64 points chosen


Our goal:
New Physics ?

Our goal:



Event automatically classified as
interesting new signal by a system
trained on our „fast“ simulations

New Physics ?

Problem at the Large Hadron Collider is to find new over-densities in the data

compared with the SM expectation

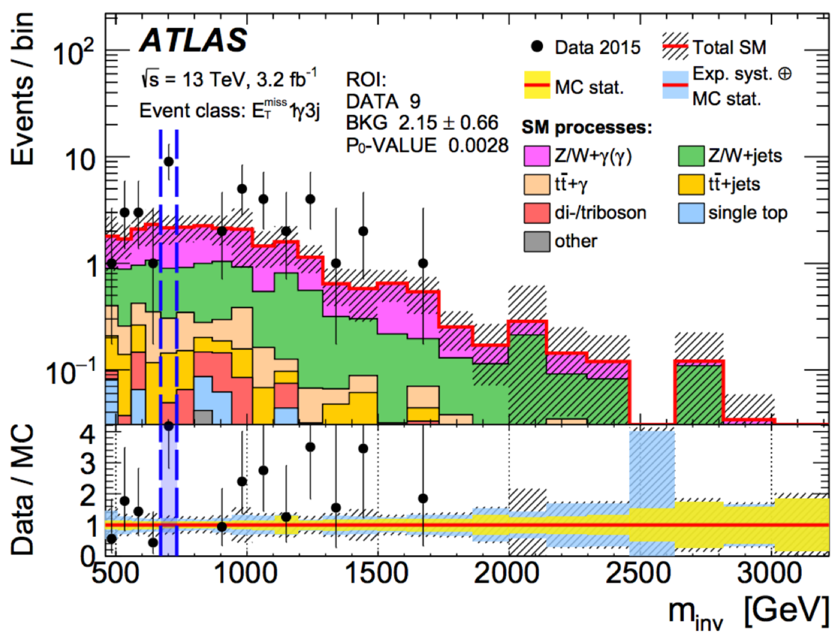
not necessary a new cluster or outliers…

… but maybe also outliers…

Our recent ATLAS approach

- Look everywhere for new overdensities
- Compare data to the SM using a test statistics and a scan algorithm

➔ e.g. General Search (on arxiv now: https://arxiv.org/abs/1807.07447)



Automatize:
>1600 distributions
>800 channels
>10^5 regions

Which quantity is optimal ?
How to determine background ?
How many hypothesis tests are optimal?