I falsari moderni

Maurizio Pierini

Potete credere ai vostri occhi?

- *TEST1: quali pittori hanno dipinto queste rivisitazioni della Gioconda?*

Potete credere ai vostri occhi?

- *TEST2: Chi credete che siano le persone ritratte in queste foto*

Potete credere ai vostri occhi?

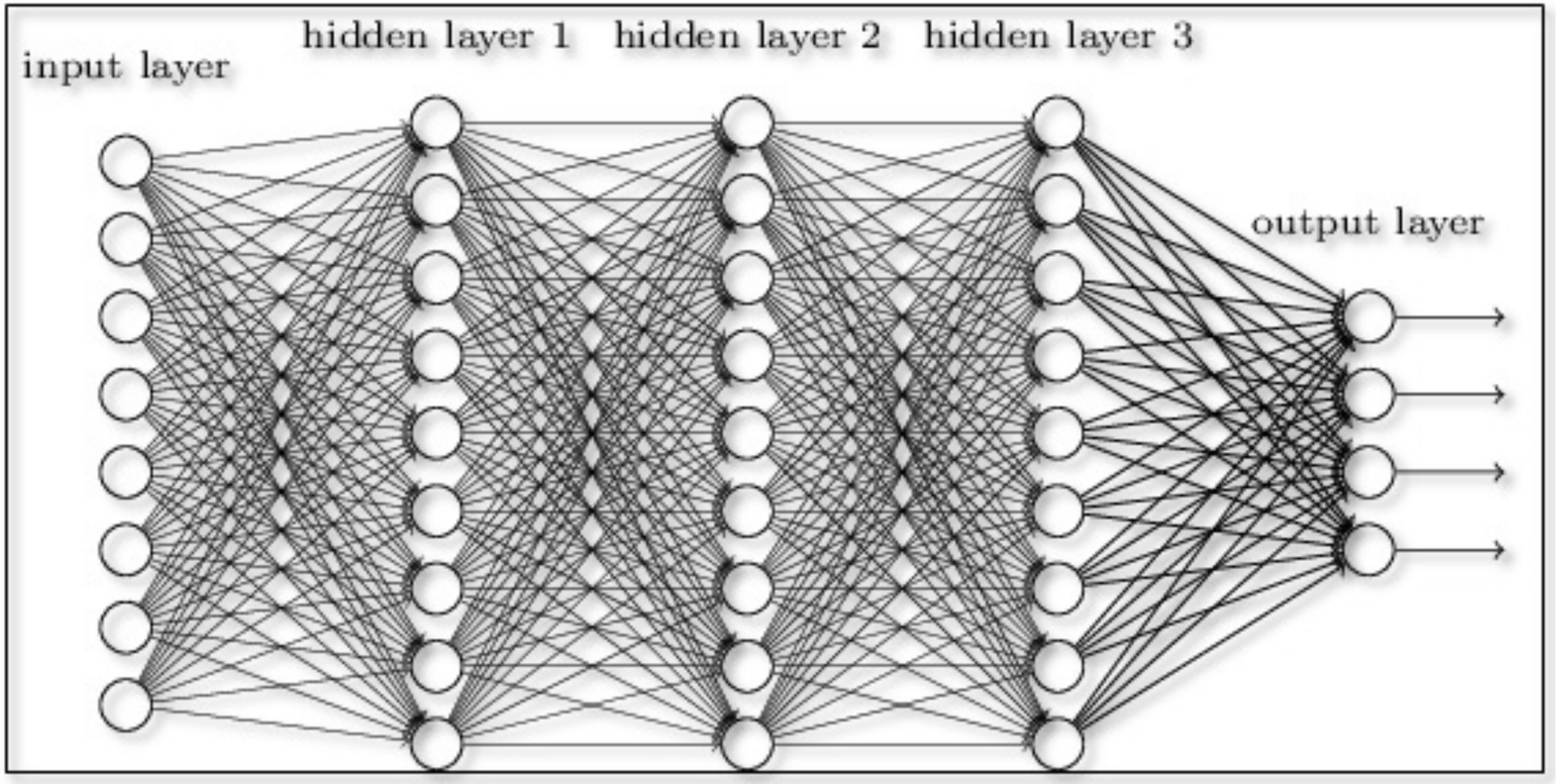
- *Queste foto sono state prodotte da una rete neurale, a partire da rumore bianco*
- *E' bastato allenare la rete ad imbrogliare un'esperto (virtuale) di celebrità*
- *La rete non ha la piu' pallida idea di cosa sia una celebrità.*

Potete credere ai vostri occhi?

- *Queste immagini sono state prodotte da un algoritmo basato su: due reti neurali, immagine della gioconda, un po' di quadri del vostro pittore preferito*

In questo talk

- *Cercherò di spiegarvi come funziona una rete neurale*
- *e come può essere utilizzata per creare immagini*
  - *ma anche suoni, video etc.*
- *In pratica, vi convincerò del fatto che in 10 anni (al massimo) non potrete piu' credere neanche a voi stessi*
- *Poi vi spiegherò cosa questo abbia a che fare con il CERN*



Deep Learning

European Research Council

Rete Neurale in due secondi

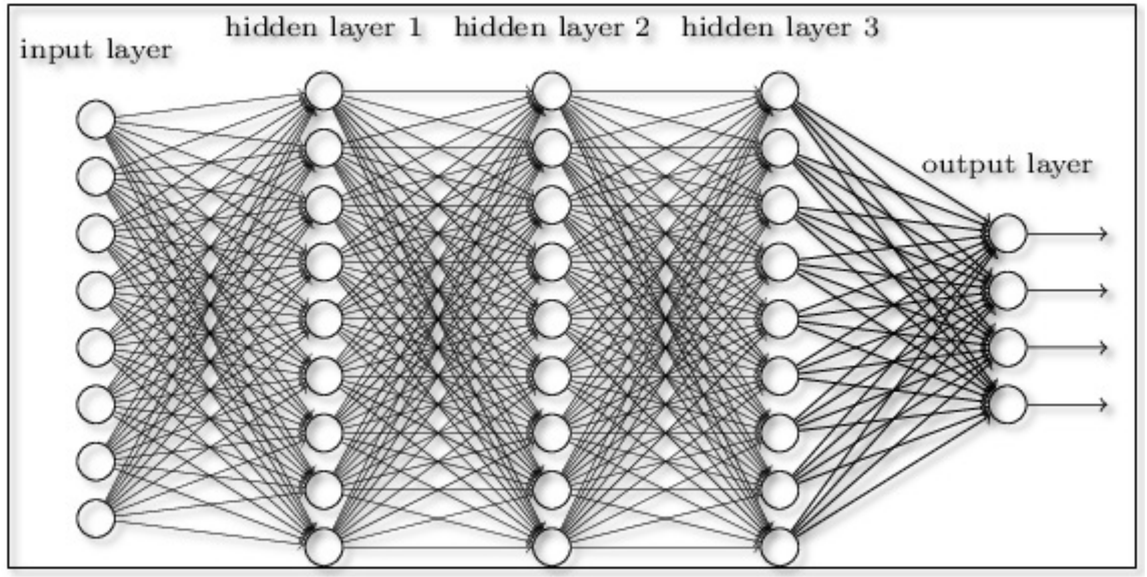
- Una rete neurale e' un algoritmo che cerca di "mimare" una serie di neuroni connessi

- Ogni nodo

  - … riceve informazioni dai nodi di sinistra a cui e' connesso

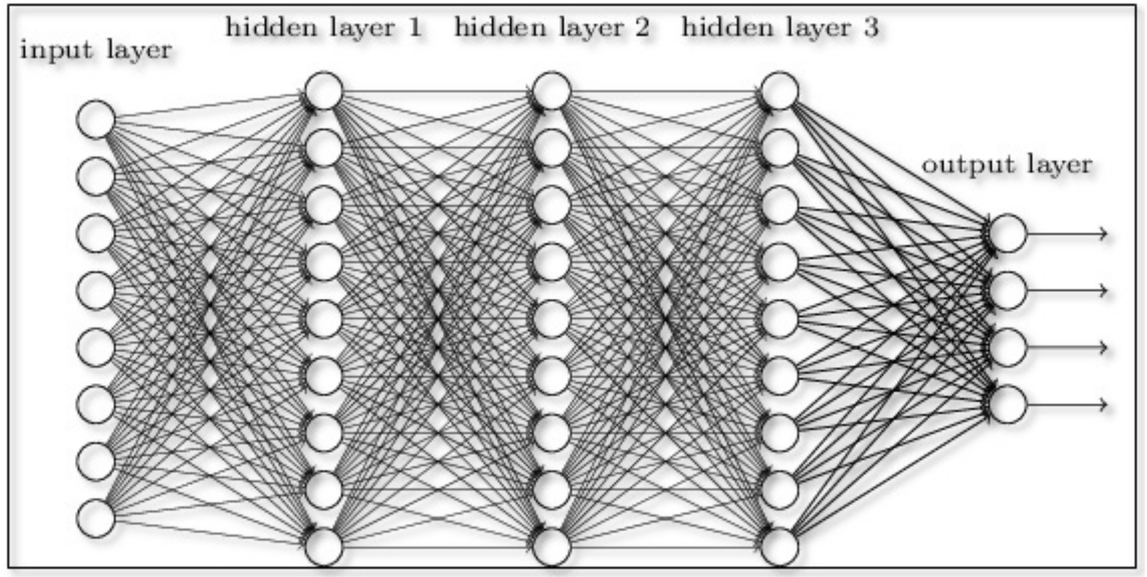
  - … li processa

  - … li trasmette ai nodi di destra a cui e' collegato



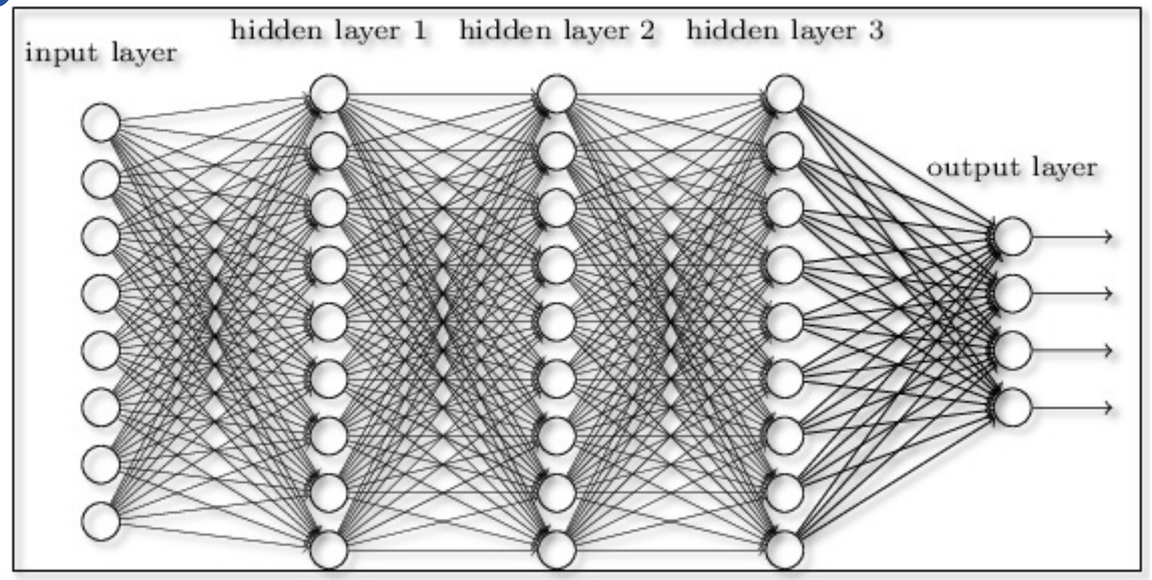
L'Input

- I dati di input entrano nella rete da sinistra
- In genere, si tratta di numeri, risultato della digitizzazione di
  - immagini
  - suoni
  - etc…



