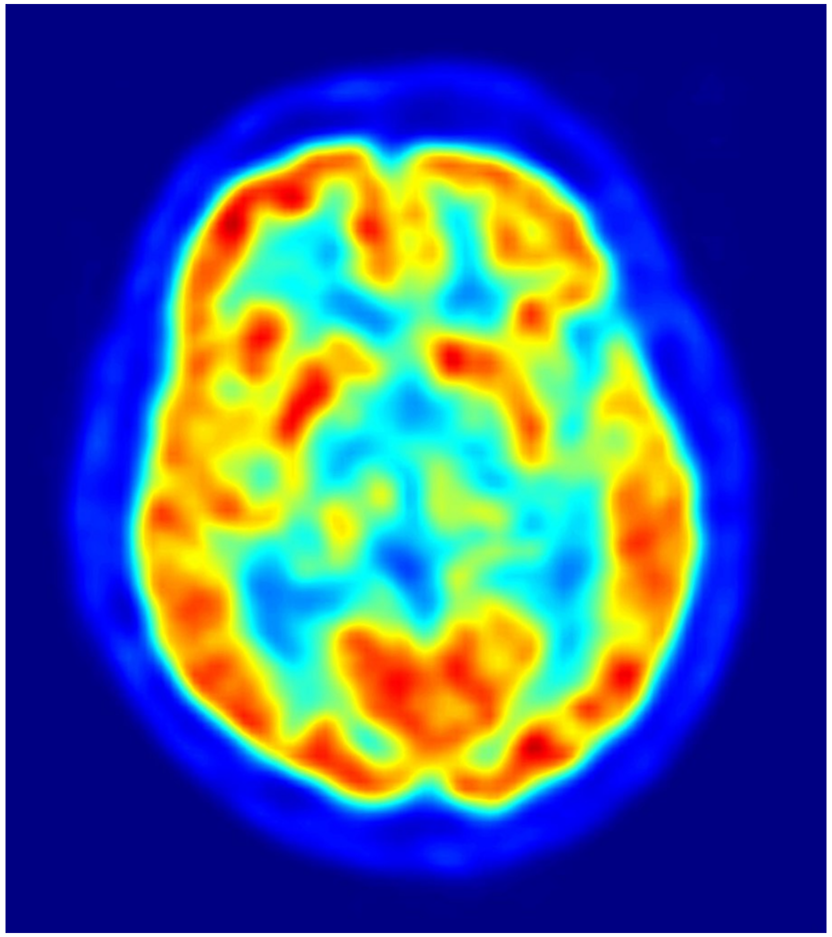
Reinforcement Learning in HEP

Andrea Mauri

UZH seminar

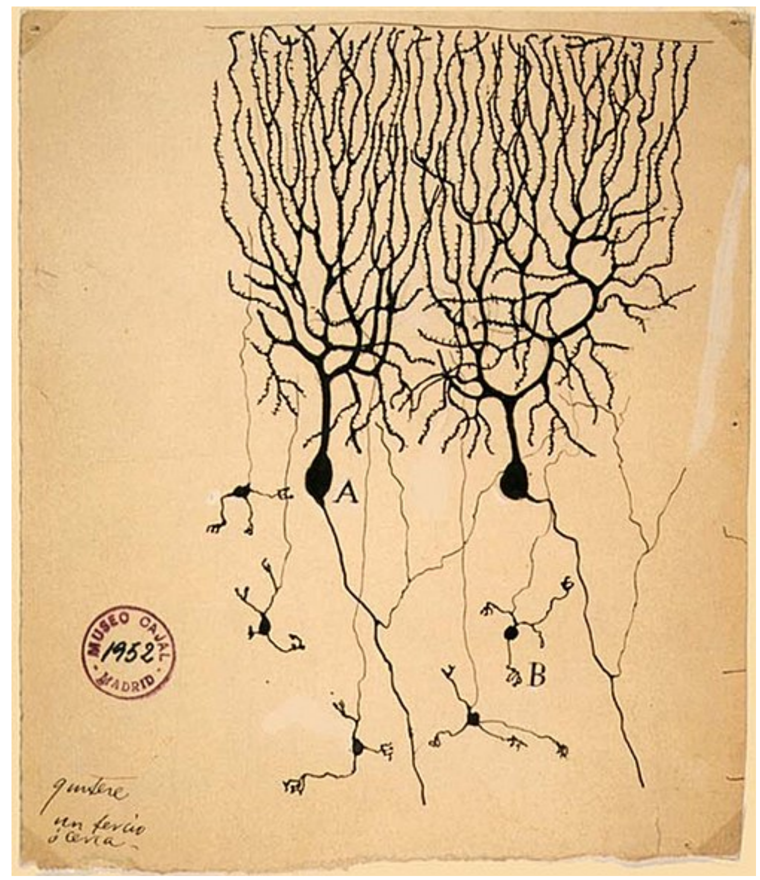
May 10th, 2021

output



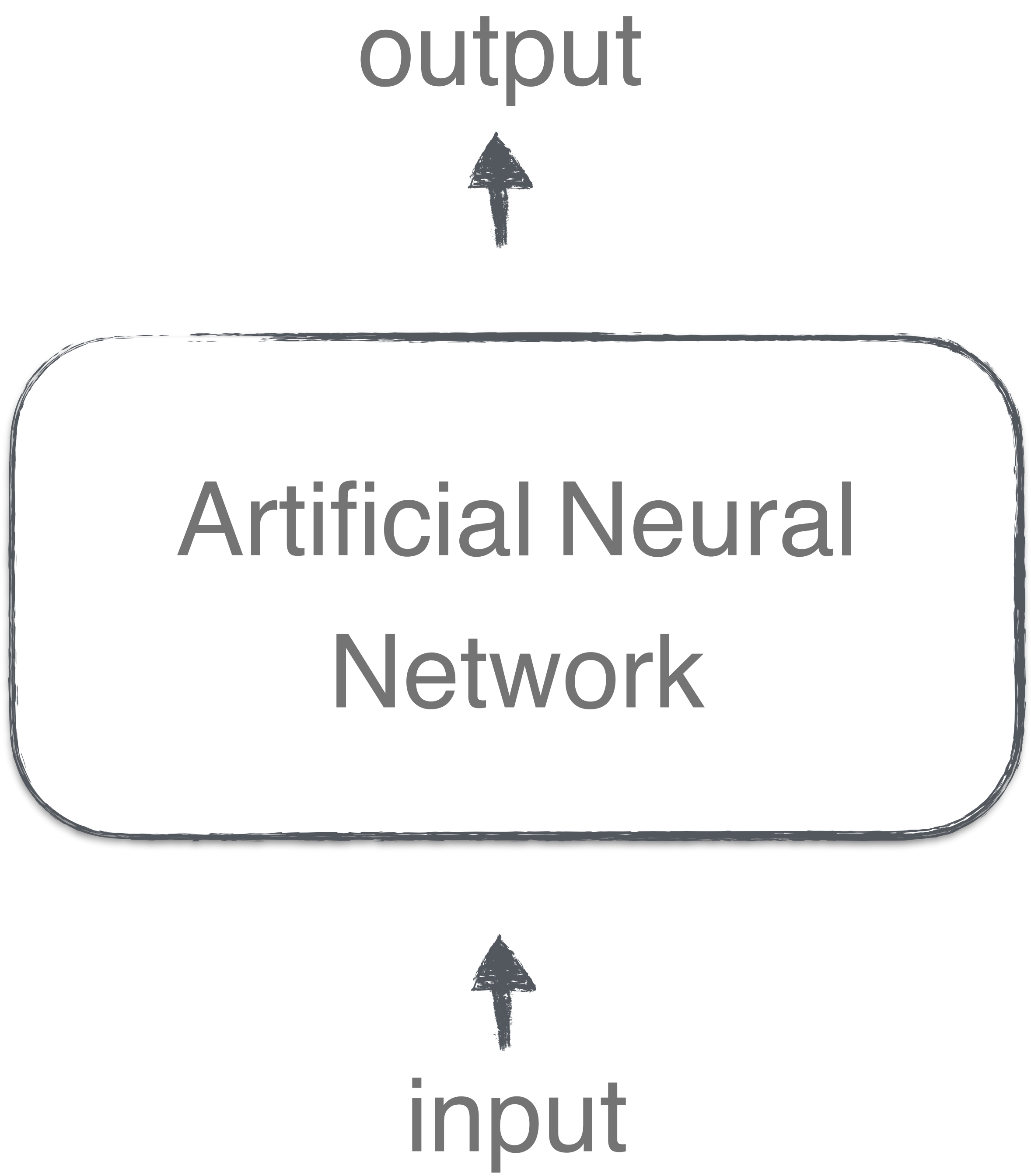
input

output

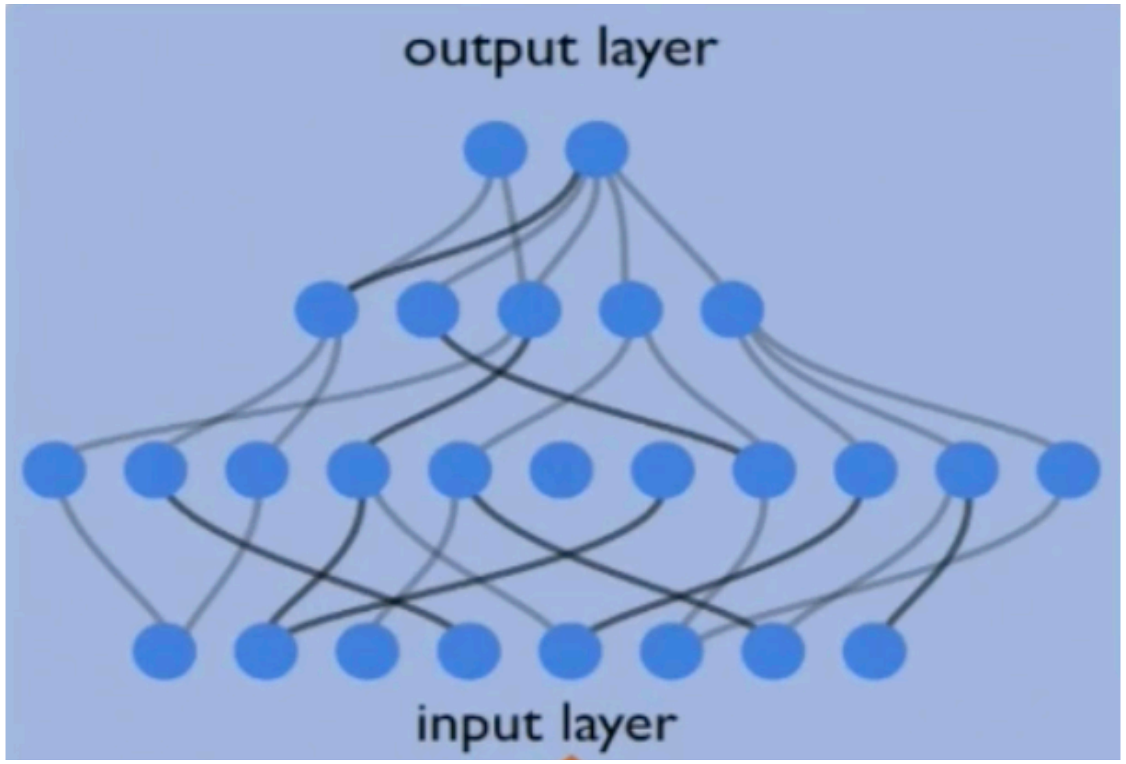


Ramón y Cajal, 1899

input

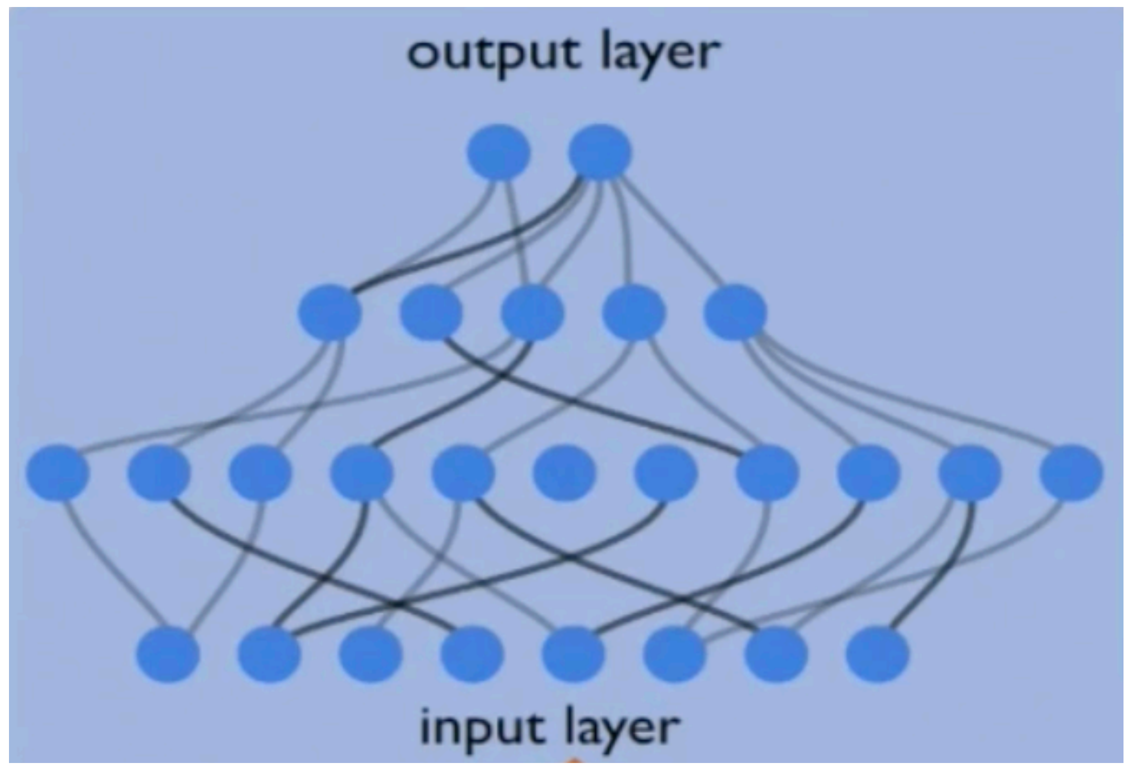

output
Artificial Neural Network
input

output



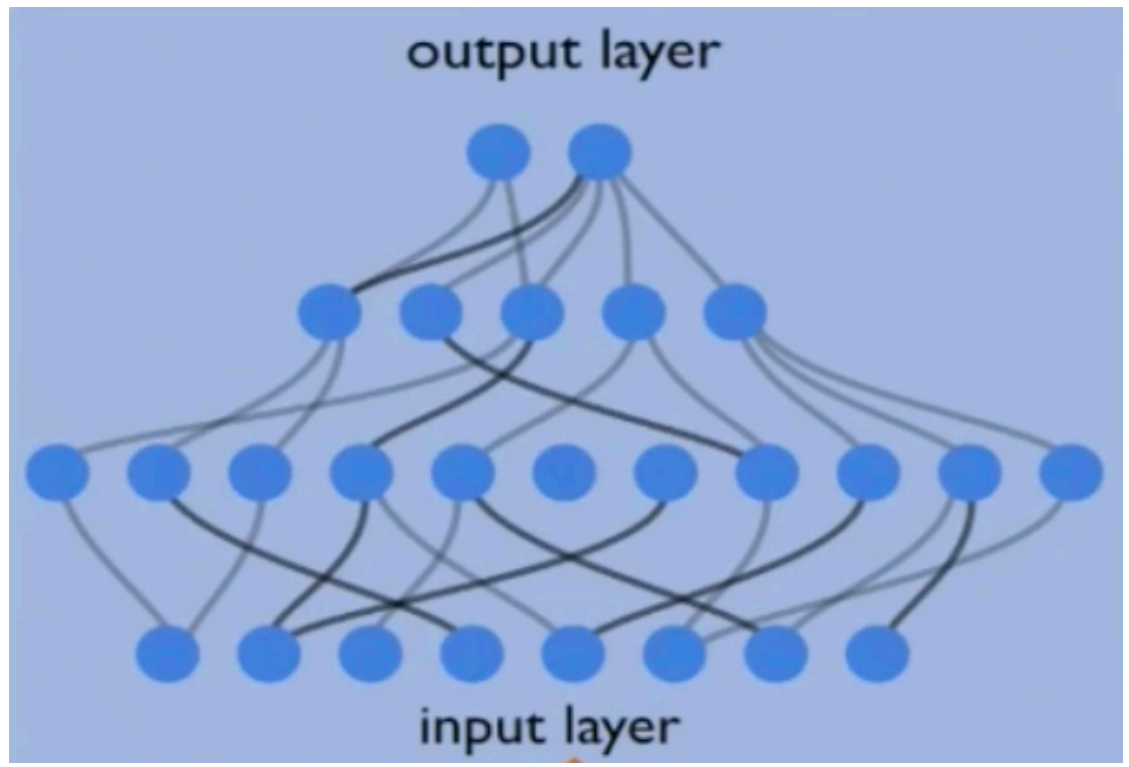
input

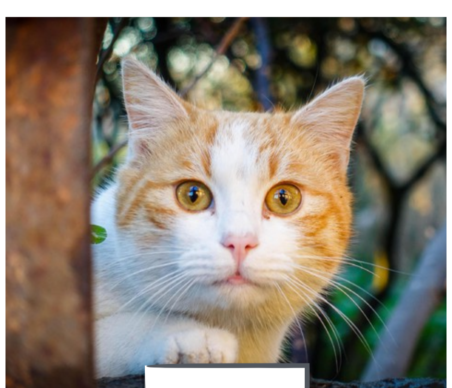
"cat"



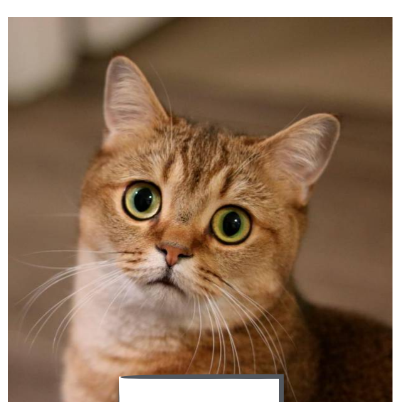
Supervised learning

"cat"





cat



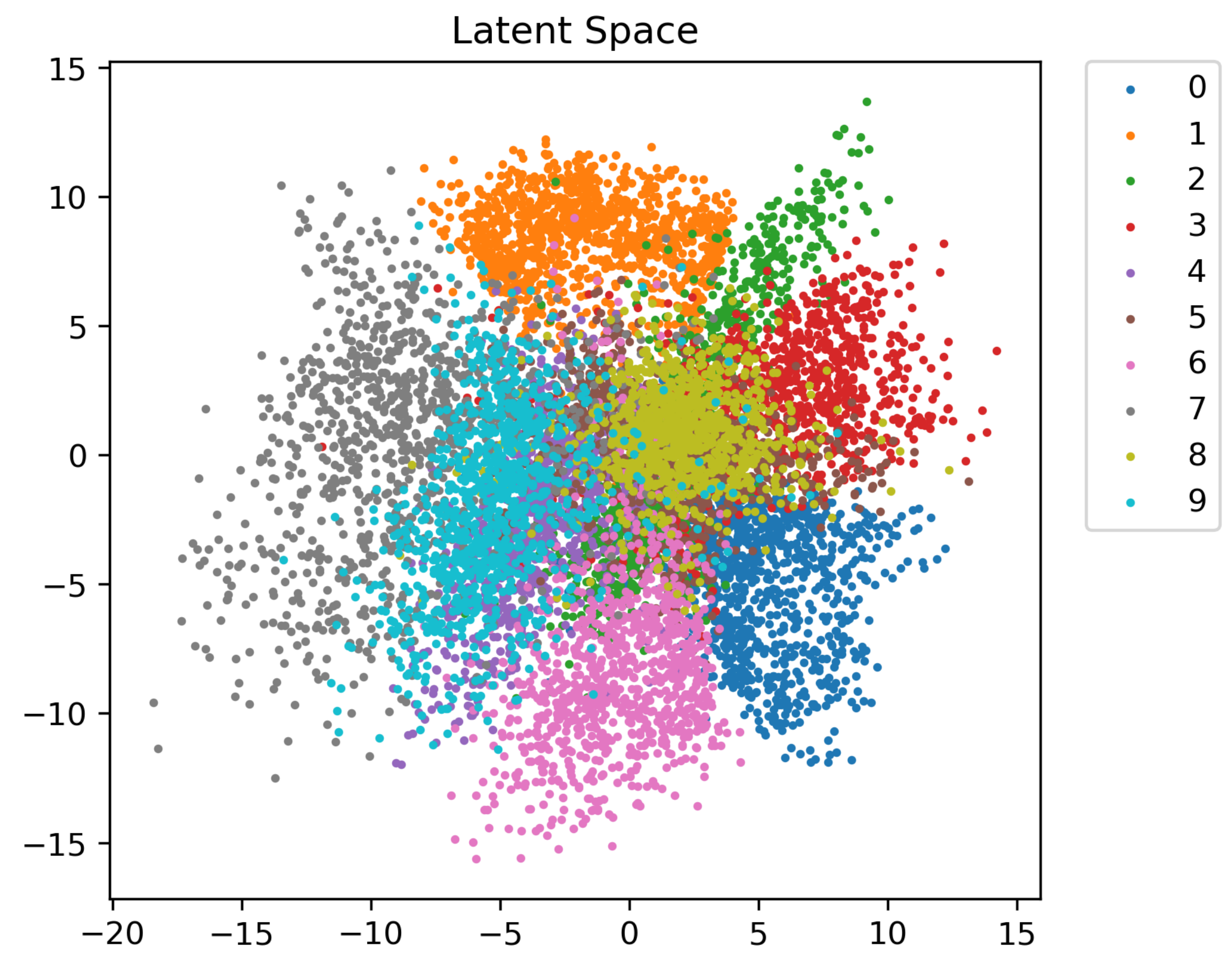
cat

more cats...

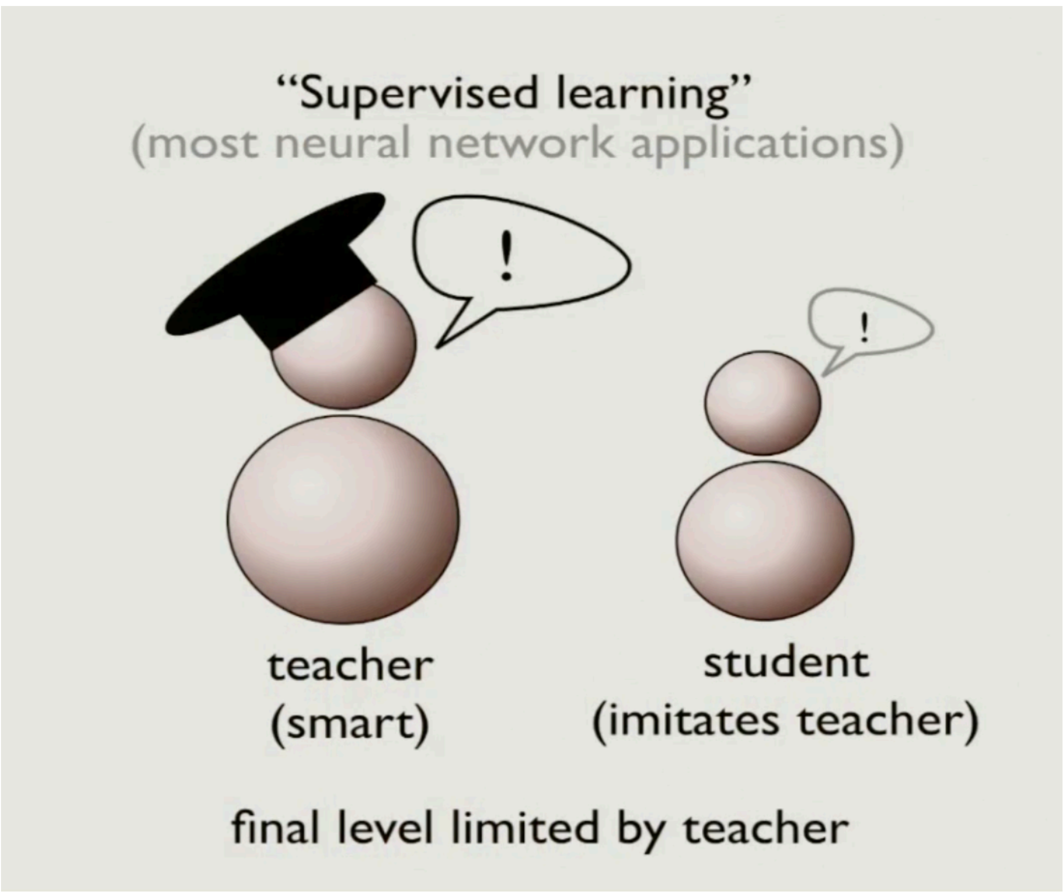
?