Cosa fa un nodo

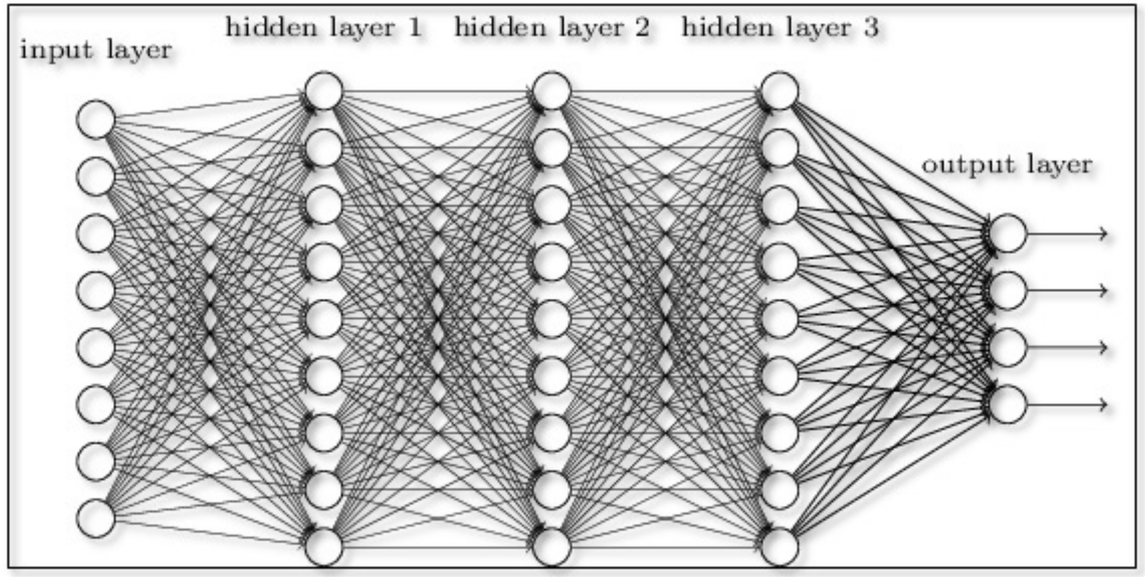
- **Moltiplica l'input di ogni nodo di sinistra per un peso w**
- Somma tutti i valori
- aggiunge un "bias" b
- applica una funzione di attivazione



$$w_{ij}x_j$$

Cosa fa un nodo

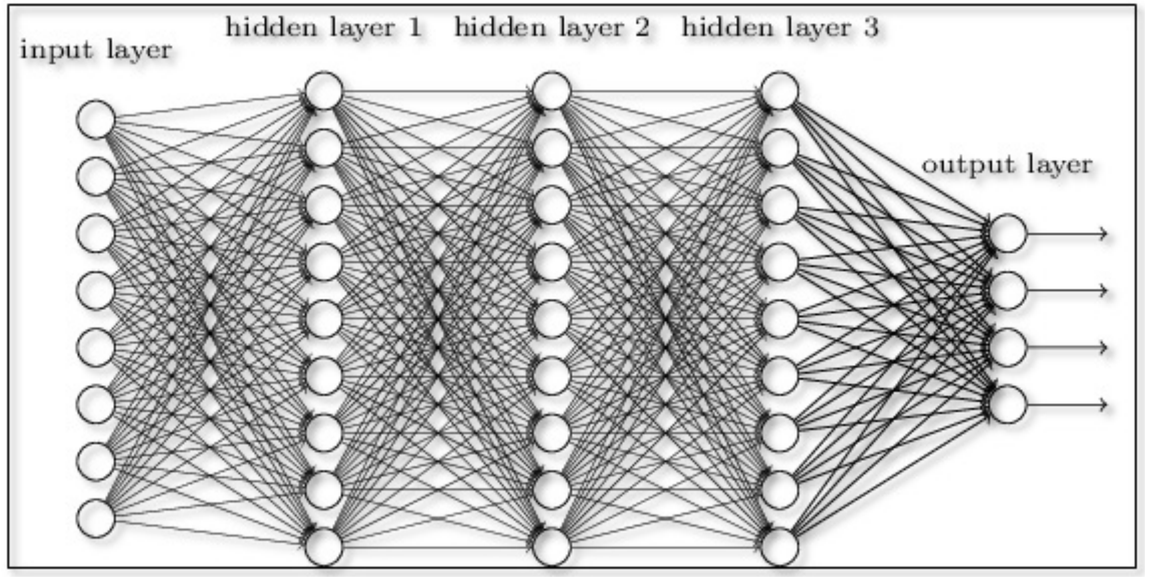
- Moltiplica l'input di ogni nodo di sinistra per un peso w
- **Somma tutti i valori**
- aggiunge un "bias" b
- applica una funzione di attivazione



$$\Sigma_j w_{ij} x_j$$

Cosa fa un nodo

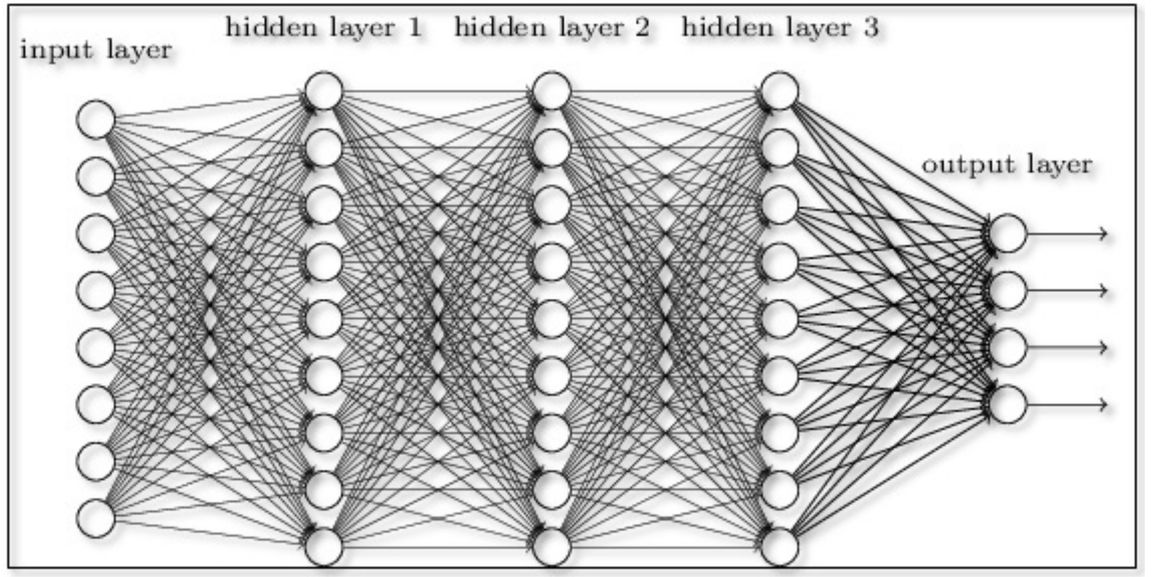
- Moltiplica l'input di ogni nodo di sinistra per un peso w
- Somma tutti i valori
- **aggiunge un "bias" b**
- applica una funzione di attivazione

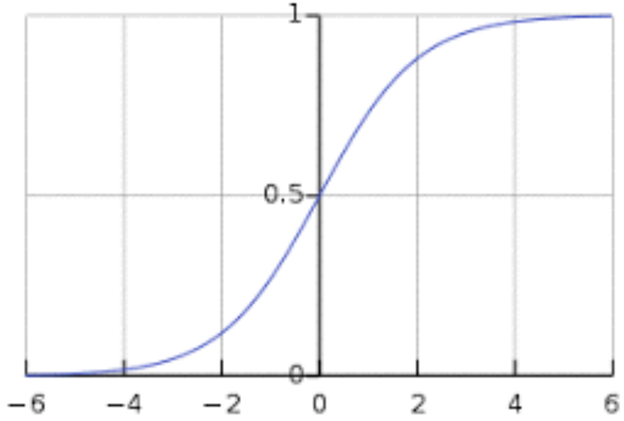


$$\Sigma_j w_{ij} x_j + b_i$$

Cosa fa un nodo

- Moltiplica l'input di ogni nodo di sinistra per un peso w
- Somma tutti i valori
- aggiunge un "bias" b
- **applica una funzione di attivazione**

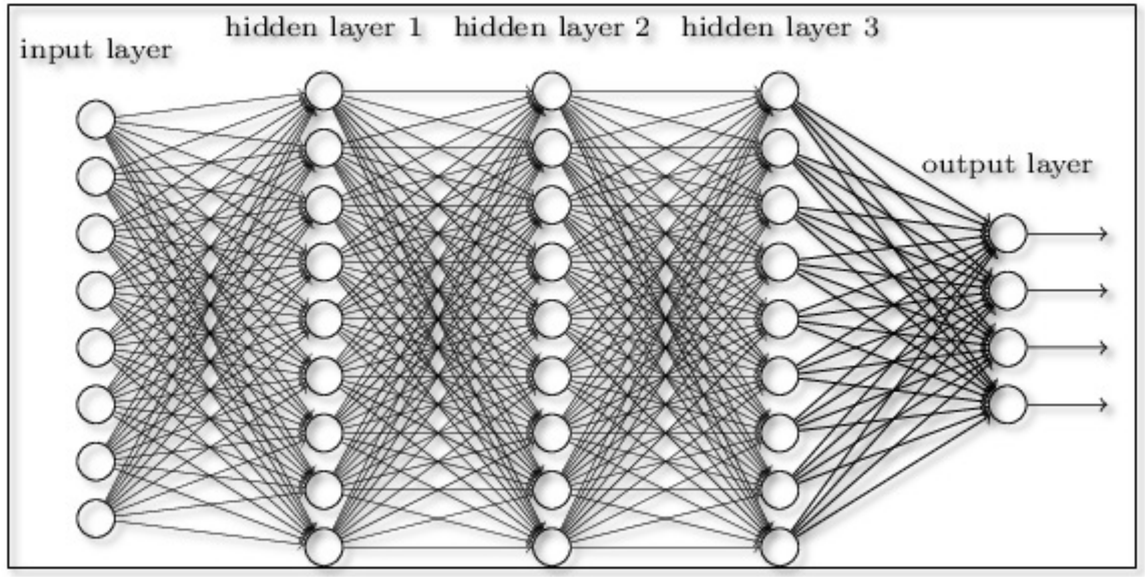




$$y_i = f(\Sigma_j w_{ij} x_j + b_i)$$

Cosa fa la rete

- *La rete processa i dati di input facendo passare l'informazione attraverso le connessioni*
  - *Ogni "layer" di nodi modifica l'informazione d'ingresso e la passa a quella d'uscita*
- *Il risultato finale e' una funzione matematica da **N** input a **K** output*

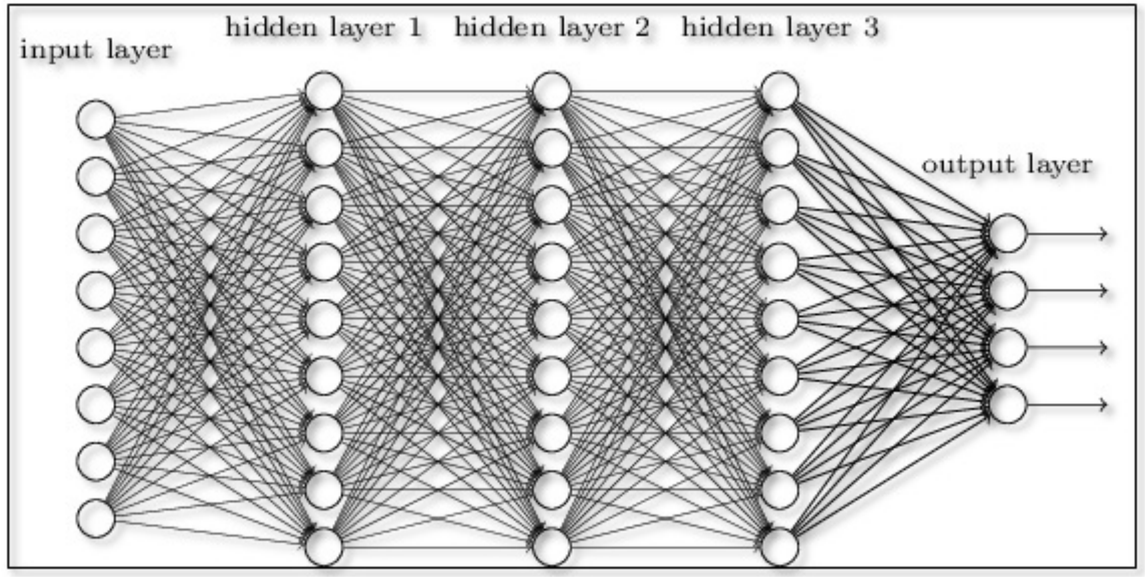


$$y_j = \Sigma_l w_{jl}^{(3)} (\Sigma_k w_{lk}^{(2)} (\Sigma_i w_{ki}^{(1)} x_i + b_k^{(k)}) + b_l^{(2)}) + b_j^{(3)}$$