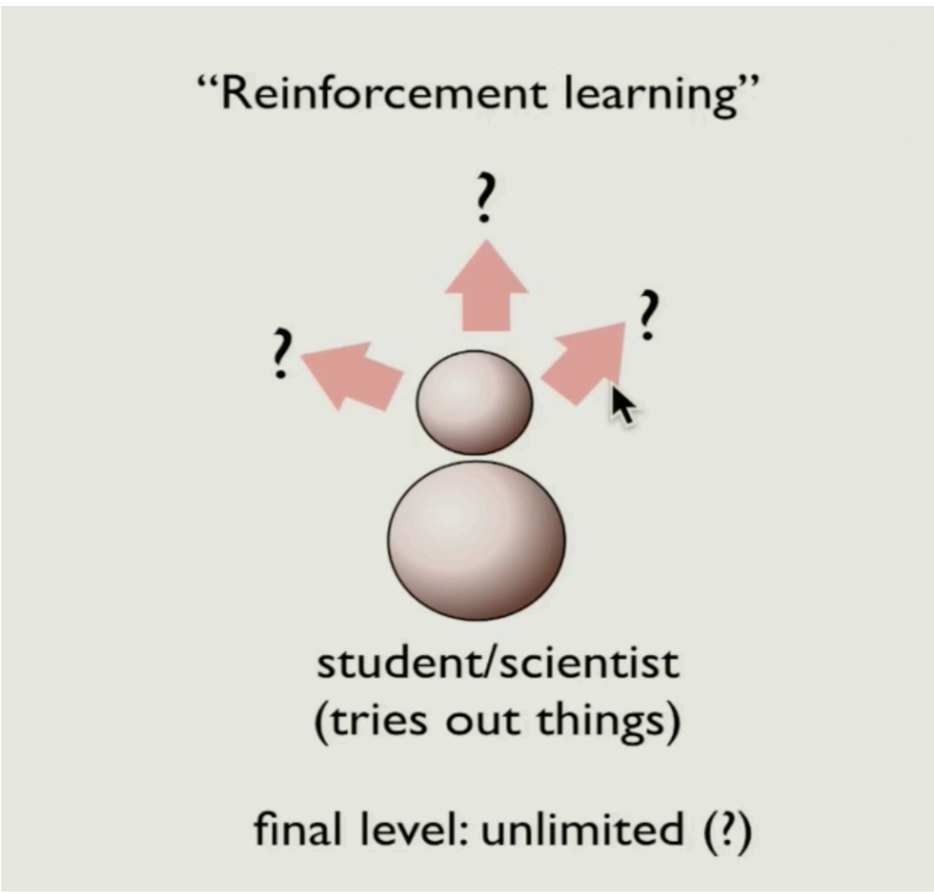
Unsupervised learning

- Extract crucial features without any guidance



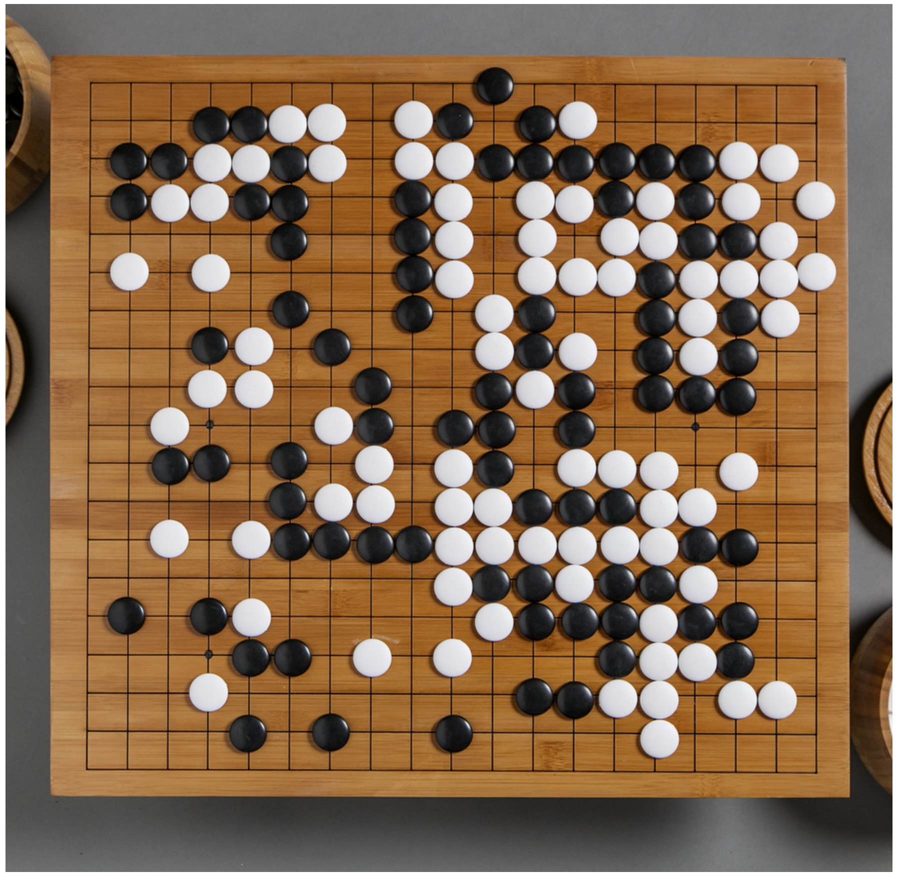
Reinforcement Learning

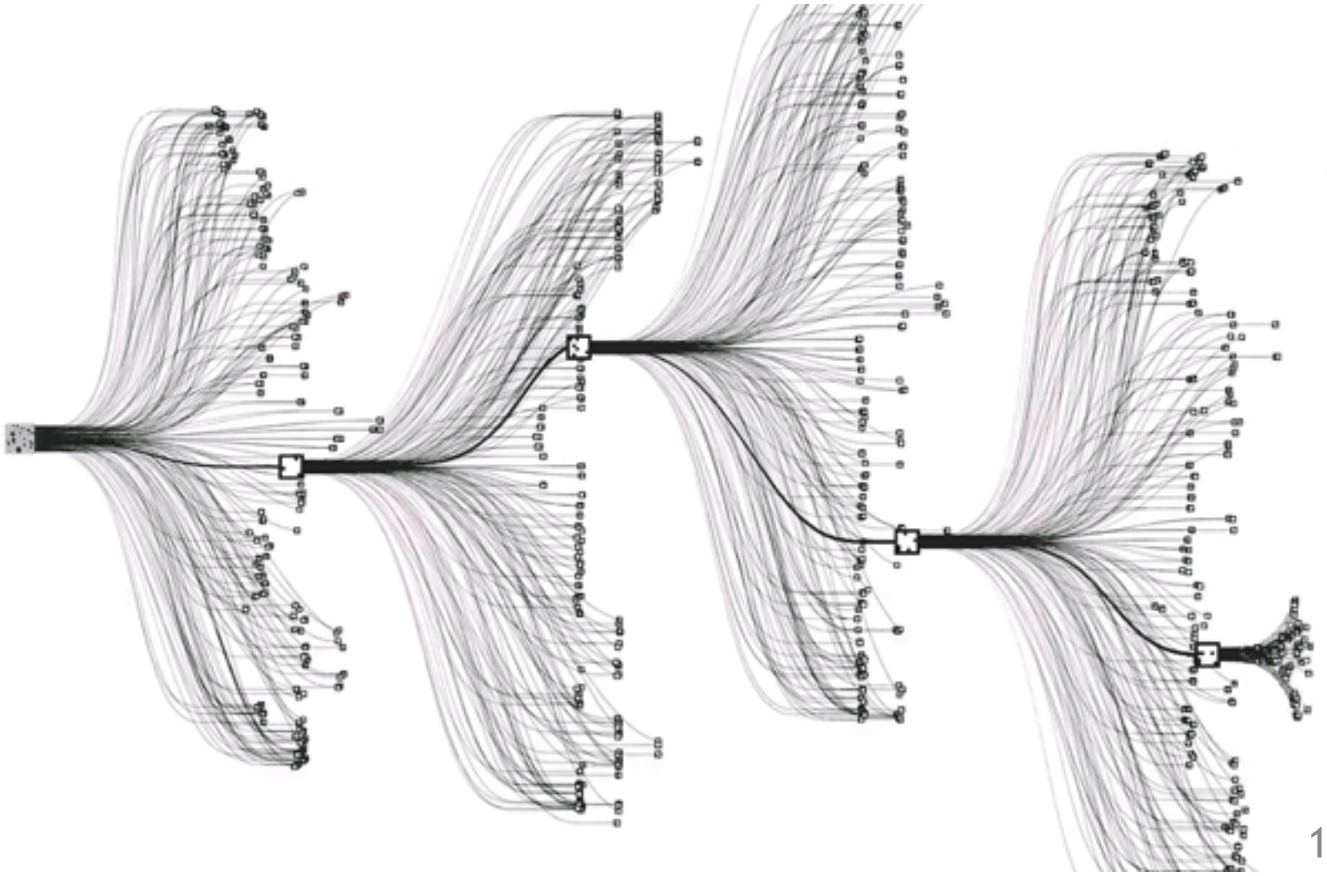

"Supervised learning"
(most neural network applications)
!
!
teacher
(smart)
student
(imitates teacher)
final level limited by teacher


"Reinforcement learning"
?
?
?
student/scientist
(tries out things)
final level: unlimited (?)

AlphaGo

- In 2016 AlphaGo defeated world champion Lee Sedol
- only one kind of move: place a stone
- 19 x 19 board
- win by surrounding more territories than your opponent
- 10^{170} possible board configuration
- experts player often motivate moves by intuition

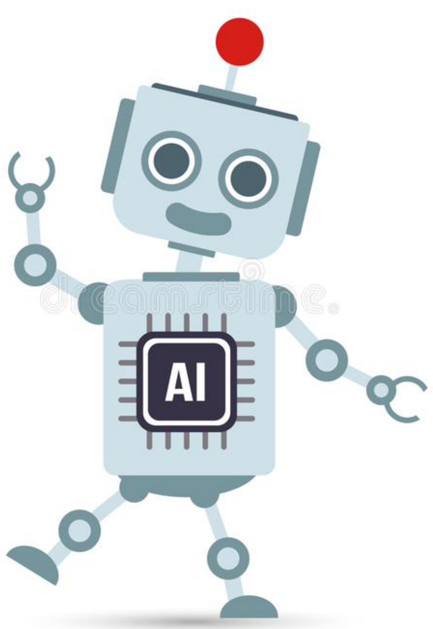


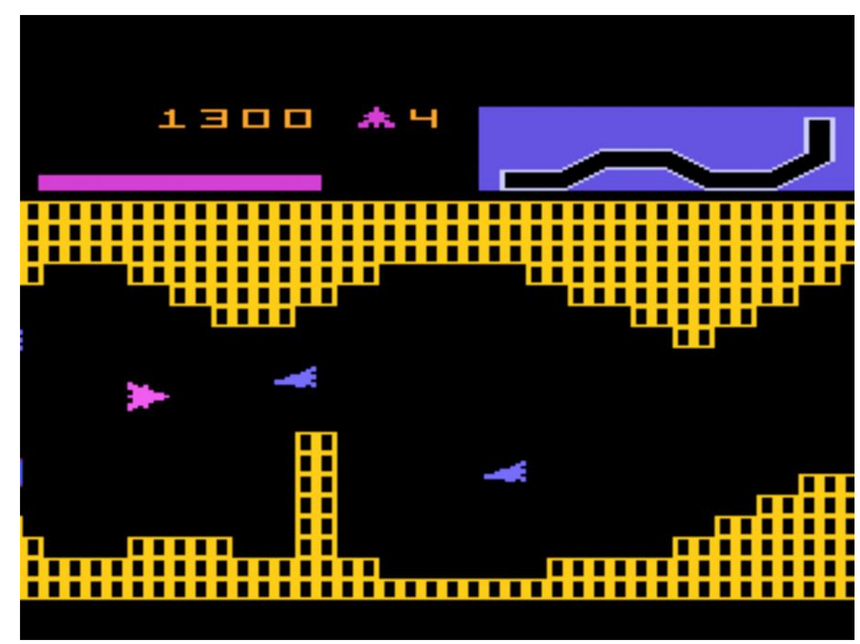


Reinforcement learning

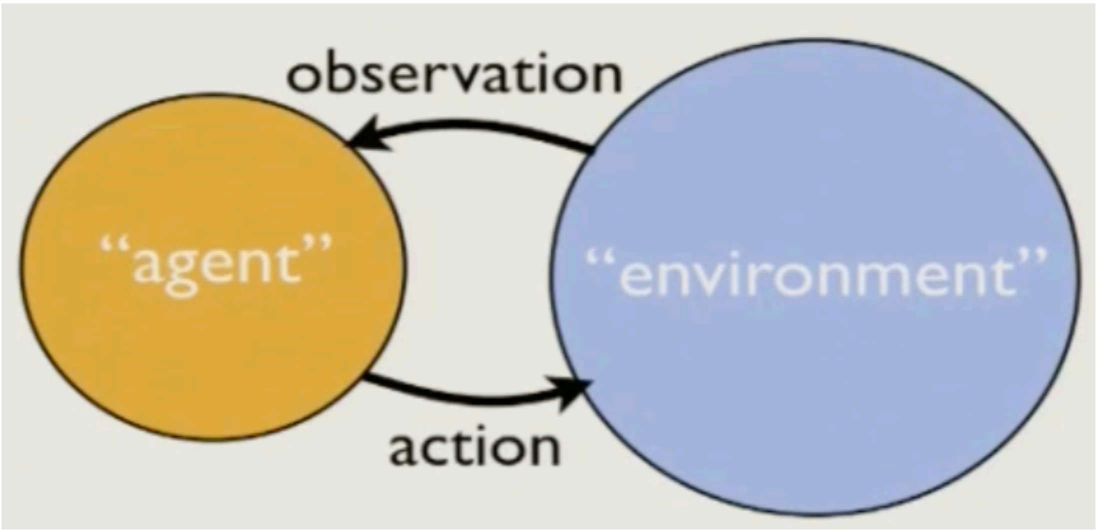
- Define a *goal*
  - We do not tell how to reach the goal, we only say what is *good* and what is *bad*
- We are not the *"teacher"* anymore, more like a *"customer"*...

Closest concept to Artificial Intelligence (AI)





Reinforcement learning

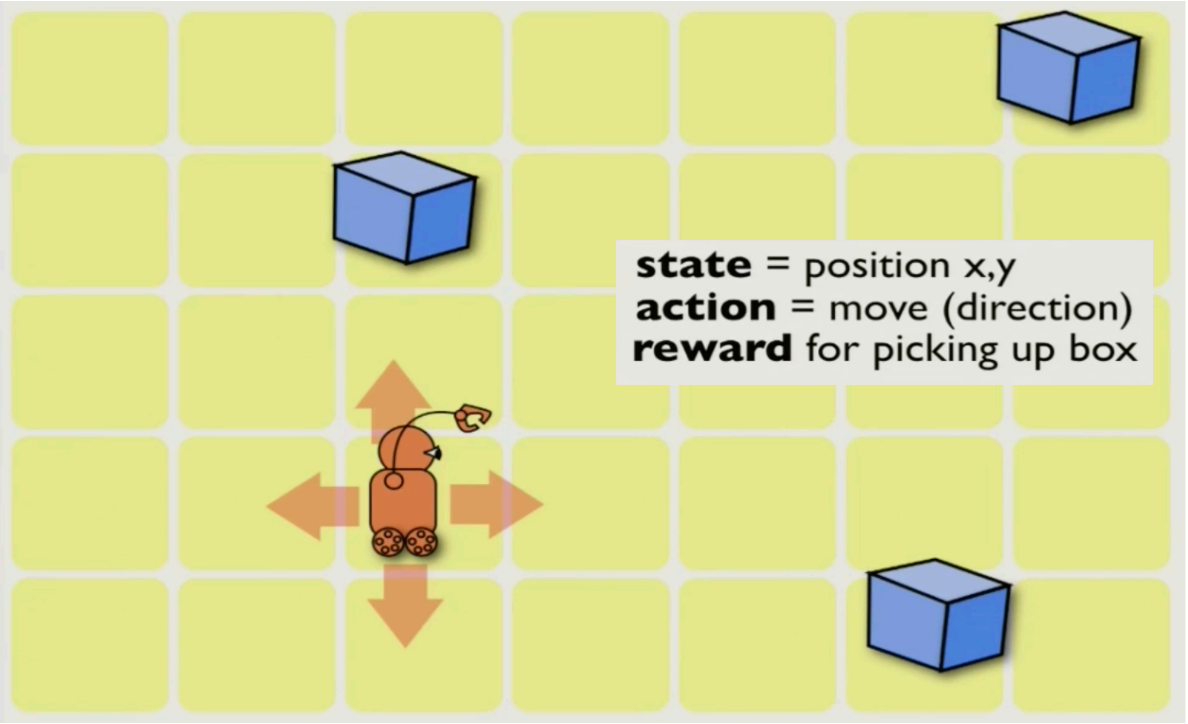


fully or partially observed

state of the environment

- The "correct" action is not known!
  (no supervised learning...)

- How to know what is right or wrong?
  $\Longrightarrow$ *Reward* system
  - can be defined only at the end...



Value-based RL algorithms: Q-learning

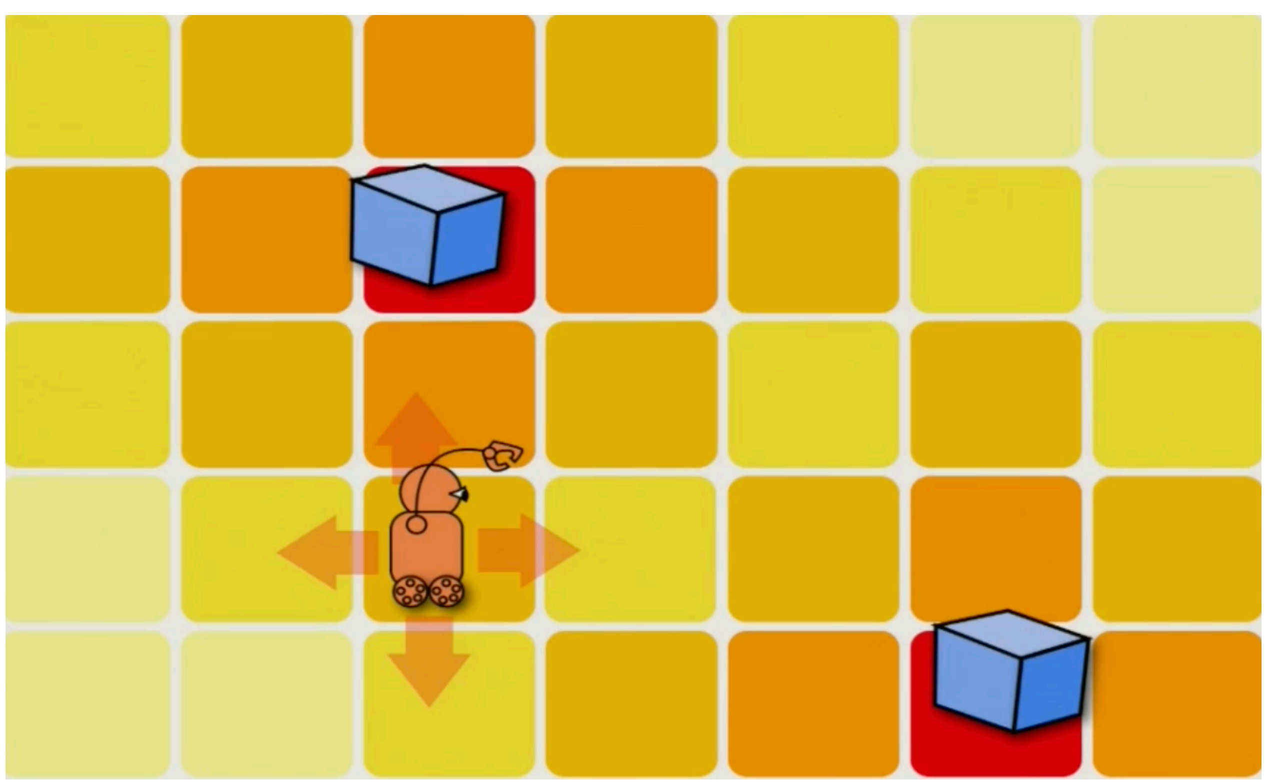
$\Longrightarrow$ *Value* (V) / Quality (Q) functions

- expected future rewards for a given state / action
  - how "valuable" is a given state / action
- discounted future reward $R_t \equiv \sum_{k=t+1}^{\infty} \gamma^{k-t-1} r_k$

$$V_\pi(s_t) \equiv \mathbb{E}\left[R_t \mid s_t\right]$$

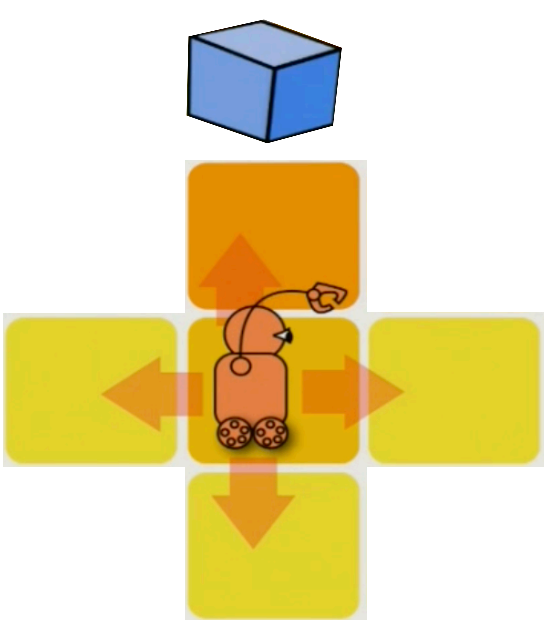
$$Q_\pi(s_t, a_t) \equiv \mathbb{E}\left[R_t \mid s_t, a_t\right]$$

"Value" of a state



"Quality" of 4 actions

"going up / down / left / right"

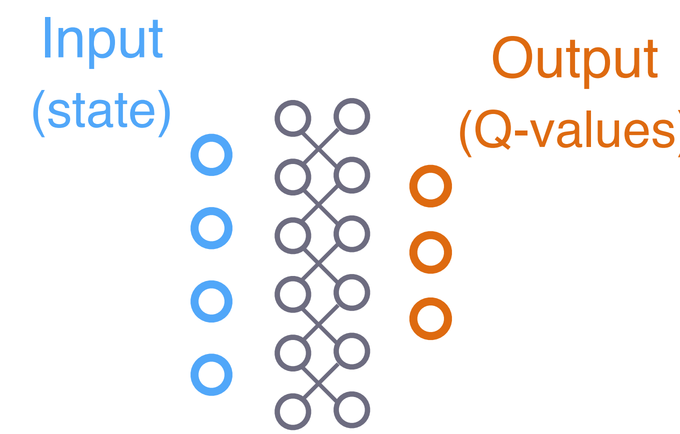


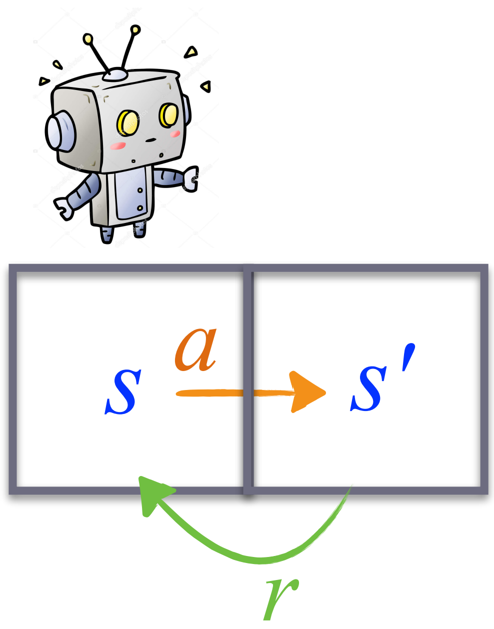
Note:

$V(s) = \max_a Q(s, a)$

Q-learning

How do we calculate the Q-function…? → **Neural Network!**





Q-learning algorithm:

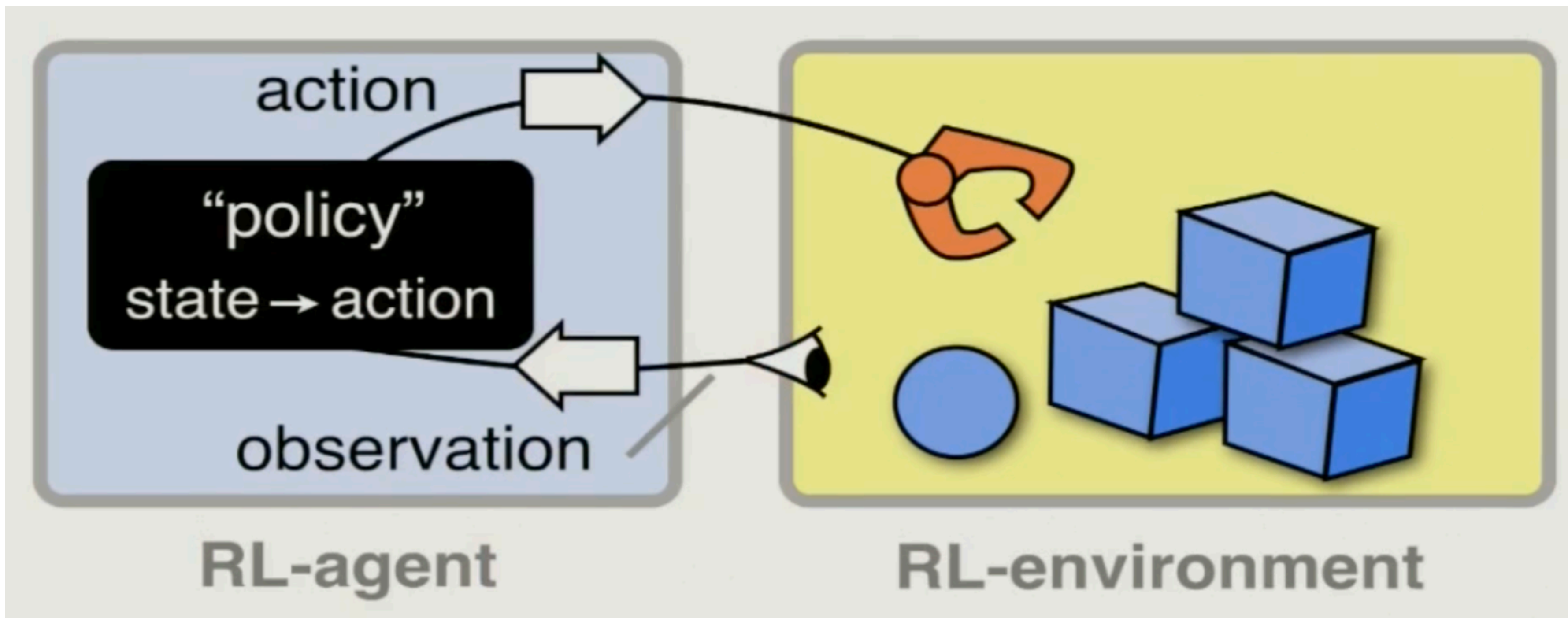
- Observe s
- Select and execute a
- Receive the reward r
- Update the Q-value: $Q^{new}(s, a) \leftarrow$

$$Q^{old}(s,a) + \alpha_n \, (\underbrace{r + \gamma \max_{a'} Q^{old}(s', a')}_{\text{target}} - Q^{old}(s,a))$$

learning rate

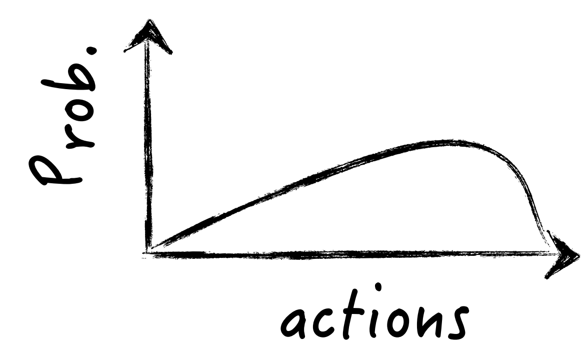
- NN update:
  - target Q(s,a): $r + \gamma V(s')$
- Deterministic policy
  - *greedy* (always pick action with best Q-value)
  - *ε-greedy* (balance between exploitation and exploration)