- *E quindi?*

Cosa fa la rete

- *Si puo' dimostrare che, per reti sufficientemente complesse, ogni trasformazione dei dati di input in un output desiderato puo' essere rappresentato da una rete neurale*



$$y_j = \Sigma_l w_{jl}^{(3)} (\Sigma_k w_{lk}^{(2)} (\Sigma_i w_{ki}^{(1)} x_i + b_k^{(1)}) + b_l^{(2)}) + b_j^{(3)}$$

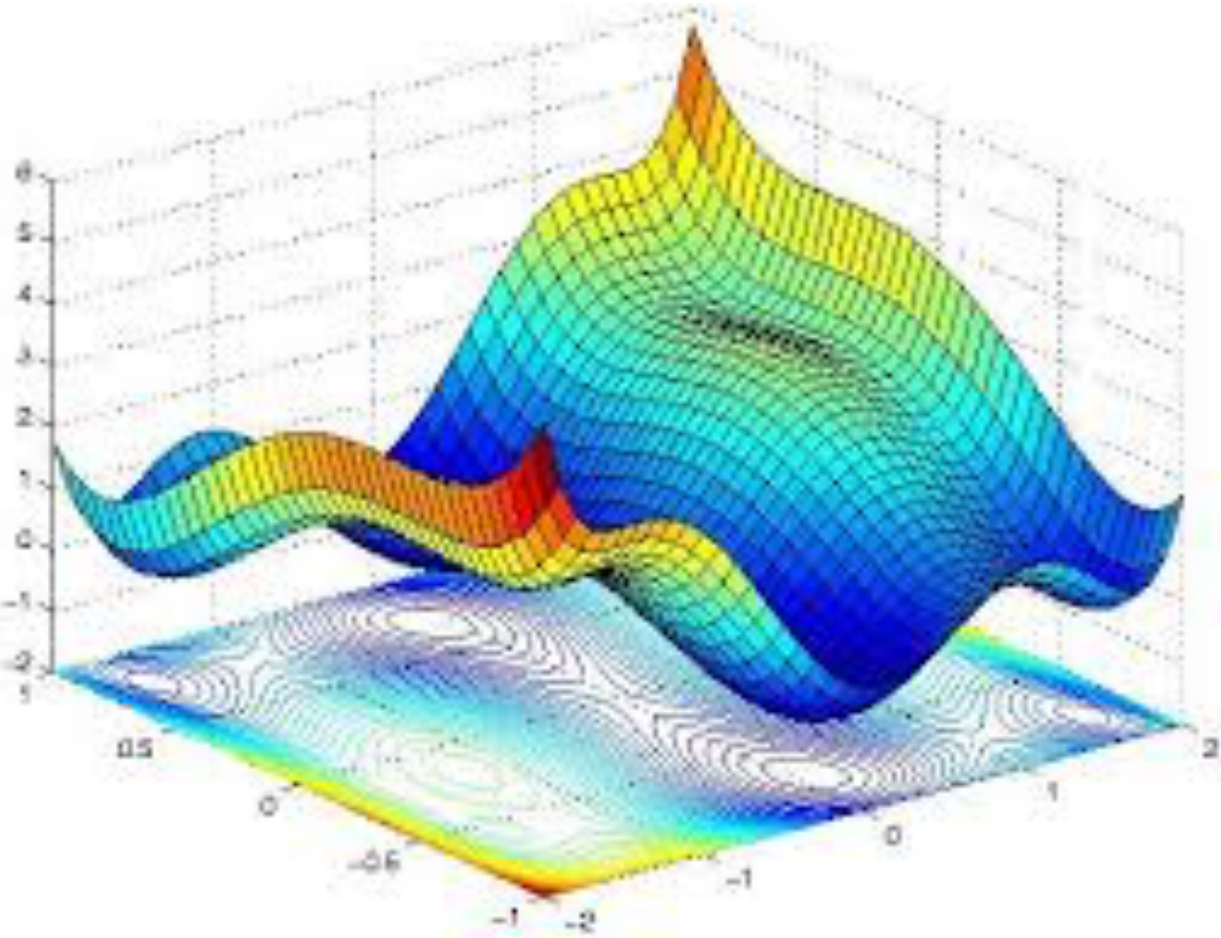
- *E quindi?*

Target

- *Il target e' una rappresentazione numerica della risposta che la rete impari a darvi*
  - *ad esempio, se volete distinguere cani da gatti in una serie di foto, potete associare gatto a 0 e cane ad 1*
  - *dovete poi fornire alla rete un campione di allenamento (foto che qualcuno ha guardato ed associato a cani o gatti)*
  - *la rete si allenerà a riprodurre il target, dandovi numeri vicini a 0 (1) per immagini che crede essere piu' simili a gatti (cani)*
  - *l'allenamento consiste nell'aggiustare le **w** e le **b** in modo da minimizzare l'errore*

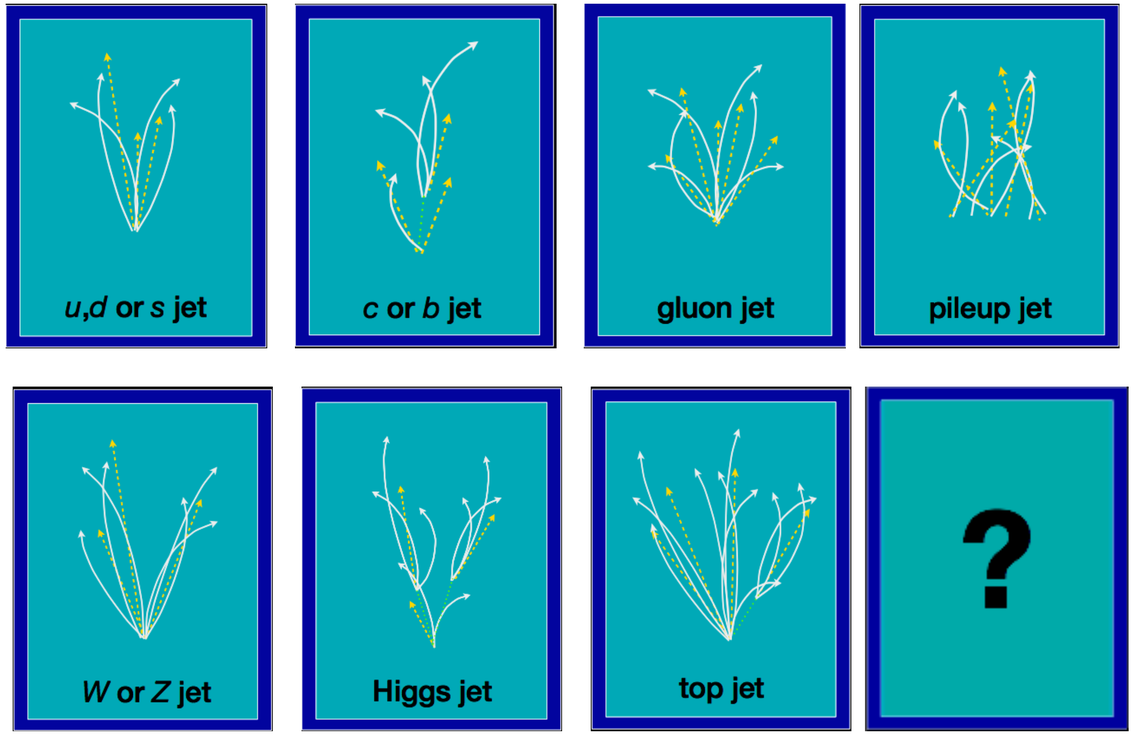
Loss Function

- *La funzione di loss quantifica quanto la risposta della rete sia sbagliata*
- *Minimizzare la funzione di loss significa migliorare la rete*
- *La scelta della loss function dipende dal problema che si sta trattando*
- *L'allenamento consiste nel trovare i valori di **w** e **b** che minimizzano la loss*

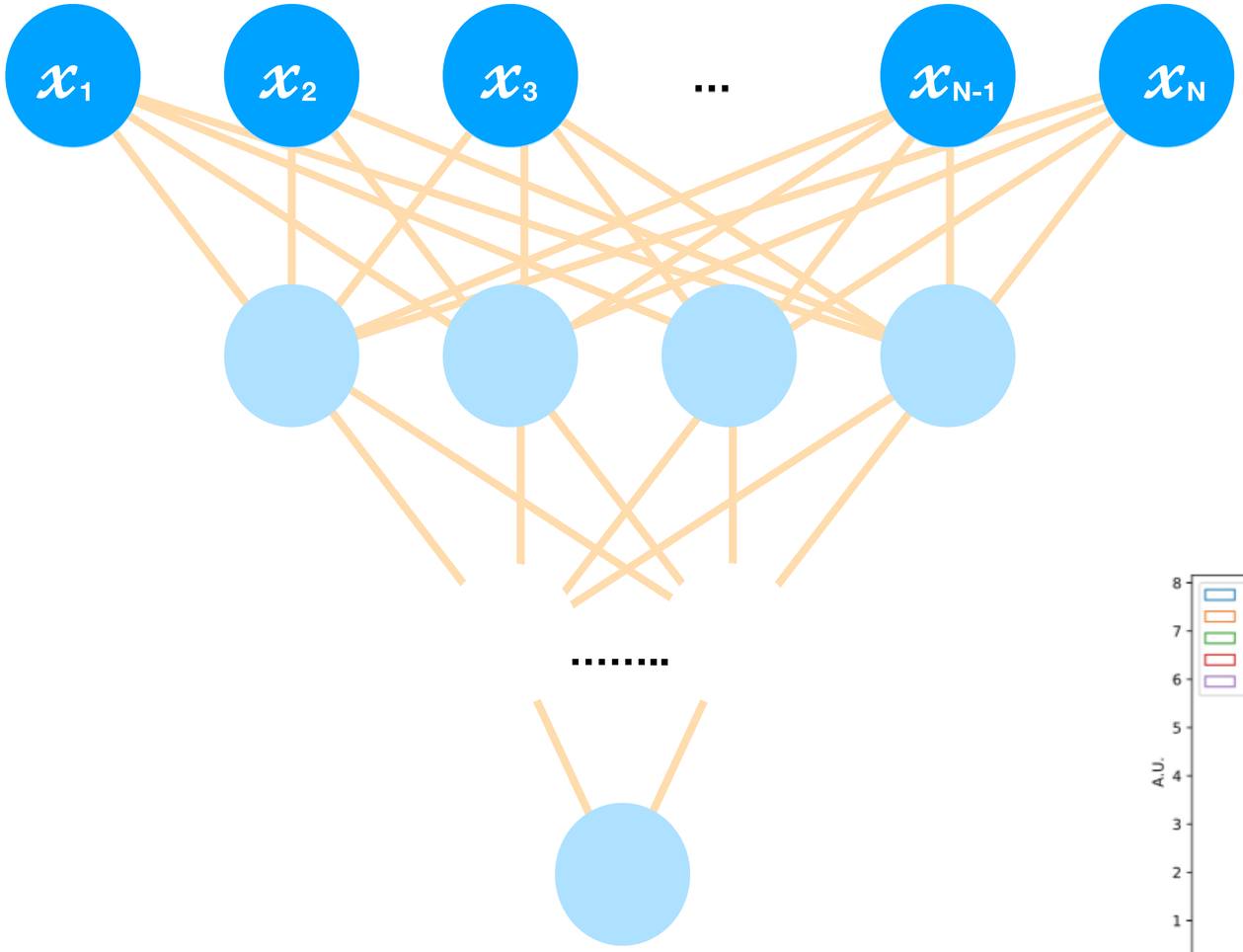


Example: jet tagging

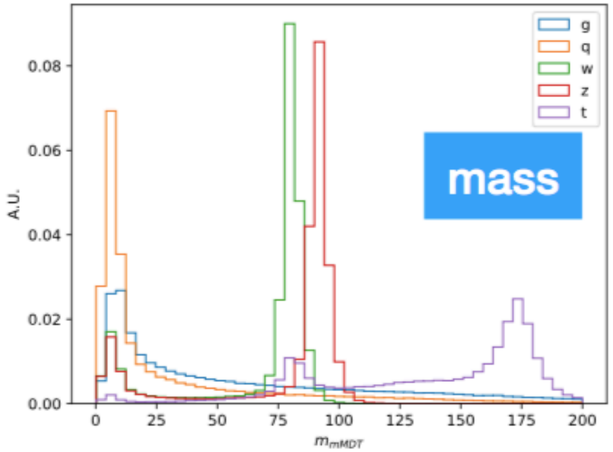
- *You have a jet at LHC: spray of hadrons coming from a "shower" initiated by a fundamental particle of some kind (quark, gluon, W/Z/H bosons, top quark)*
- *You have a set of jet features whose distribution depends on the nature of the initial particle*
- *You can train a network to start from the values of these quantities and guess the nature of your jet*
- *To do this you need a sample for which you know the answer*

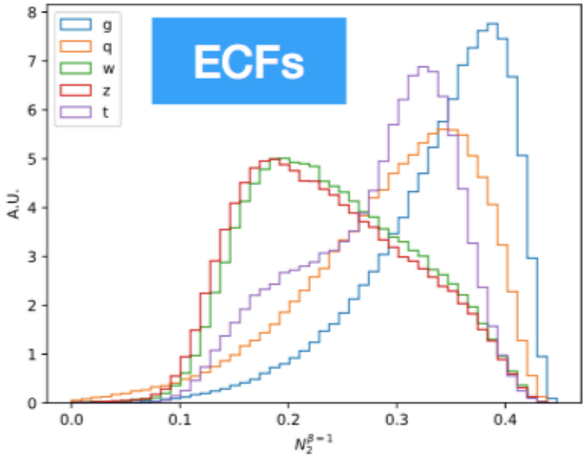


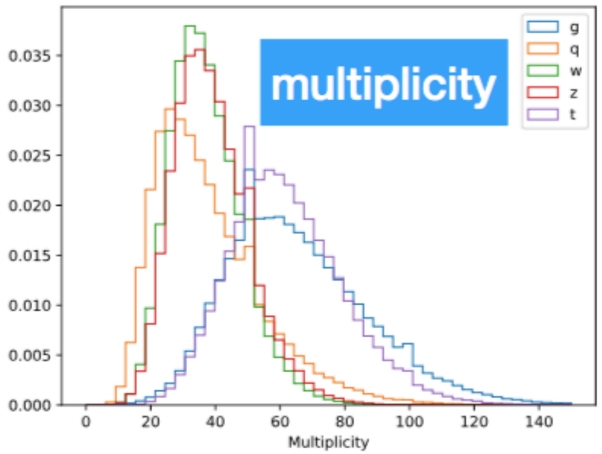
How does it work?



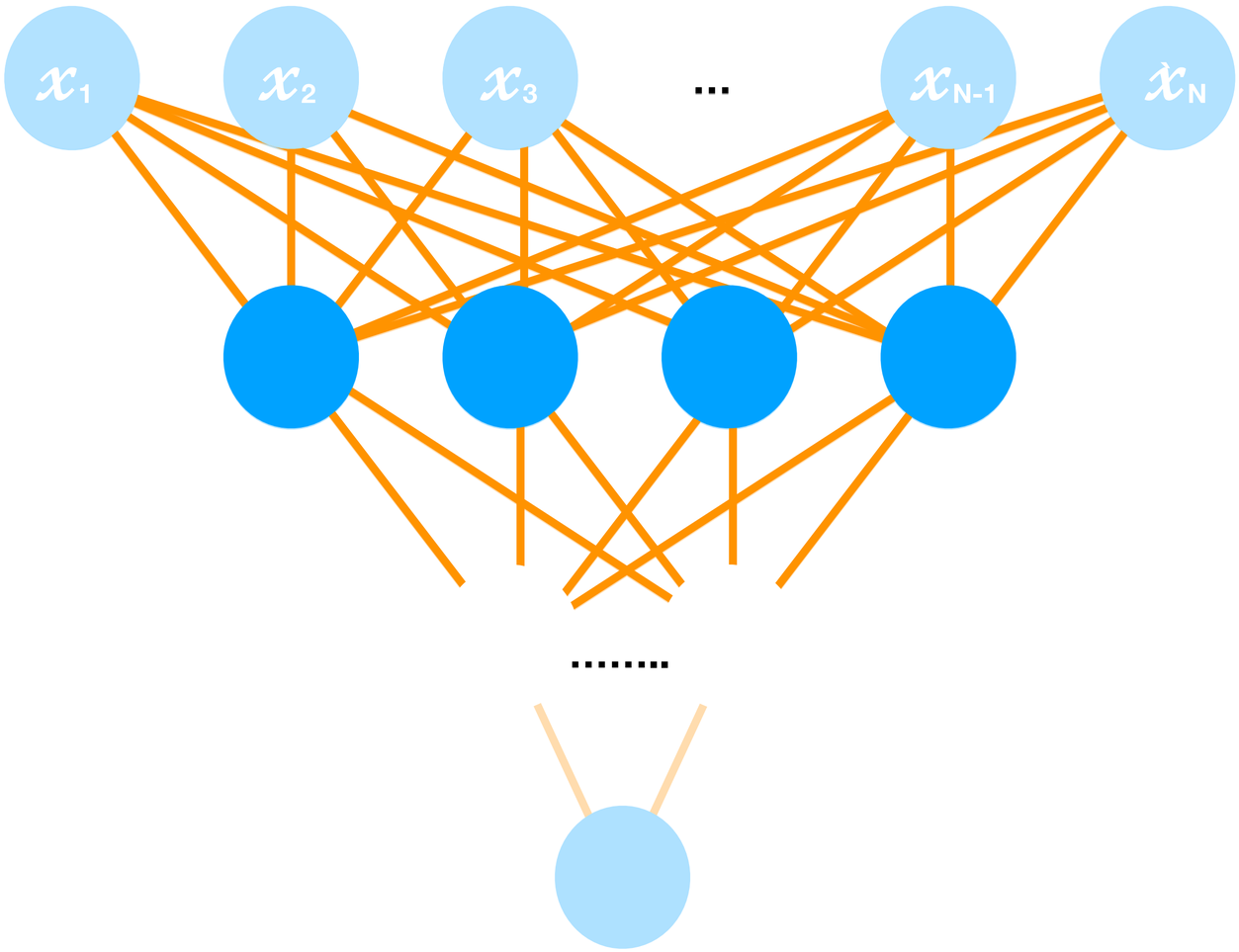
Each input layer transmits a feature to the following hidden layer







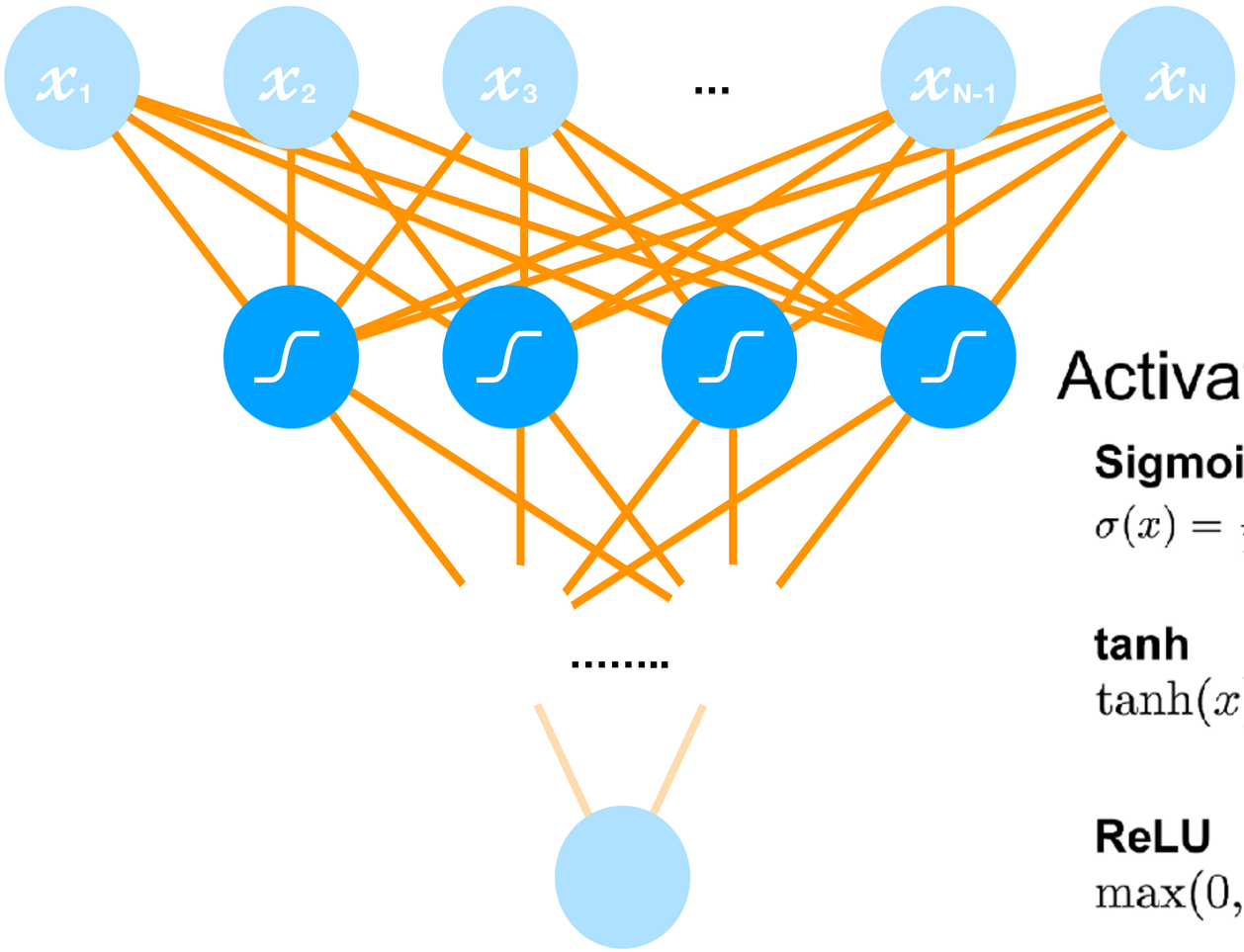
How does it work?



Each hidden neuron computes the weighted sum, shifted by a bias b

$$\sum_j w_{ij} x_j + b_i$$

How does it work?