Policy-based RL algorithms



$\Longrightarrow$ *Policy:* $\pi_\theta(a_t | s_t)$

- probability to pick up action a_t given observed state s_t

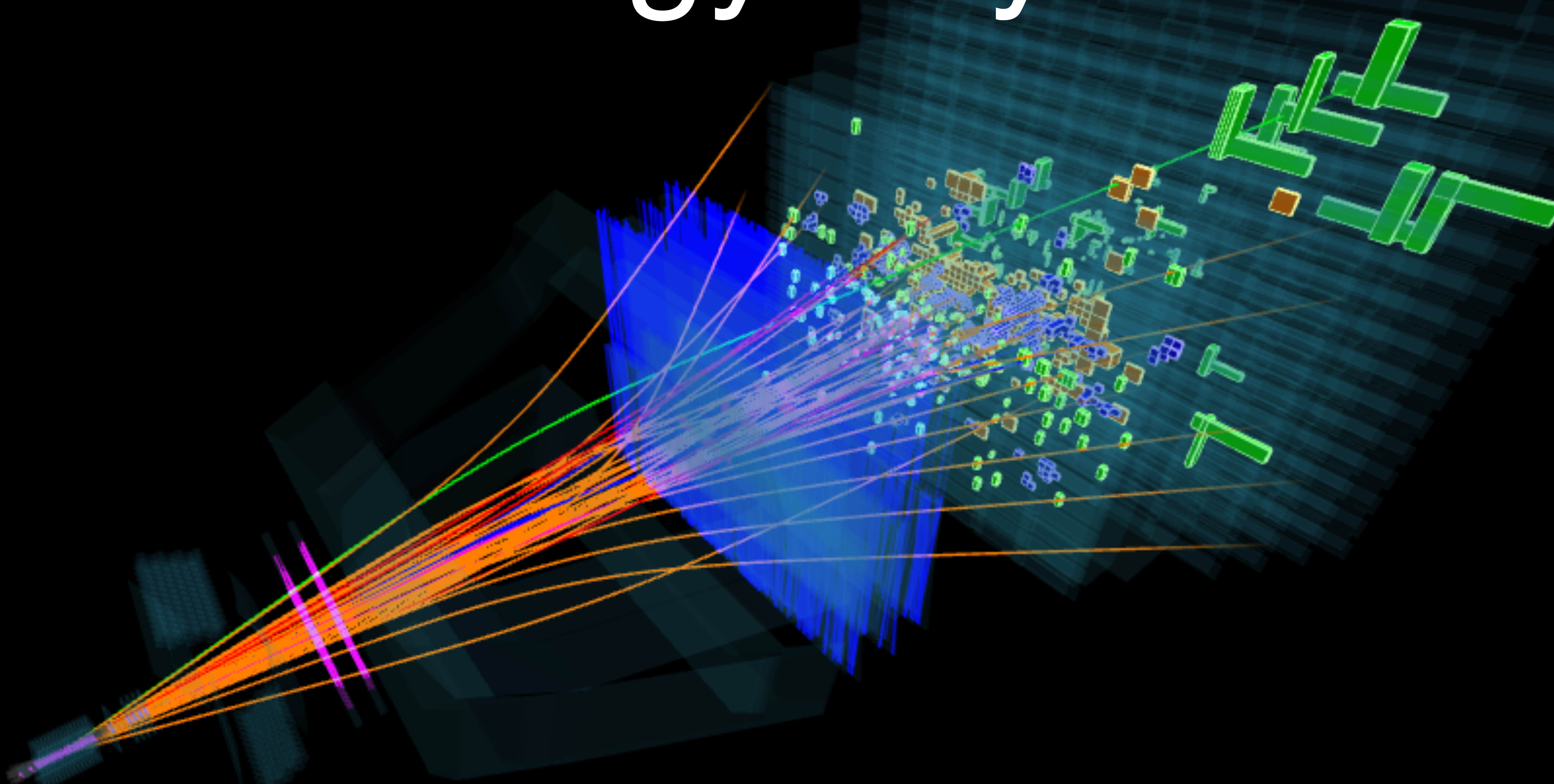


- Find the optimal policy
  - maximise the total expected reward
  - run many trajectories to get $\mathbb{E}[\ldots]$

$$\frac{\partial \bar{R}}{\partial \theta} = \sum_t \mathbb{E}[R \frac{\partial \ln \pi_\theta(a_t|s_t)}{\partial \theta}]$$

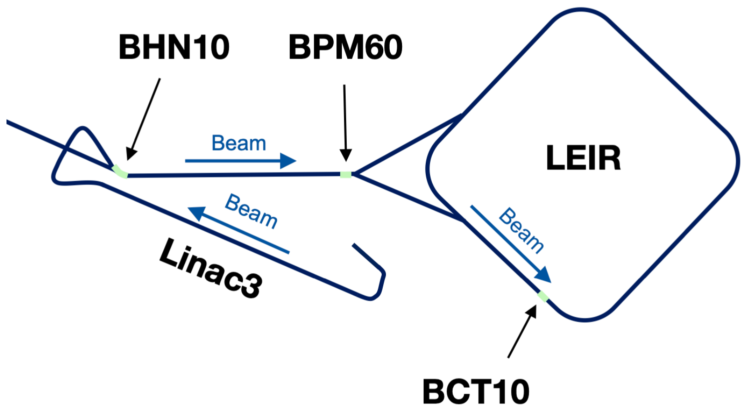

180 2 1

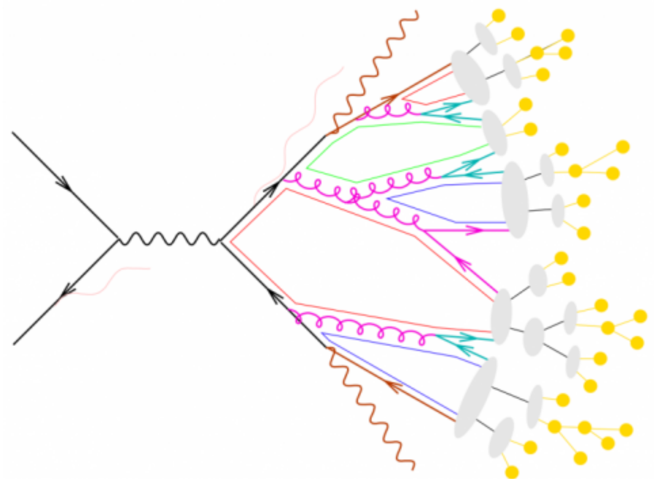
RL in High Energy Physics



RL in HEP

- Very recent development…
  - “*Automatic performance optimisation and first steps towards reinforcement learning at the CERN Low Energy Ion Ring*”, 2nd ICFA Workshop on Machine Learning for Charged Particle Accelerators (2019)
  - “*Real-time Artificial Intelligence for Accelerator Control: A Study at the Fermilab Booster*”, arXiv:2011.07371
  - “*Jet grooming through reinforcement learning*”, arXiv:1903.09644
  - “*Hierarchical clustering in particle physics through reinforcement learning*”, arXiv:2011.08191





RL to control accelerator systems arXiv:2011.07371

- Accelerator physics is often too complex to use analytical models or Monte Carlo simulations
  - requires a lot of hand tuning by experts
  - risk of hidden inefficiencies

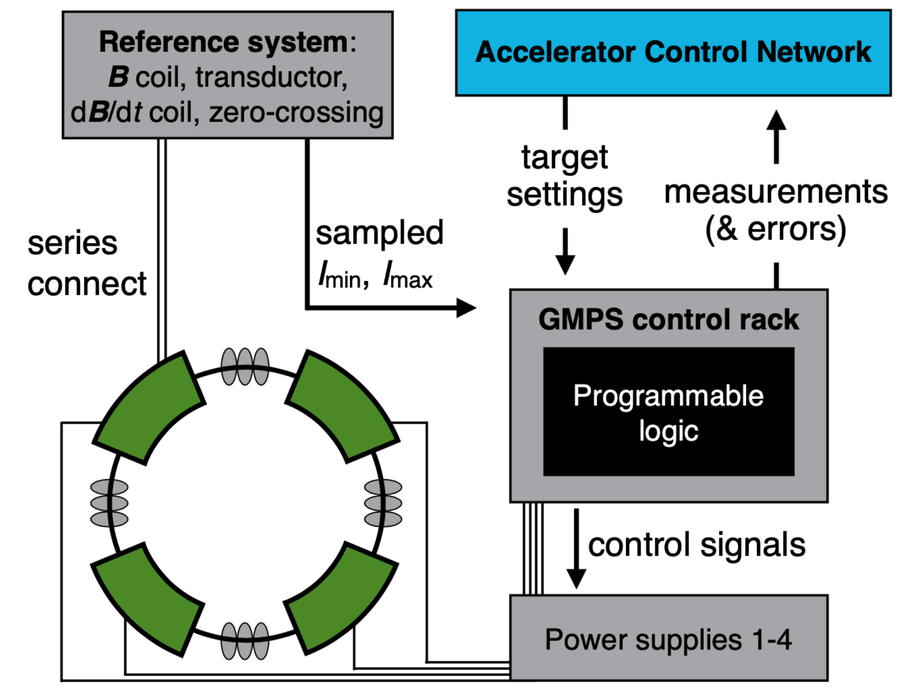
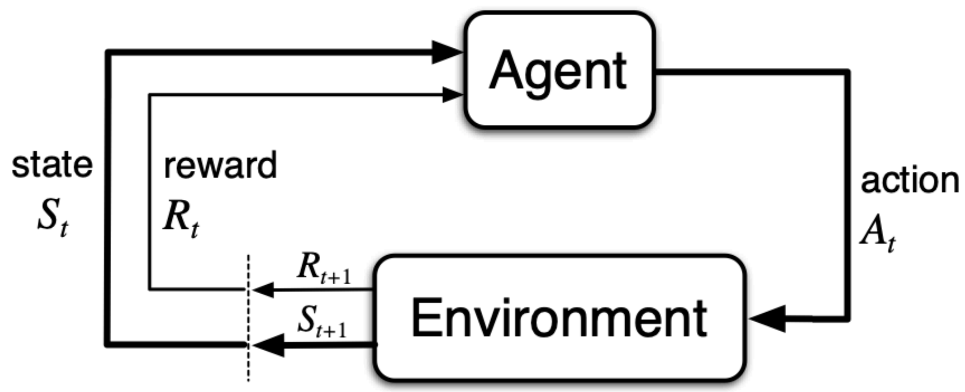


FIG. 1. Schematic view of the GMPS control environment. The human operator specifies a target program via the Accelerator Control Network that is transmitted to the GMPS control board.

Goal: adjust power supply to reach optimal condition

What is the environment?



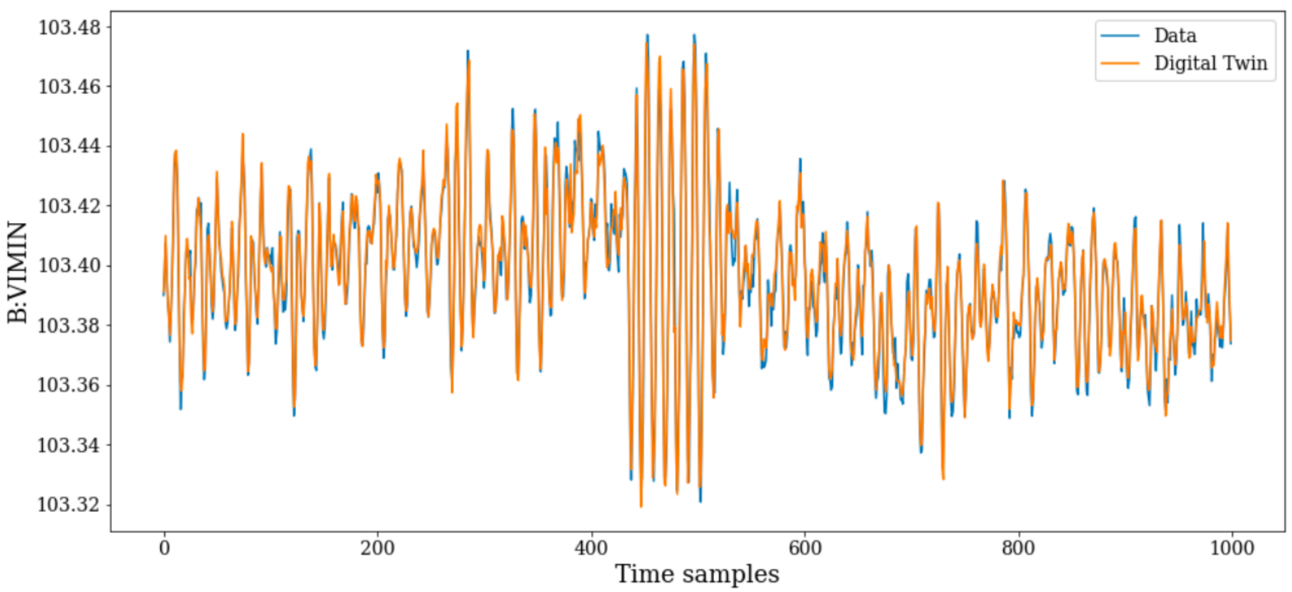
Challenge: create a model that reproduce the behaviour of the real-world system

RL to control accelerator systems arXiv:2011.07371

Two-fold application of ML

- Recurrent Neural Network (RNN) used to model the accelerator system response
  - Supervised learning on real data (time series of a set of variables)