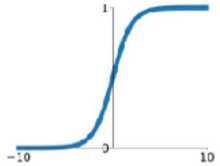


The weighted sum is given as input to an activation function

Activation Functions

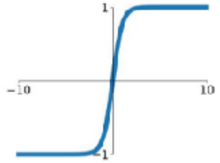
Sigmoid

$\sigma(x) = \frac{1}{1+e^{-x}}$



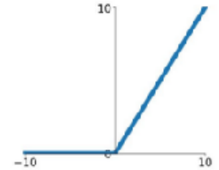
tanh

$\tanh(x)$



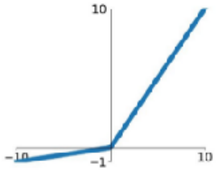
ReLU

$\max(0, x)$



Leaky ReLU

$\max(0.1x, x)$

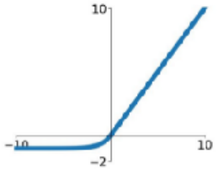


Maxout

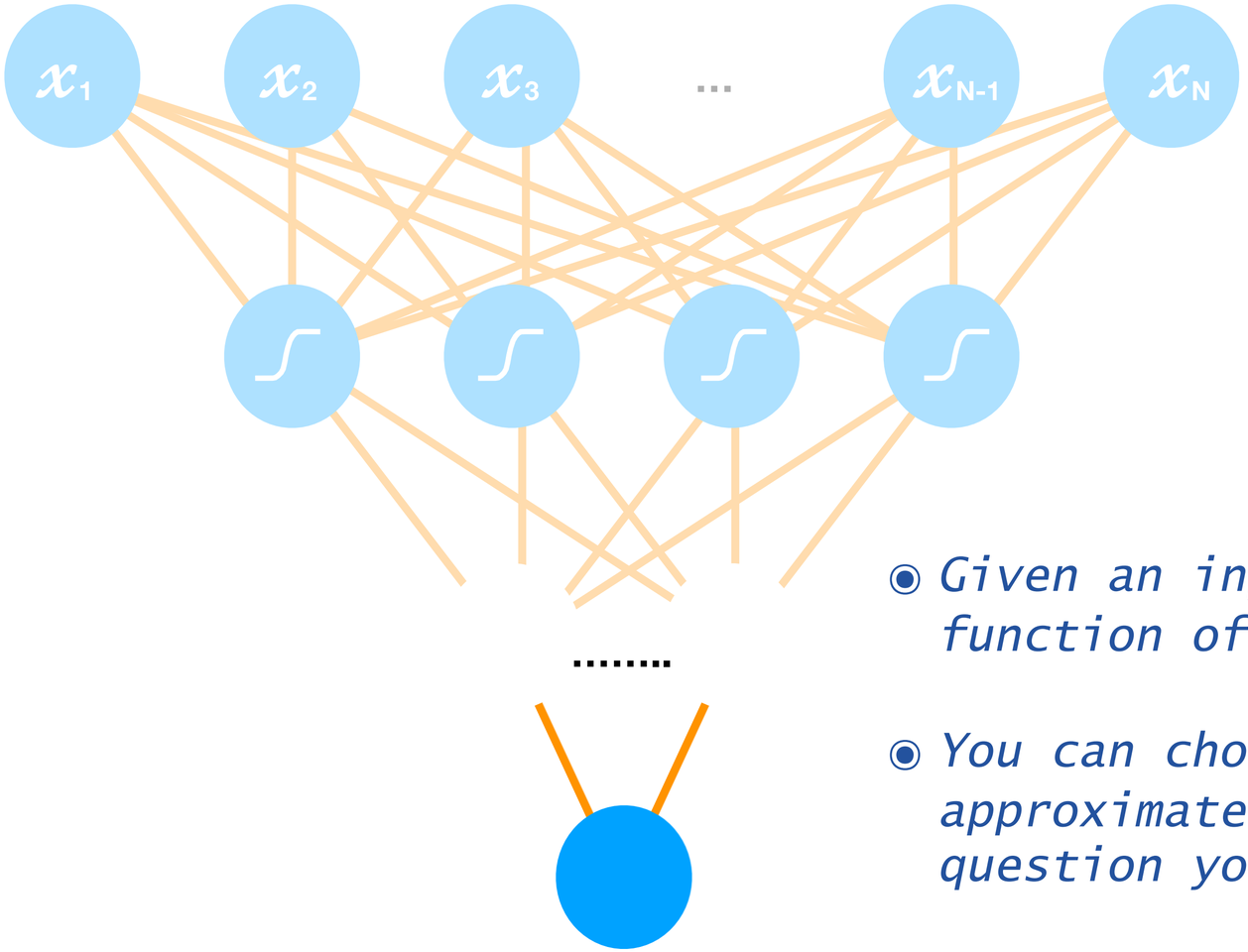
$\max(w_1^T x + b_1, w_2^T x + b_2)$

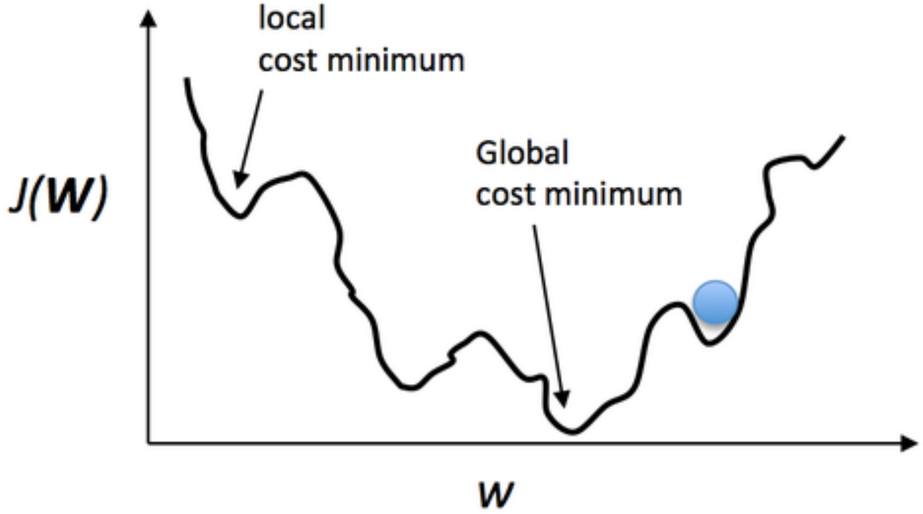
ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



How does it work?





- *Given an input example x, the output ŷ is a function of w and b*
- *You can choose w and b parameters such that ŷ approximates some true value y, the answer to the question you are asking (e.g., which jet is this?)*
- *This is done writing a loss function, a "distance" metric between ŷ and y*

Deep Learning nella vita quotidiana

Everything is a Recommendation

Over 75% of what people watch comes from our recommendations

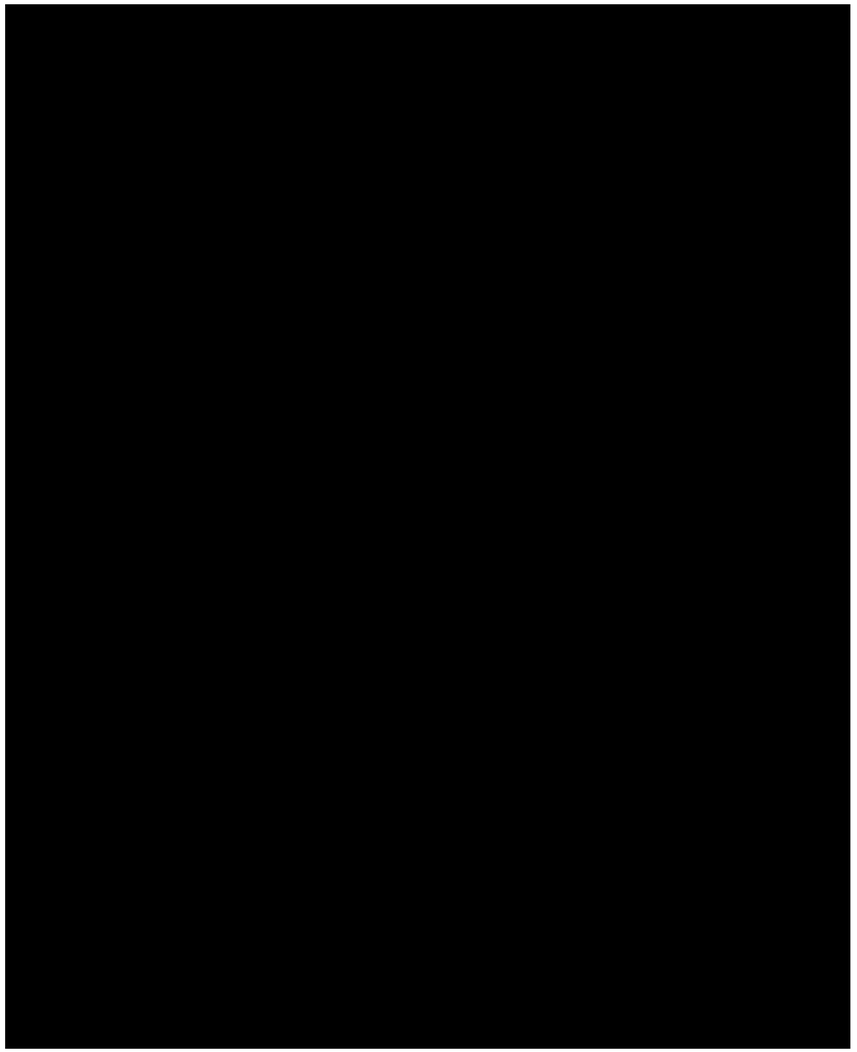
Recommendations are driven by **Machine Learning**

NETFLIX

European Research Council

Does it work? Why?

- *The non-linearity of the activation function allows to approximate non-linear functions y = f(x)*
- *The network learns by example to reproduce a result*
- *Can be used as a shortcut of complex algorithms when you have short time to get your answer*
- *The network is NOT intelligent in a human sense: don't ask the network to explain you how it gets the right answer*
- *Still, networks tend to get the right answers (i.e., they are intelligent in the most practical way)*

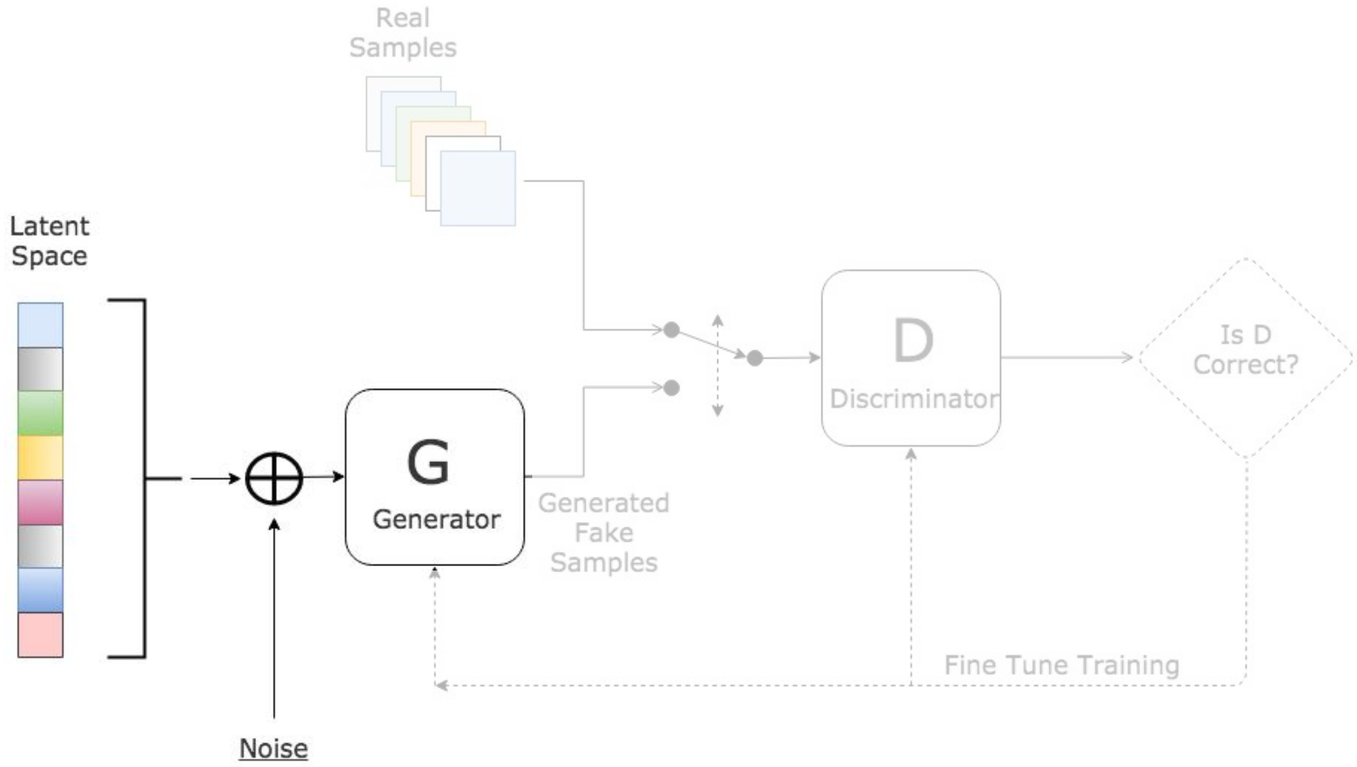


Generative Adversarial Networks

European Research Council

Generative Adversarial Training

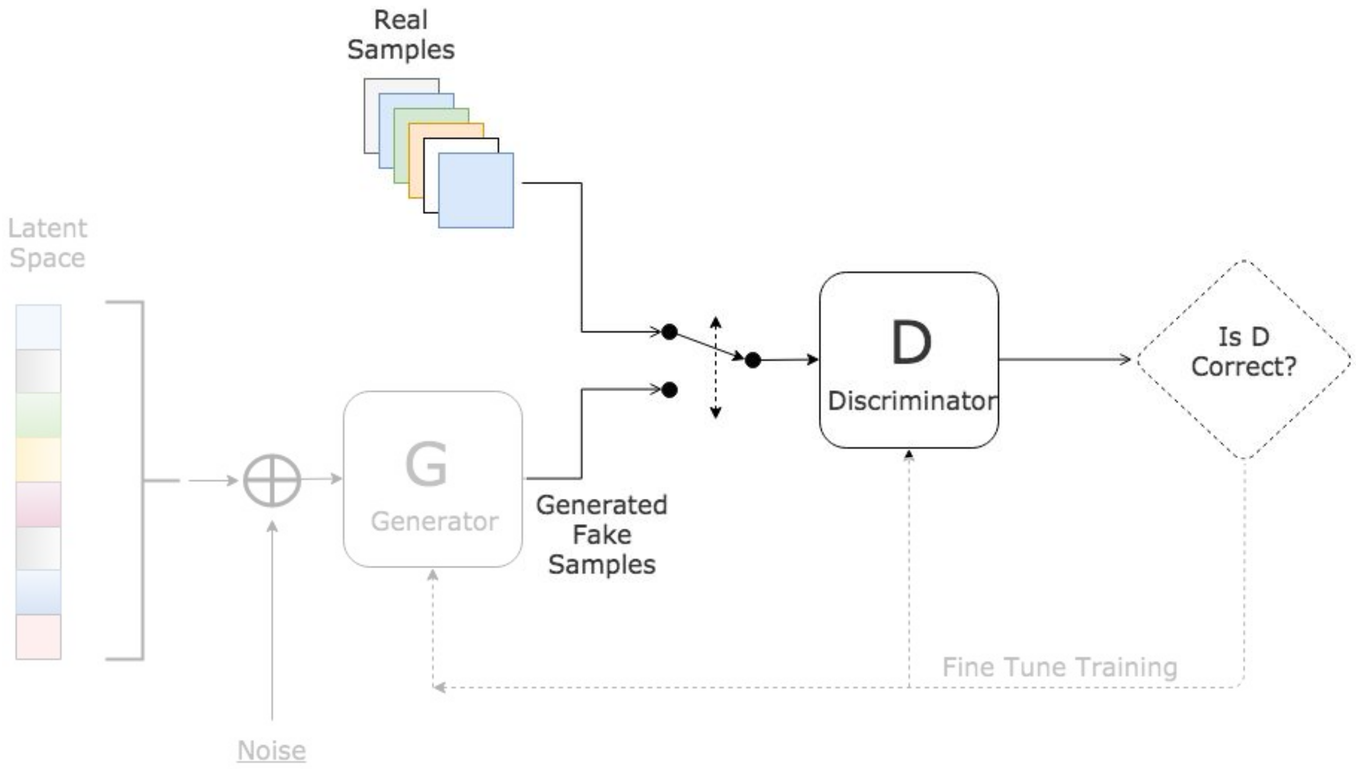
- *Due reti allenate in competizione*
  - ***Generatore: crea immagini (in modo piu' o meno casuale)***
  - *Discriminatore: cerca di capire se una data immagine e' vera o se viene dal generatore*