RNN vs DATA



- on-line RL agent
  - Actions: change in the current $(0, \pm 0.001, \text{etc.})$
  - Reward: (negative) difference between the target and realized current

$$R_t \propto -\,|\,I_{real}(t) - I_{target}(t)\,|$$

RL agent: factor 2 improvement

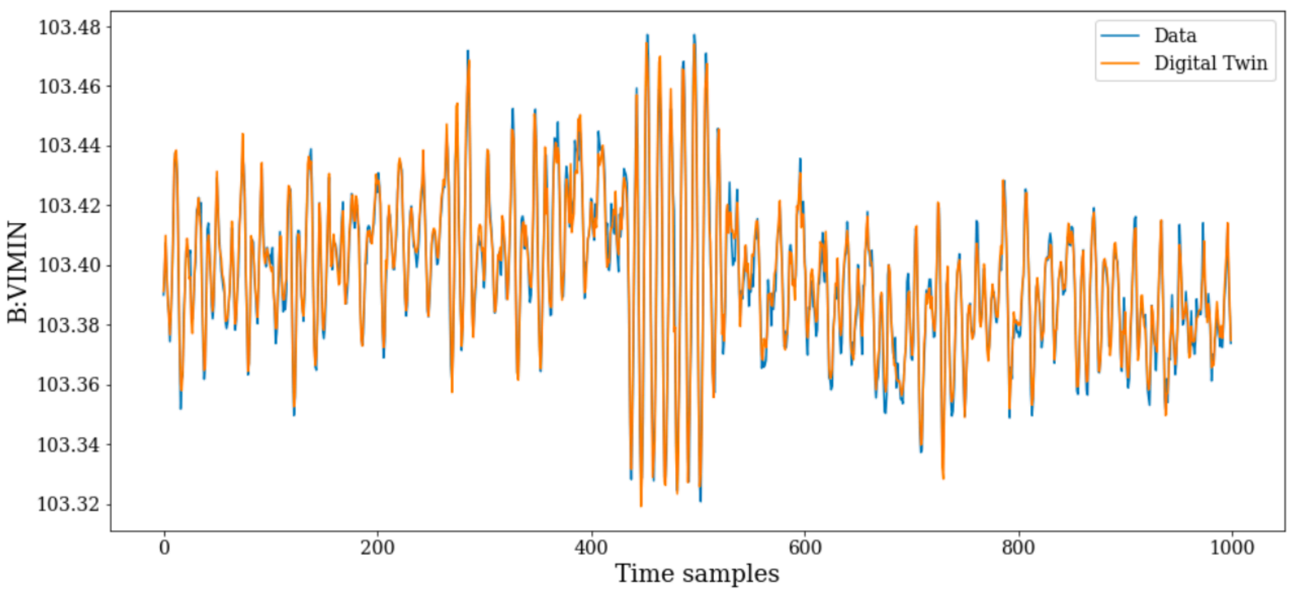
RL to control accelerator systems arXiv:2011.07371

Two-fold application of ML

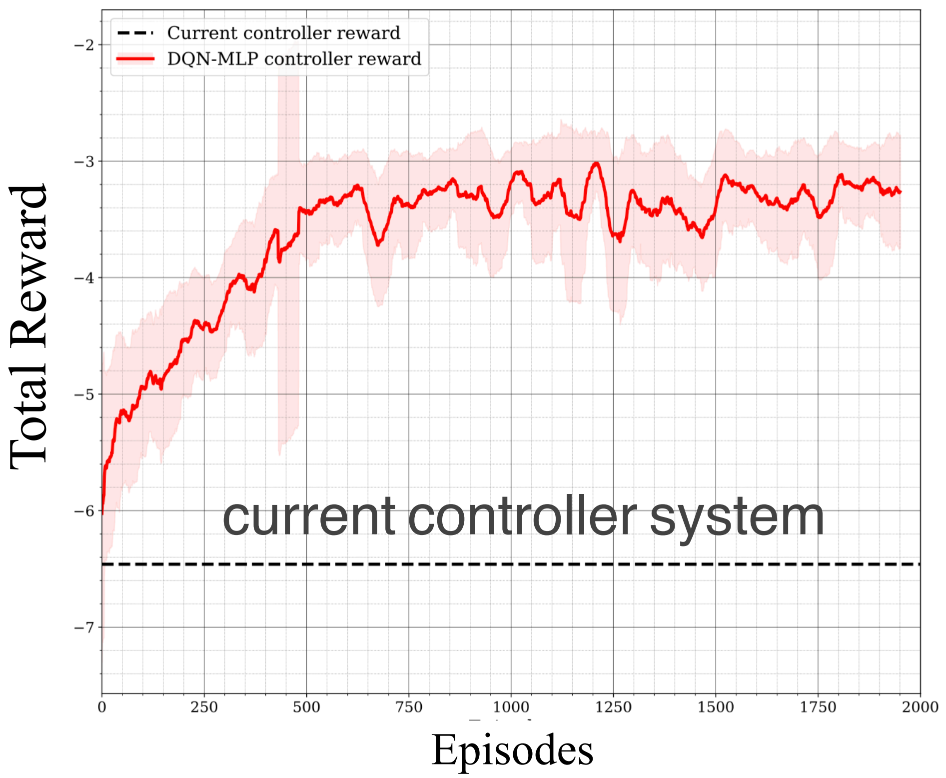
- Recurrent Neural Network (RNN) used to model the accelerator system response
  - Supervised learning on real data (time series of a set of variables)
- on-line RL agent
  - Actions: change in the current $(0, \pm 0.001, \text{etc.})$
  - Reward: (negative) difference between the target and realized current

$$R_t \propto -|I_{real}(t) - I_{target}(t)|$$

RNN vs DATA



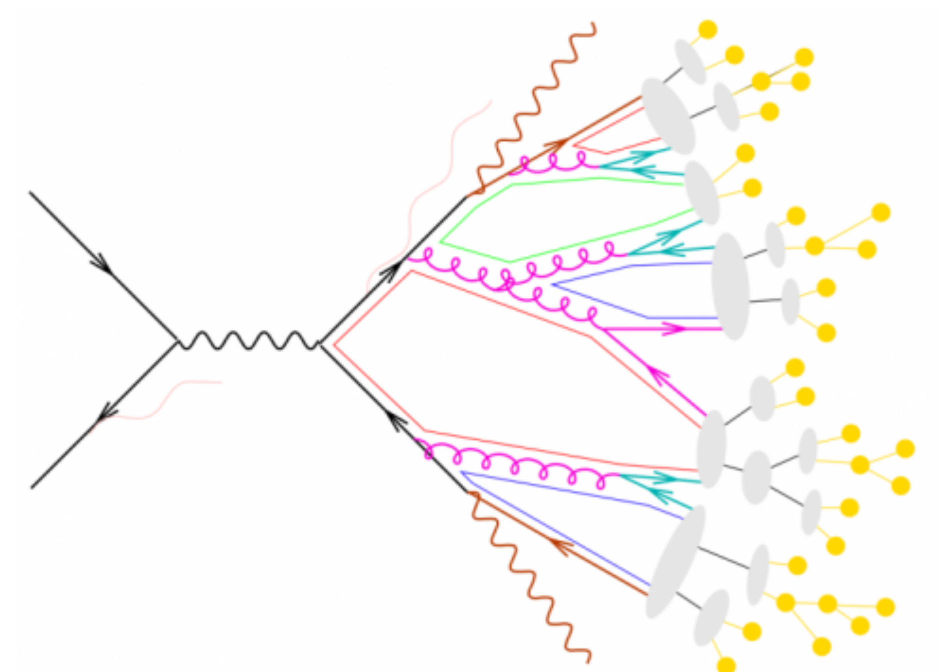
RL agent: factor 2 improvement



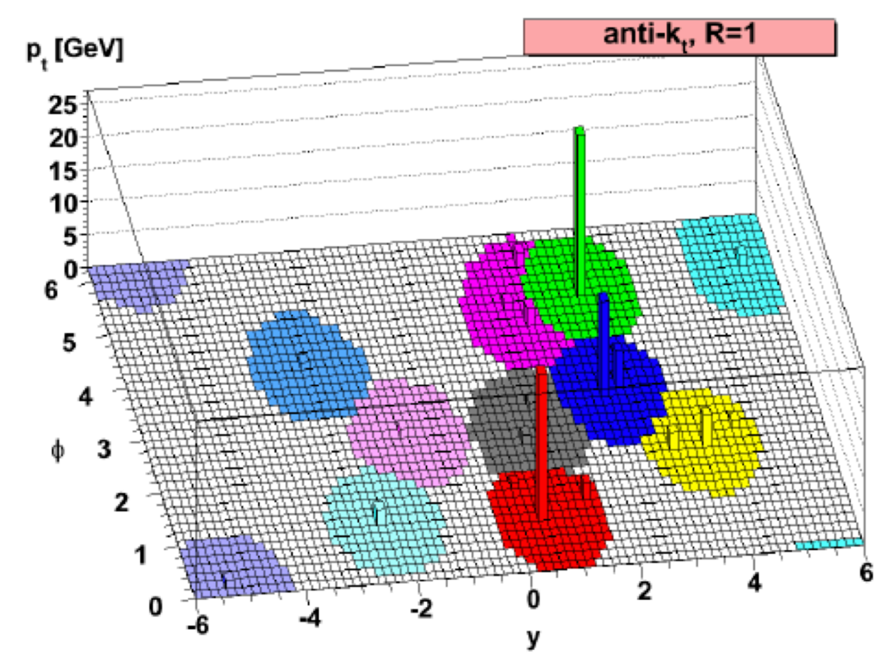
RL with jets

Jets are the result of the hadronization of quarks and gluons produced at collider experiments

$\Longrightarrow$ Reconstructing the properties of the original elementary particle is a fundamental step in data analysis



Challenge: even knowing the QCD splitting probabilities, the **large combinatorial** space make impossible to find the true maximum-likelihood solution



Popular clustering algorithms

- k_T, Cambridge-Aachen, anti-k_T

$$d_{i,j} = min(p^a_{T,i}, p^a_{T,j})\frac{\Delta R_{i,j}}{R}$$
$$d_{i,B} = p^a_{T,i}$$

$\Longrightarrow$ greedy & heuristic

RL for clustering jets arXiv:2011.08191

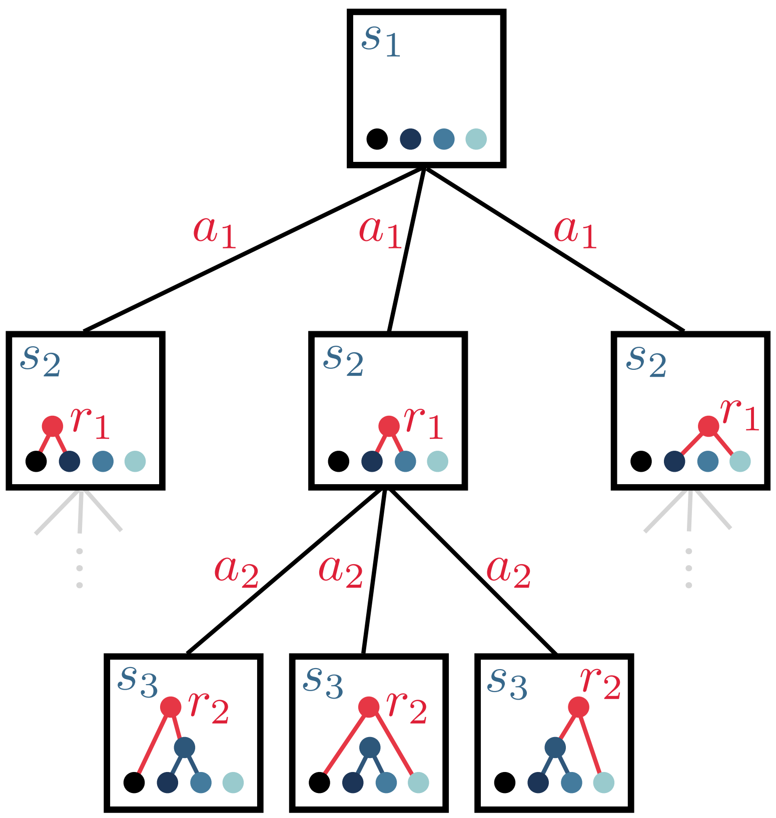
Goal: reconstruct the most plausible binary tree of particle splittings

State: particle's four-momenta $s = \{p_1, \ldots p_N\}$

Action: choice of two particles $a = (i, j)$ to be merged

Reward: splitting probabilities $R(s,a) = \log p_s(s_t|s_{t-1})$

Episode ends when only one particle is left

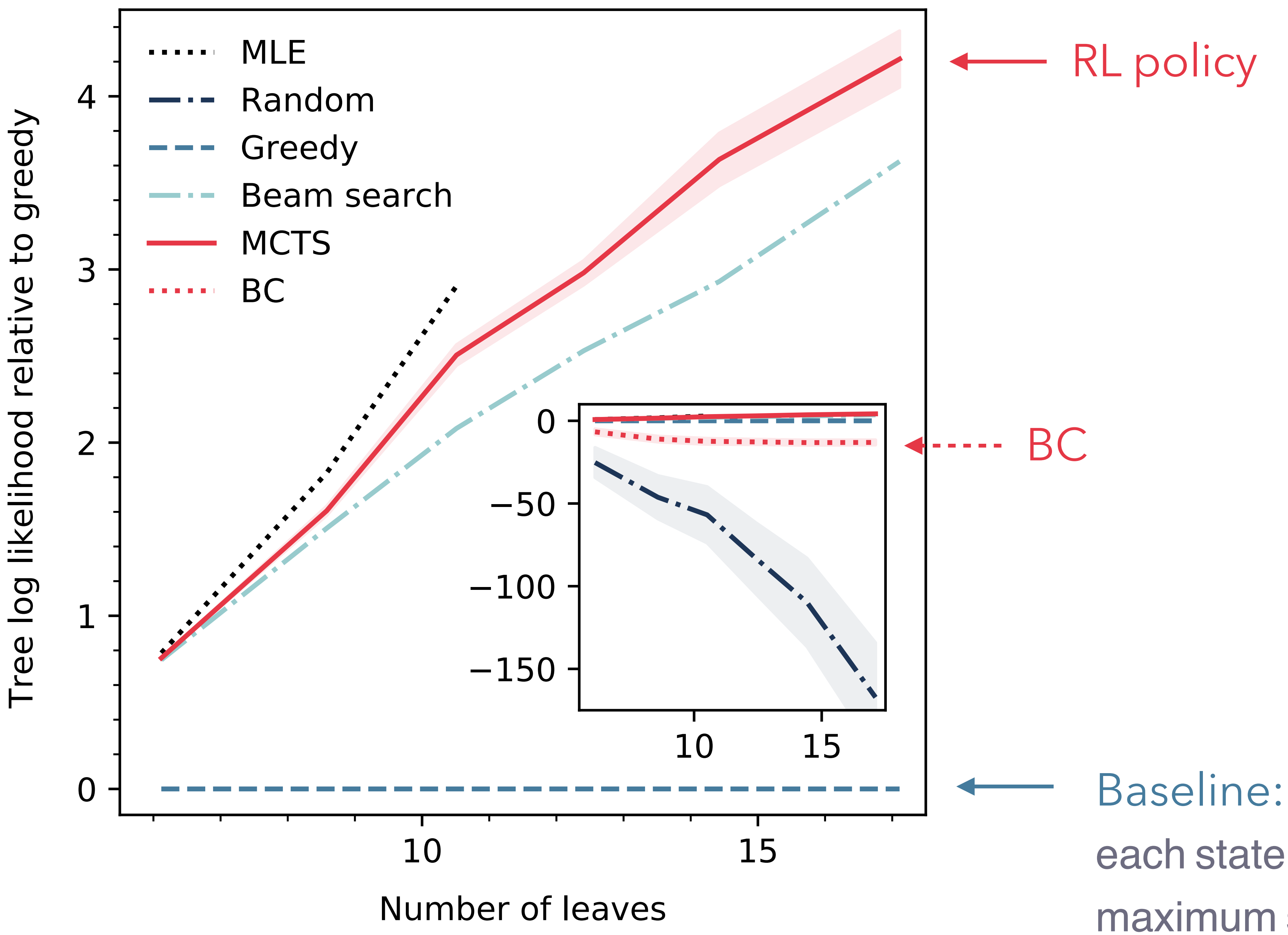


Train policy network that

(1) lead to the largest reward (via MCTS)

(2) imitate the true actions $\Longrightarrow$ Behavioural cloning (BC)

RL for clustering jets arXiv:2011.08191



Baseline: greedy algorithm that at each state picks the action with the maximum splitting likelihood p_s

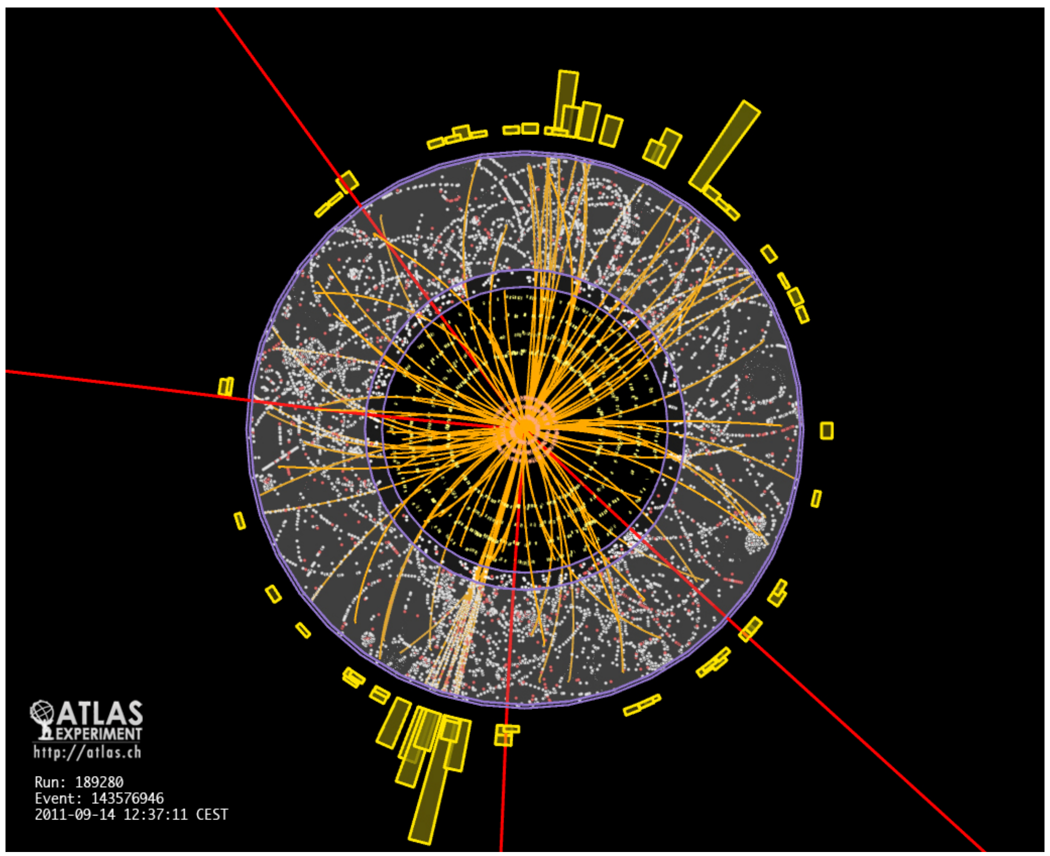
DFEI: Deep Full Event Interpretation with RL

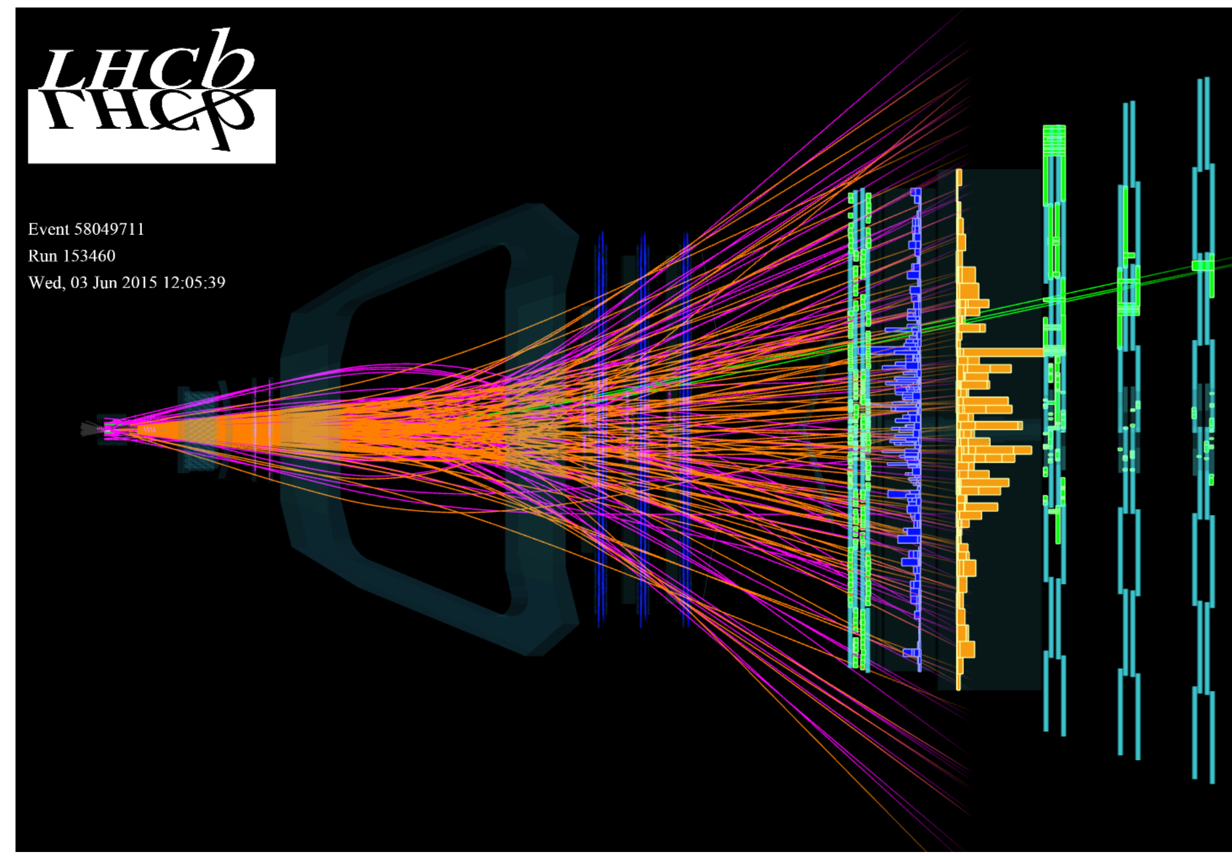
Julian Garcia Pardinas[(1)], Andrea Mauri[(1)], Marta Calvi, Jonas Eschle, Simone Meloni, Nicola Serra

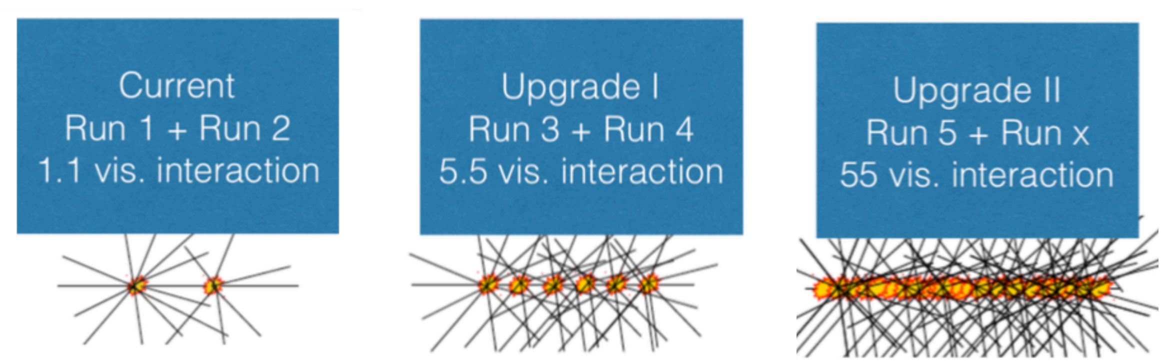
DFEI receives funding from H2020-MSCA-IF program [(1)] and SNSF PostDoc Mobility program [(2)]

Full Event Interpretation

- RL can be very useful in case of large combinatorial space





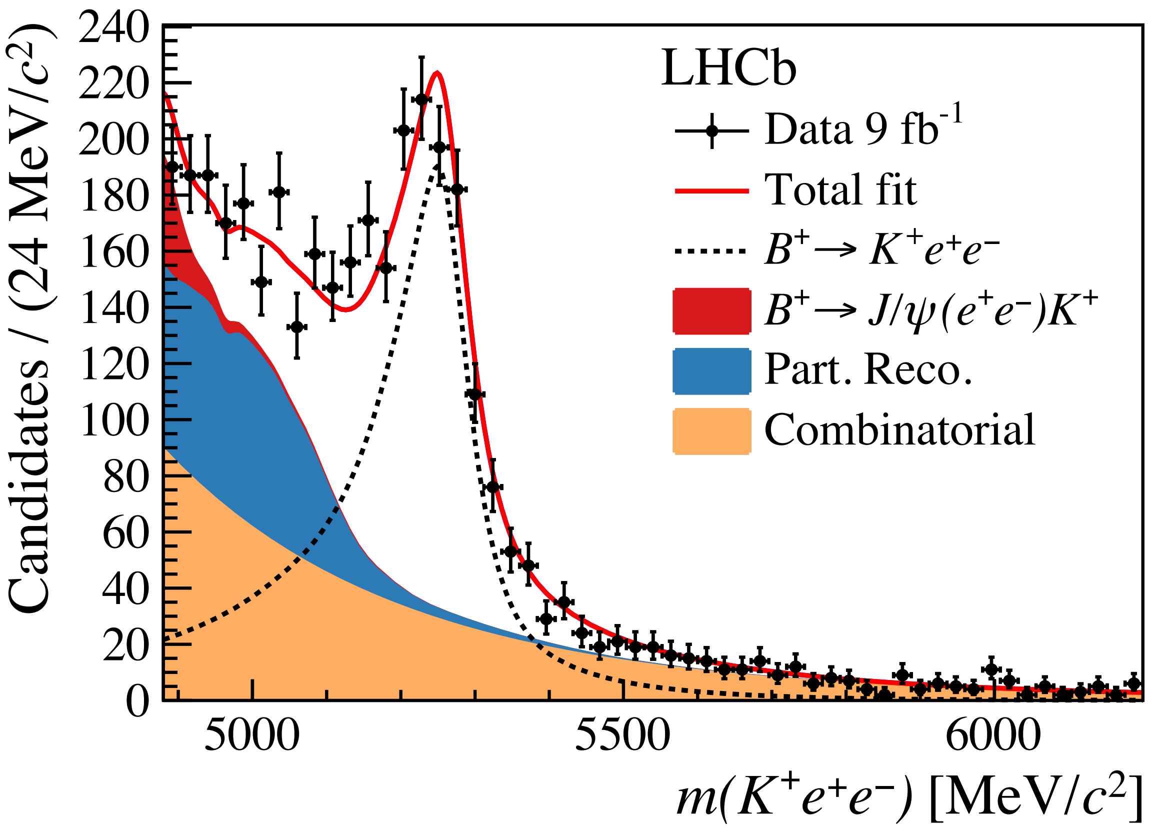


Full Event Interpretation

- So far in physics analysis at LHCb, the selection of signal candidates only focuses on a given decay mode
  - *e.g.* $B^+ \to K^+e^+e^-$