- *La loss function da minimizzare e' scritta come Loss(Gen)-Loss(Disc)*
  - *Se il discriminatore migliora , la loss aumenta*
  - *Se il generatore migliora, la loss diminuisce*
- *Nel cercare di confondere il discriminatore, il generatore impara a creare immagini realistiche*

Generative Adversarial Training

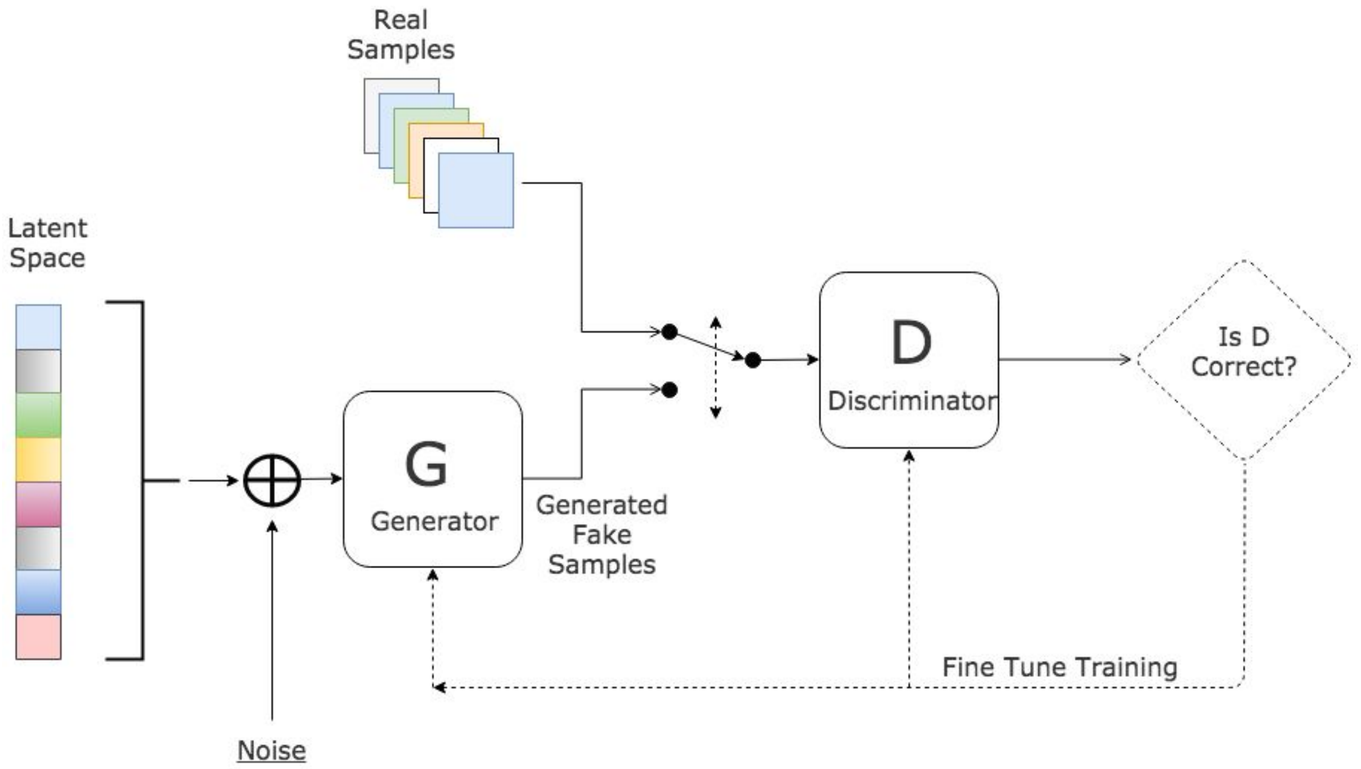
- *Due reti allenate in competizione*
  - *Generatore: crea immagini (in modo piu' o meno casuale)*
  - ***Discriminatore: cerca di capire se una data immagine e' vera o se viene dal generatore***



- *La loss function da minimizzare e' scritta come Loss(Gen)-Loss(Disc)*
  - *Se il discriminatore migliora , la loss aumenta*
  - *Se il generatore migliora, la loss diminuisce*
  - *Nel cercare di confondere il discriminatore, il generatore impara a creare immagini realistiche*

Generative Adversarial Training

- *Due reti allenate in competizione*
  - *Generatore: crea immagini (in modo piu' o meno casuale)*
  - *Discriminatore: cerca di capire se una data immagine e' vera o se viene dal generatore*



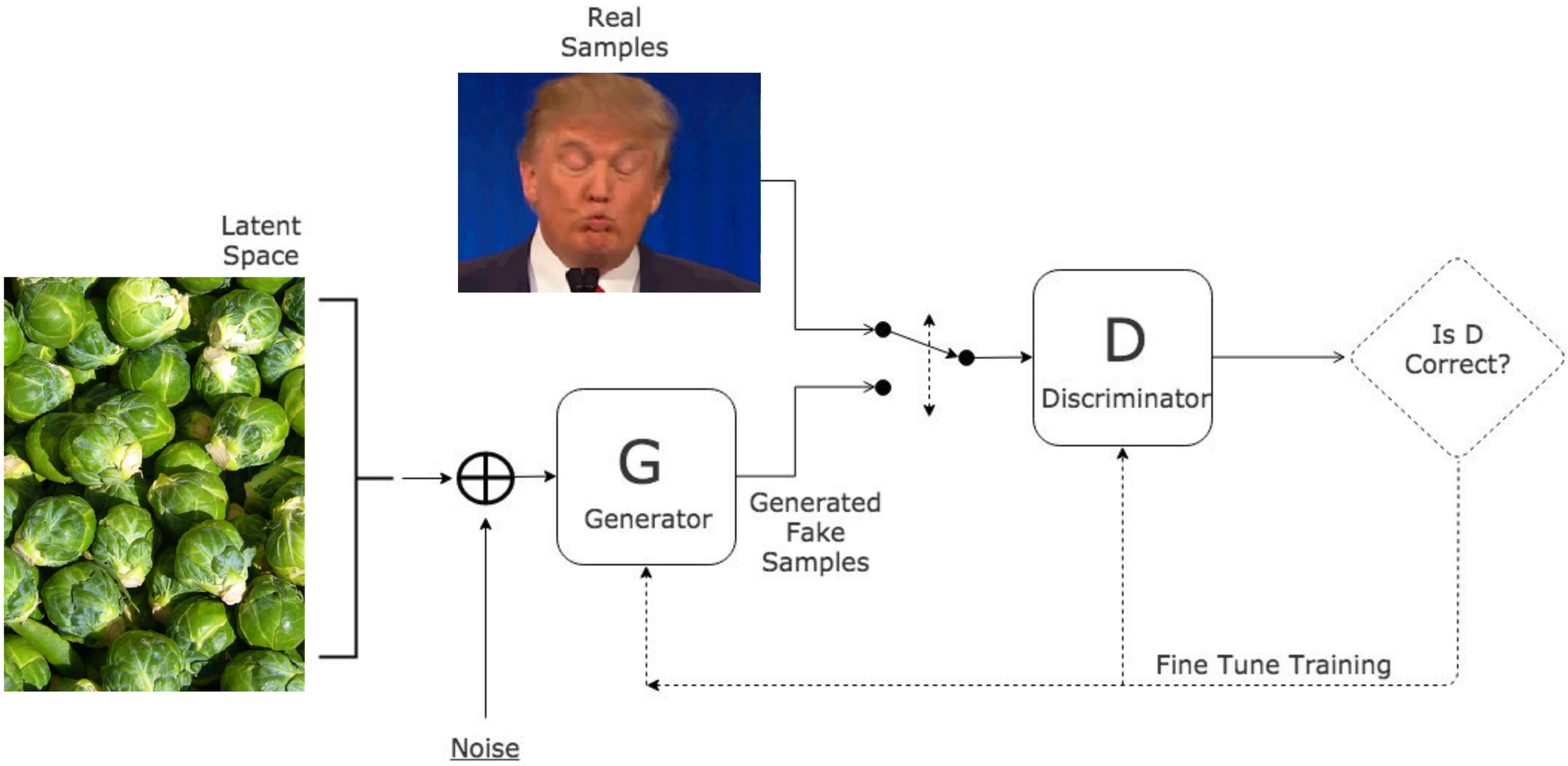
- ***La loss function da minimizzare e' scritta come Loss(Gen)-Loss(Disc)***
  - ***Se il discriminatore migliora , la loss aumenta***
  - ***Se il generatore migliora, la loss diminuisce***
  - ***Nel cercare di confondere il discriminatore, il generatore impara a creare immagini realistiche***

Adversarial training in azione

Adversarial training in azione

- *Il generatore e' una funzione matematica*
- *Cambiando in modo continuo gli input, si puo' passare in maniera continua da un volto all'altro*

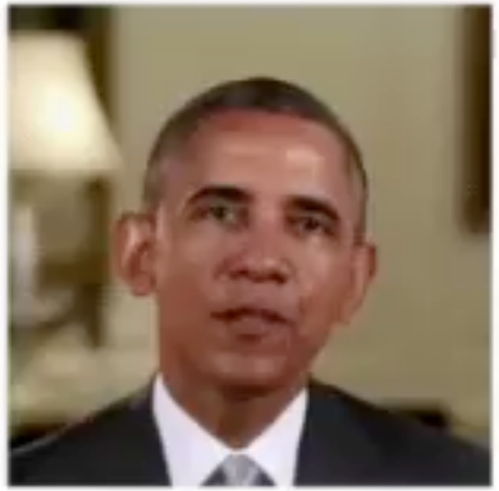
Adversarial training in azione



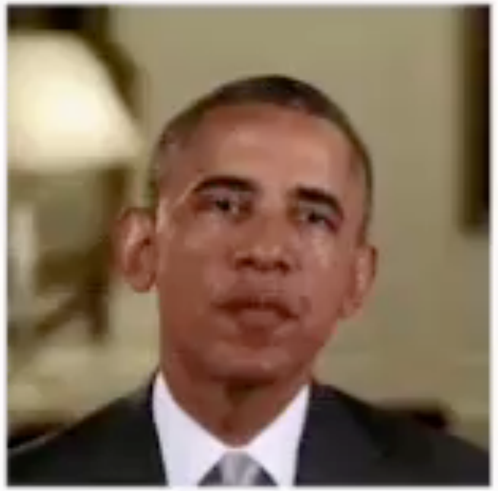
Adversarial training in azione

Adversarial training in azione

Source Sequence



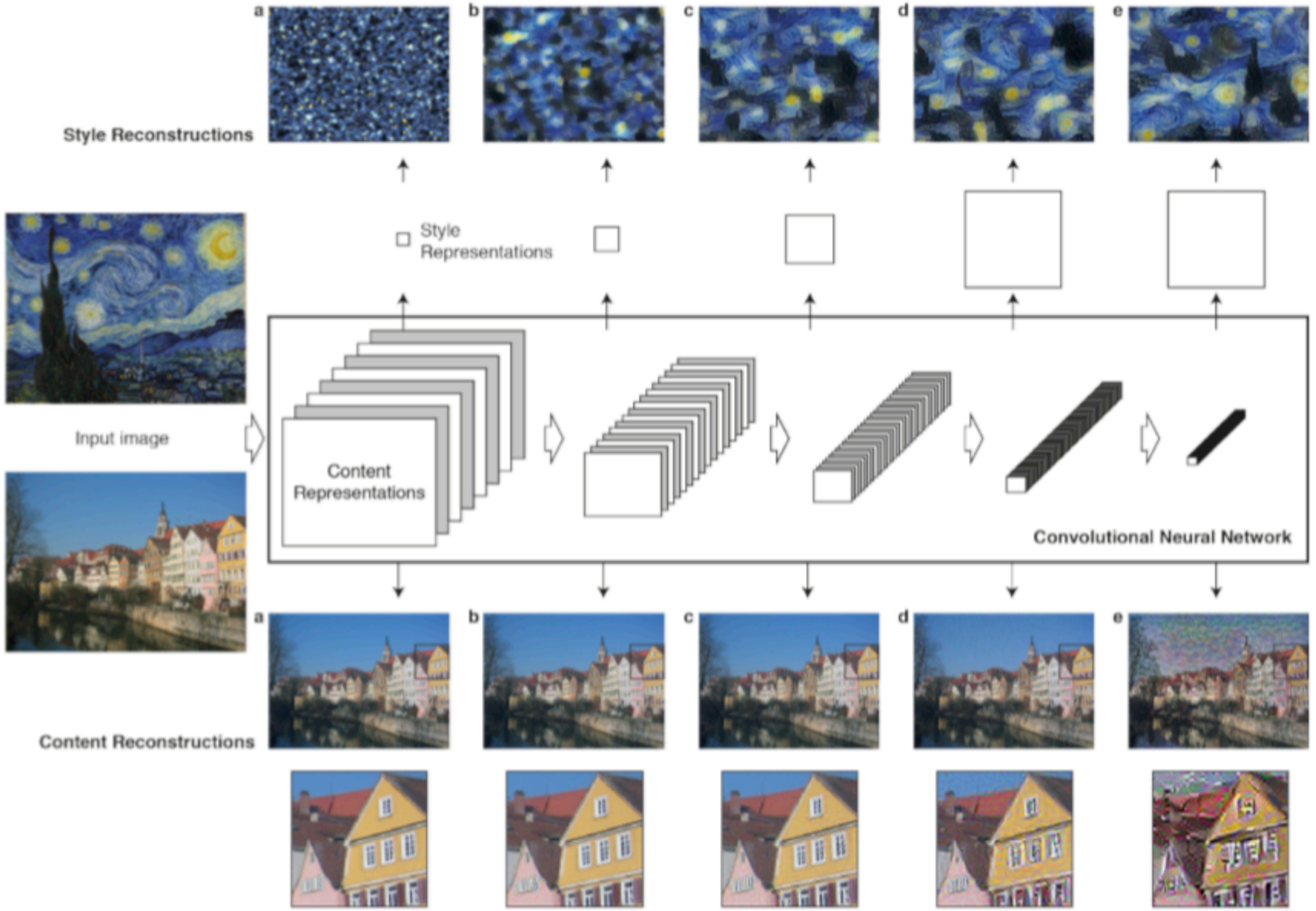
Our Reenactment
(Full Head)



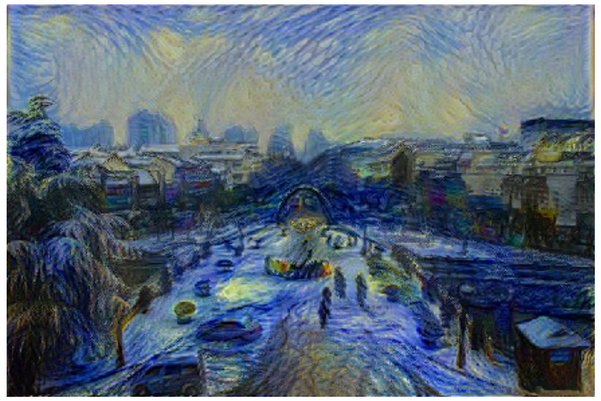
Averbuch-Elor et al. 2017

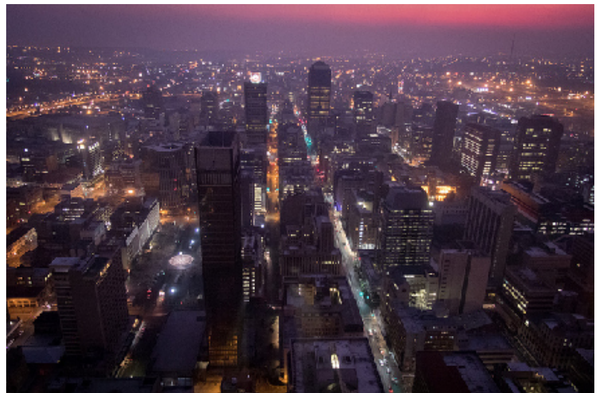
European Research Council

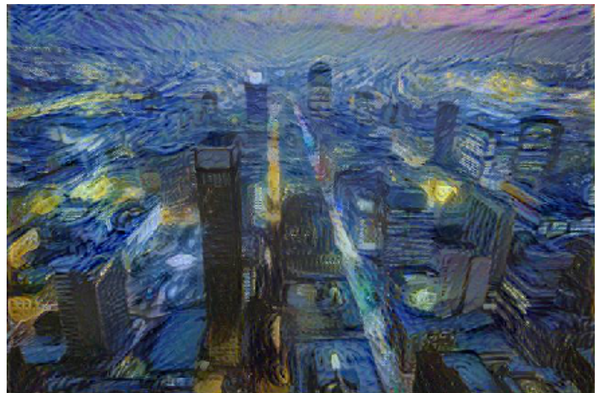
Style Transfer



Style Transfer



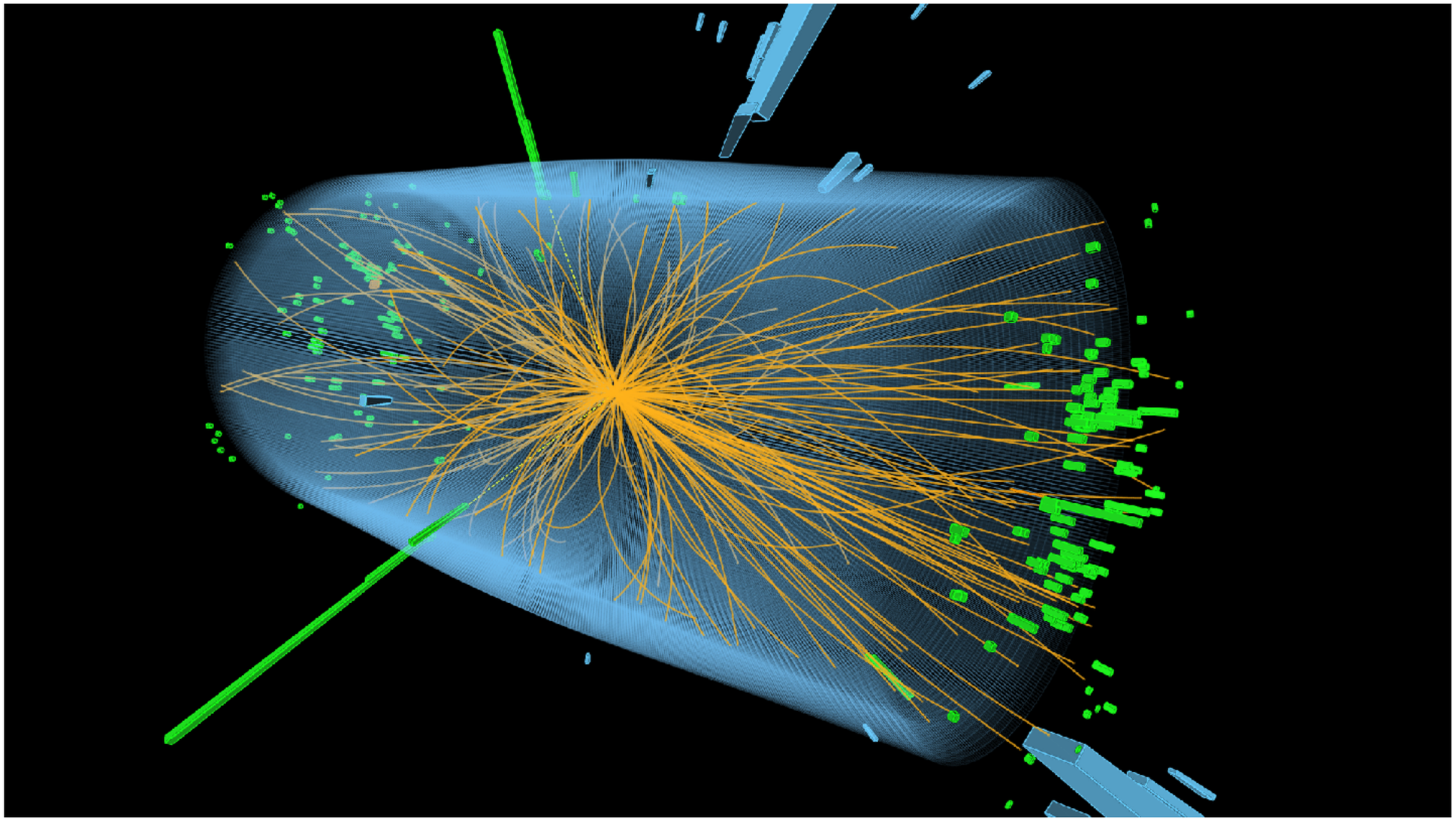




European Research Council

Style Transfer

What can we use this for?