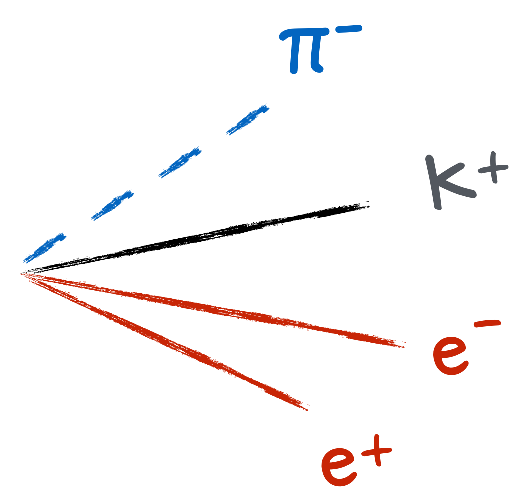
$\Longrightarrow$ select events that pass some smart criteria

arXiv:2103.11769



Partially reconstructed decays:

$$B^0 \to K^+\pi^-e^+e^-$$



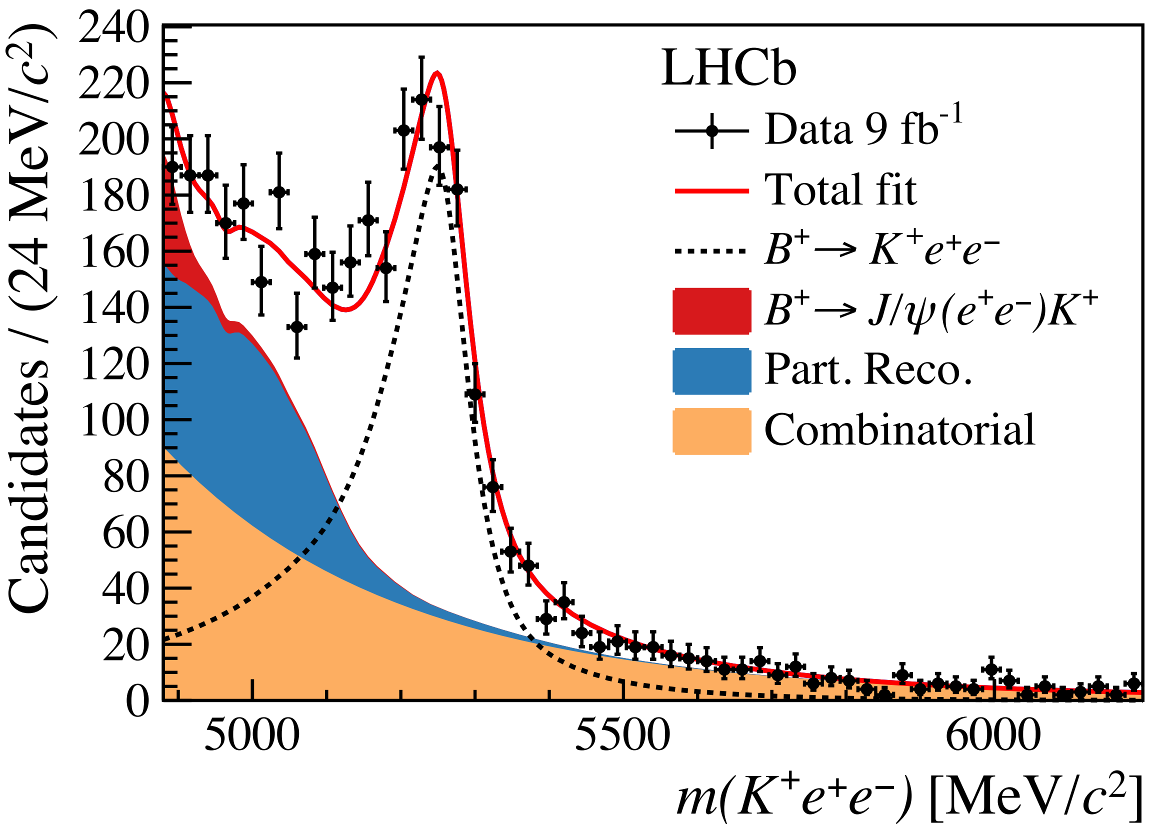
The pion is really not reconstructed or we simply don't look for it…?

Full Event Interpretation

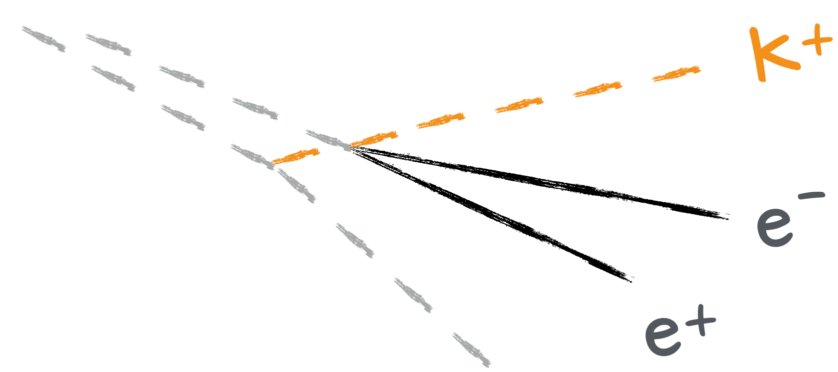
- So far in physics analysis at LHCb, the selection of signal candidates only focuses on a given decay mode
  - *e.g.* $B^+ \to K^+e^+e^-$

  $\Longrightarrow$ select events that pass some smart criteria

arXiv:2103.11769



Combinatorial background
(random combination of tracks)

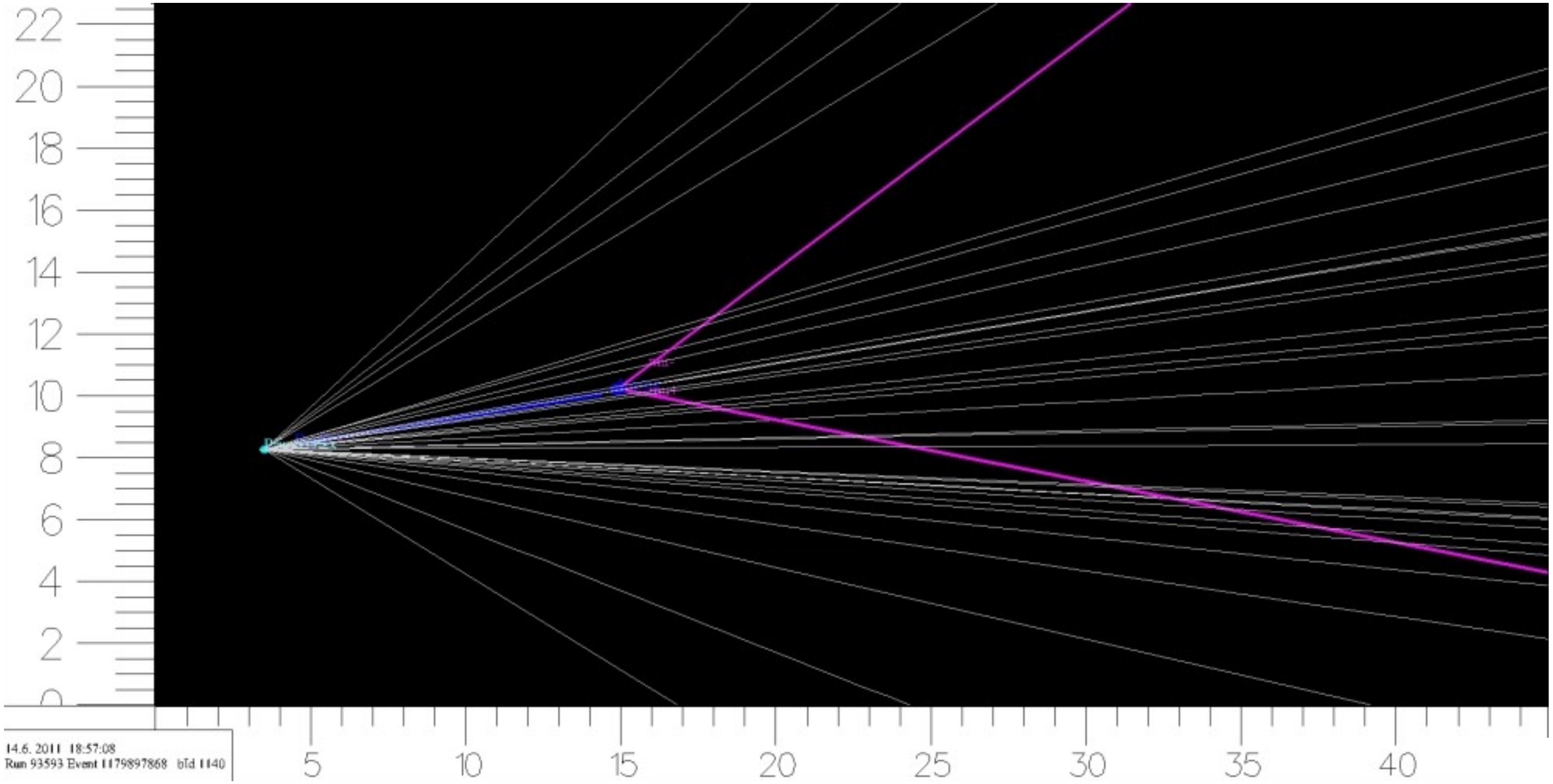


mix of decay of the other b-hadron
\+ rest of the event

Full Event Interpretation

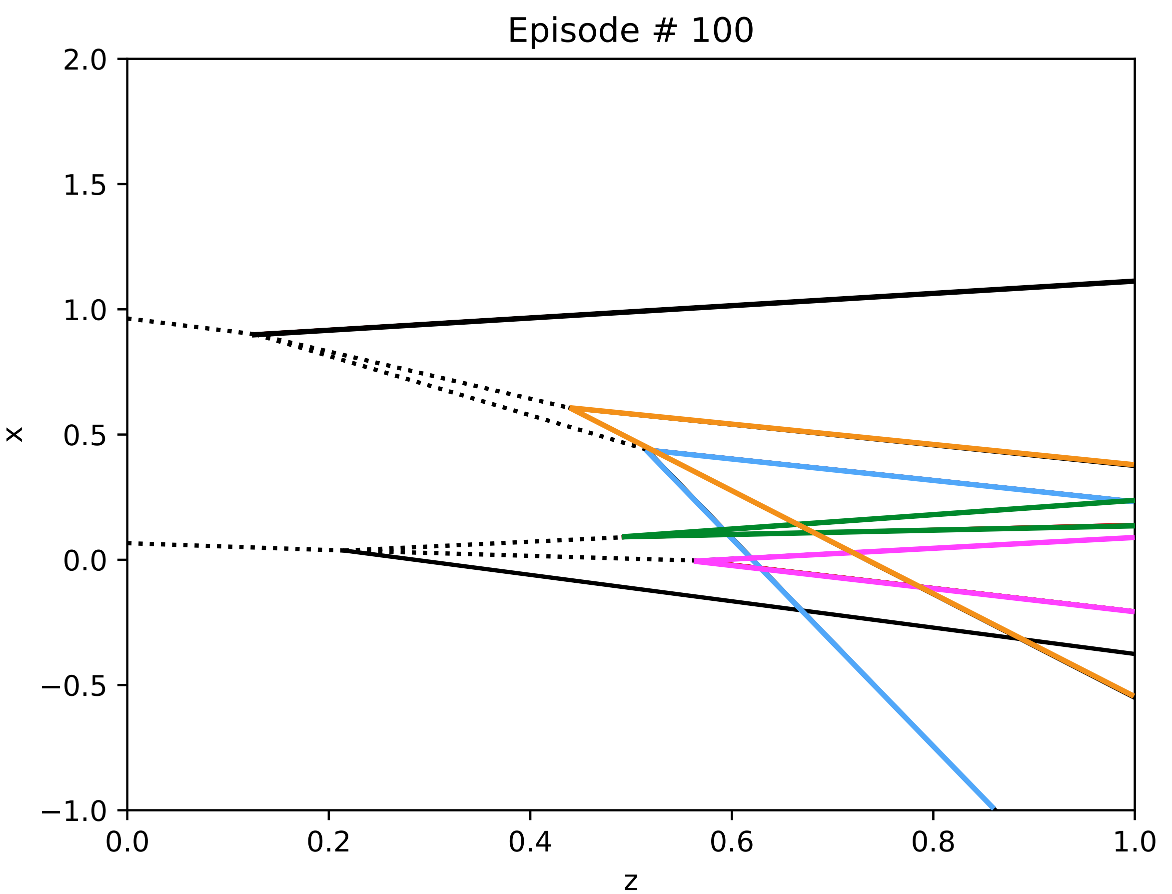
⟹ Zoom out and try to look the full event!

A **Full Event Interpretation** will create a new paradigm in event selection and signal identification



DFEI: how…?

- Goal: reconstruct most likely decay chains in the event
- State: final state particle (kinematics + PID)
- Actions: match pairs of particles
- Reward: quality of the reconstructed vertex



Promising applications:

$\Longrightarrow$ off-line: improving signal efficiency / background rejection

$\Longrightarrow$ on-line: triggering on interesting objets / identify relevant set of final state particles to be saved on disk for off-line data analysis

Work in progress…

Conclusion

- RL can provide powerful algorithm to solve highly combinatorial problems
- First application in different HEP domains seems promising
- Ongoing studies on the development of Full Event Interpretation RL algorithms

  $\Longrightarrow$ Final level: unlimited!