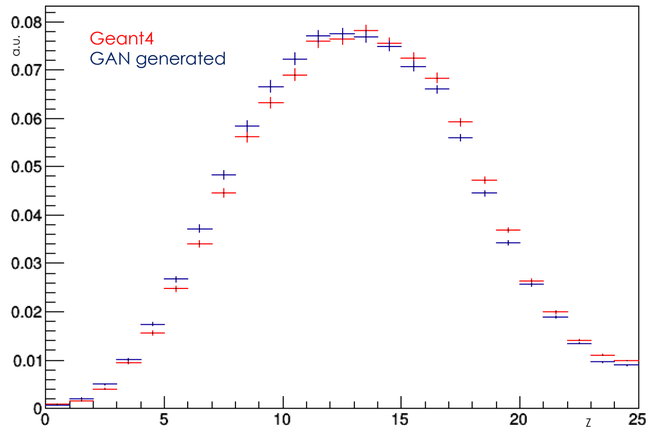


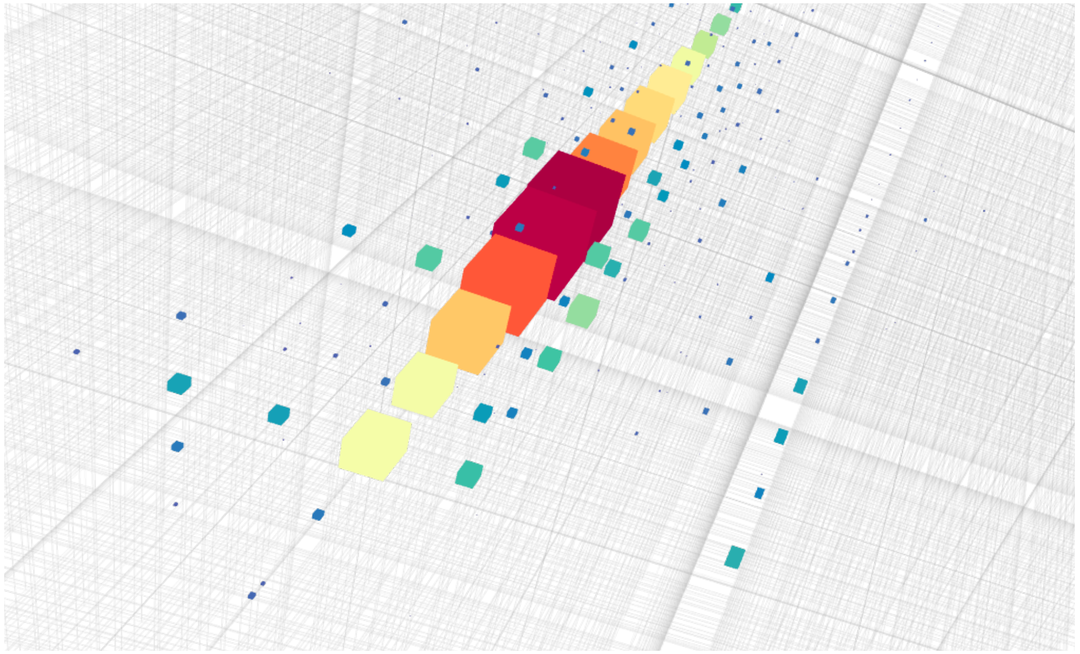
European Research Council

Image generation

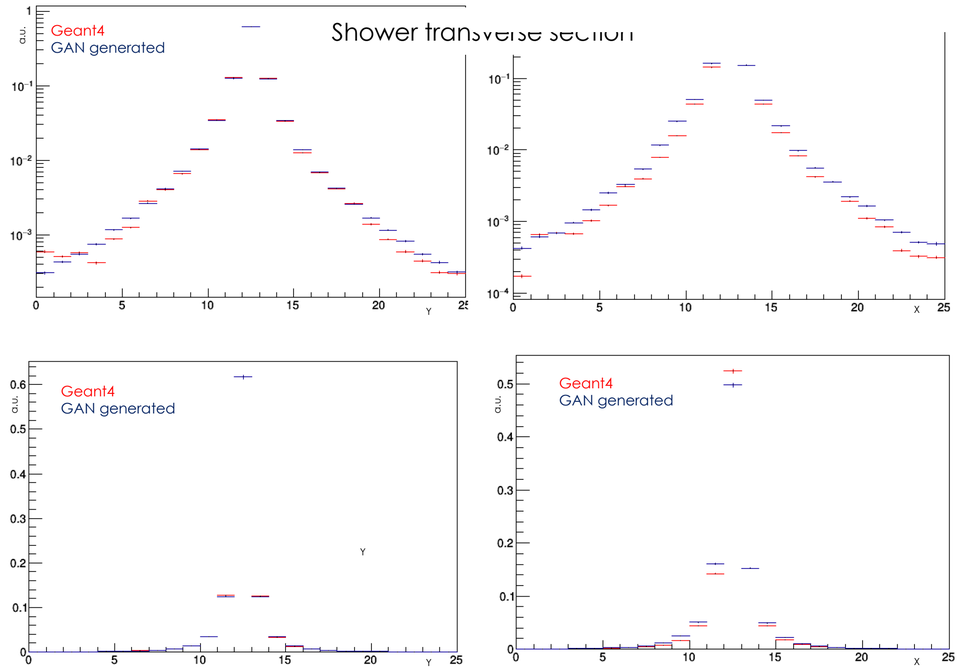
- Start from random noise
- Works very well with images
  - Applied to electron showers in digital calorimeters as a replacement of GEANT

Shower longitudinal section





Shower transverse section

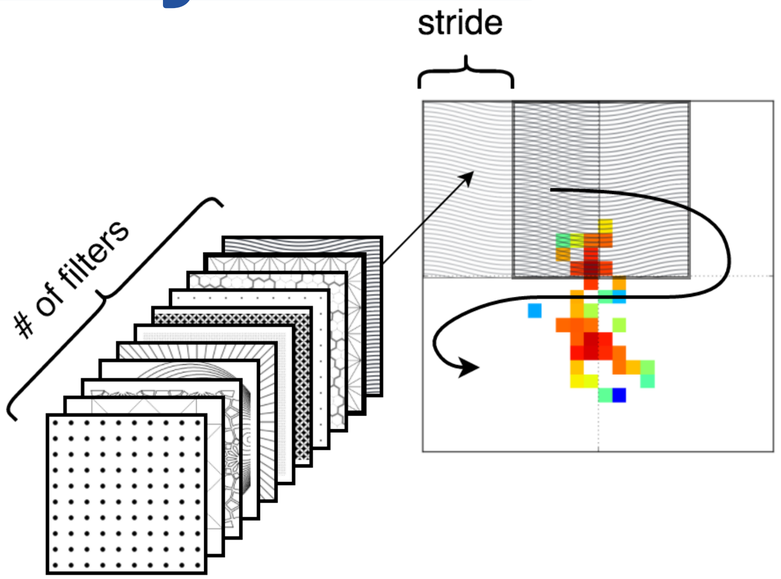


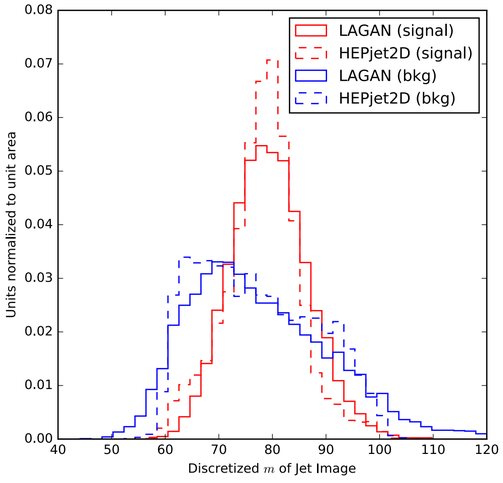
See contribution to NIPS workshop

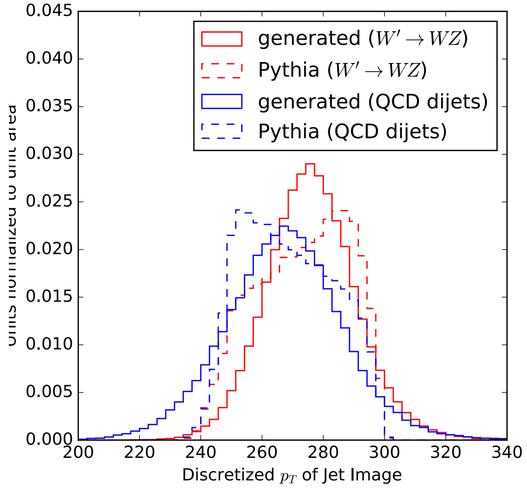
European Research Council

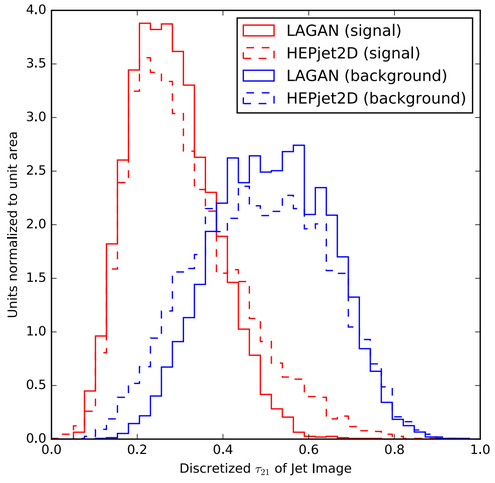
Generating full jets

- Start from random noise
- Works very well with images
  - Applied to electron showers in digital calorimeters as a replacement of GEANT





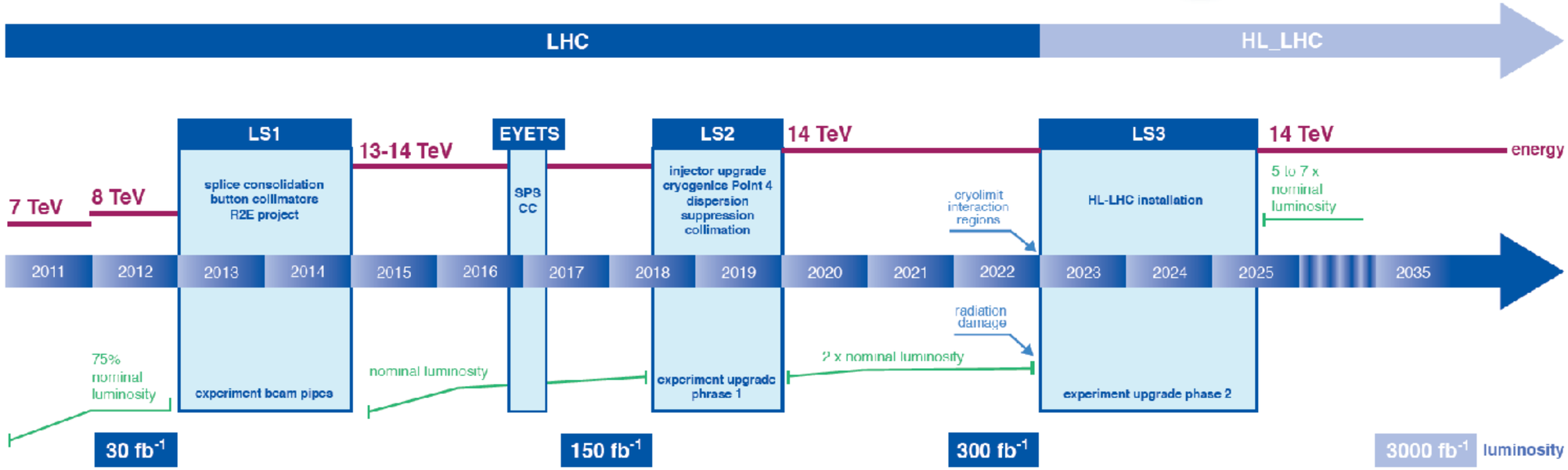




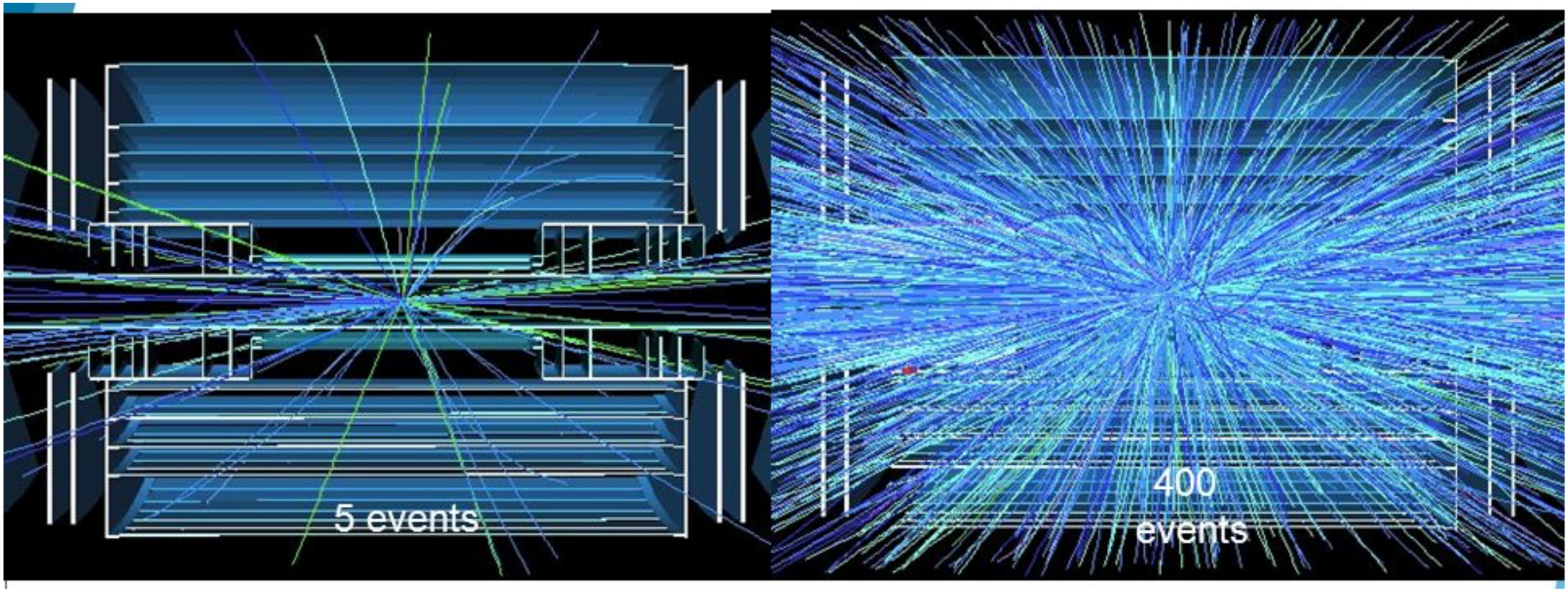
de Olivera, Paganini, and Nachman
https://arxiv.org/pdf/1701.05927.pdf

Figure 6: The distributions of image mass $m(I)$, transverse momentum $p_{\mathrm{T}}(I)$, and n-subjettiness $\tau_{21}(I)$. See the text for definitions.

But Why?



- ***We need to be ready by 2025 (High-Luminosity LHC)***
- *LHC Run 3 (2020-2022) is the ultimate demonstration opportunity*
  - *for model building, deployment and commissioning*
- ***Strong synergy with other research lines in HEP**, when Deep Learning is even easier to apply*
  - *Dark Matter underground experiments*
  - *Neutrino experiments*

European Research Council

But Why?