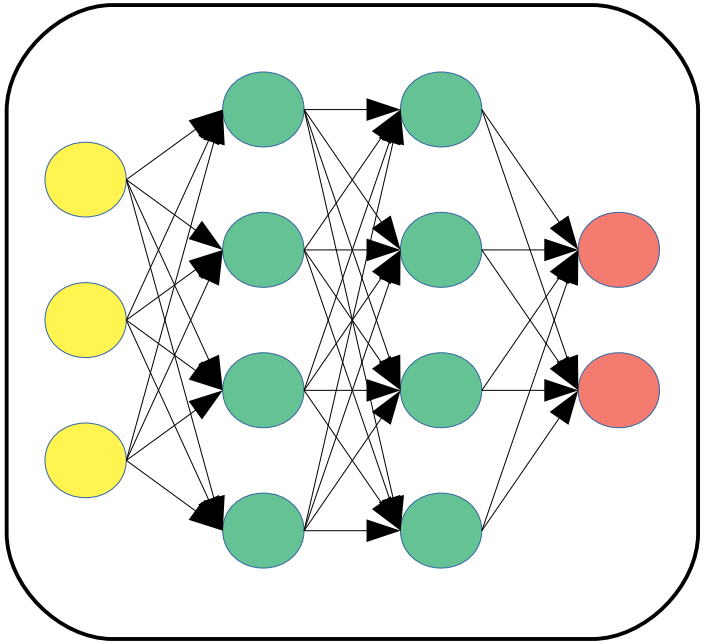
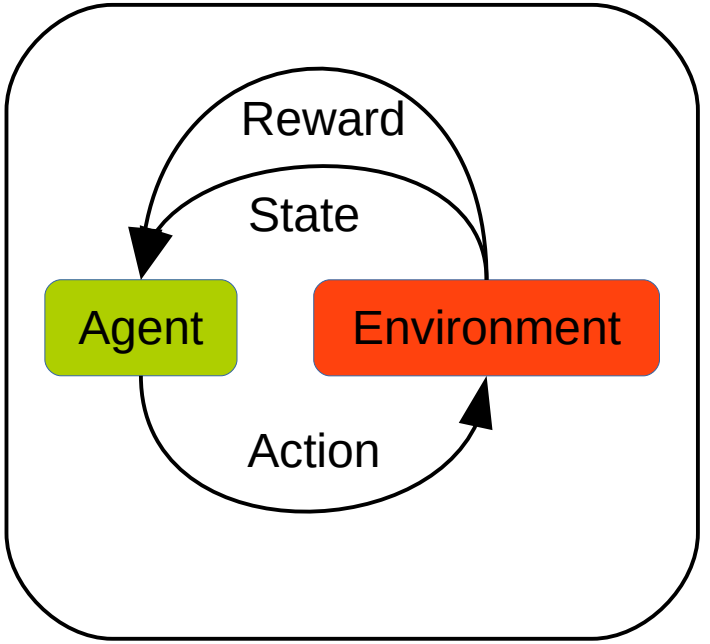
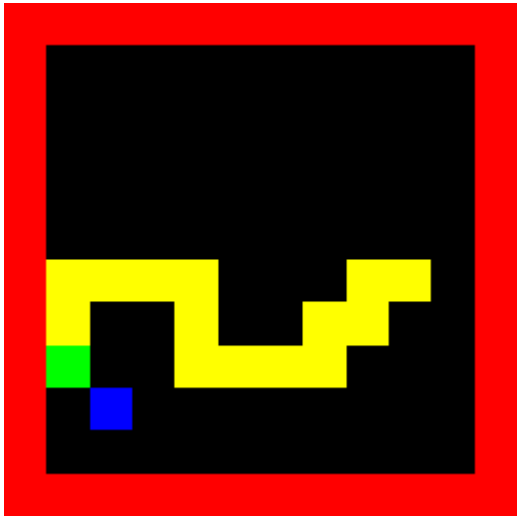


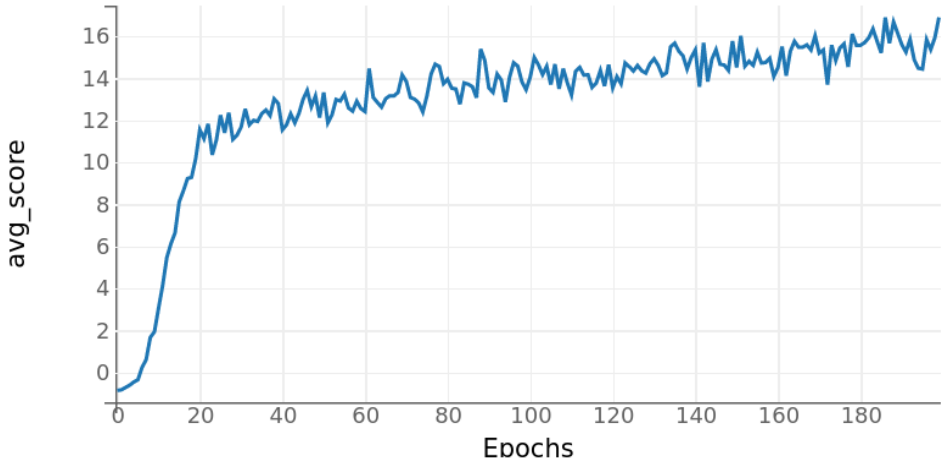


Deep Reinforcement Learning

Talk & Live-Demo
27.10.2020 19:00
Linux User Group Frankfurt

Live Demo: Snake



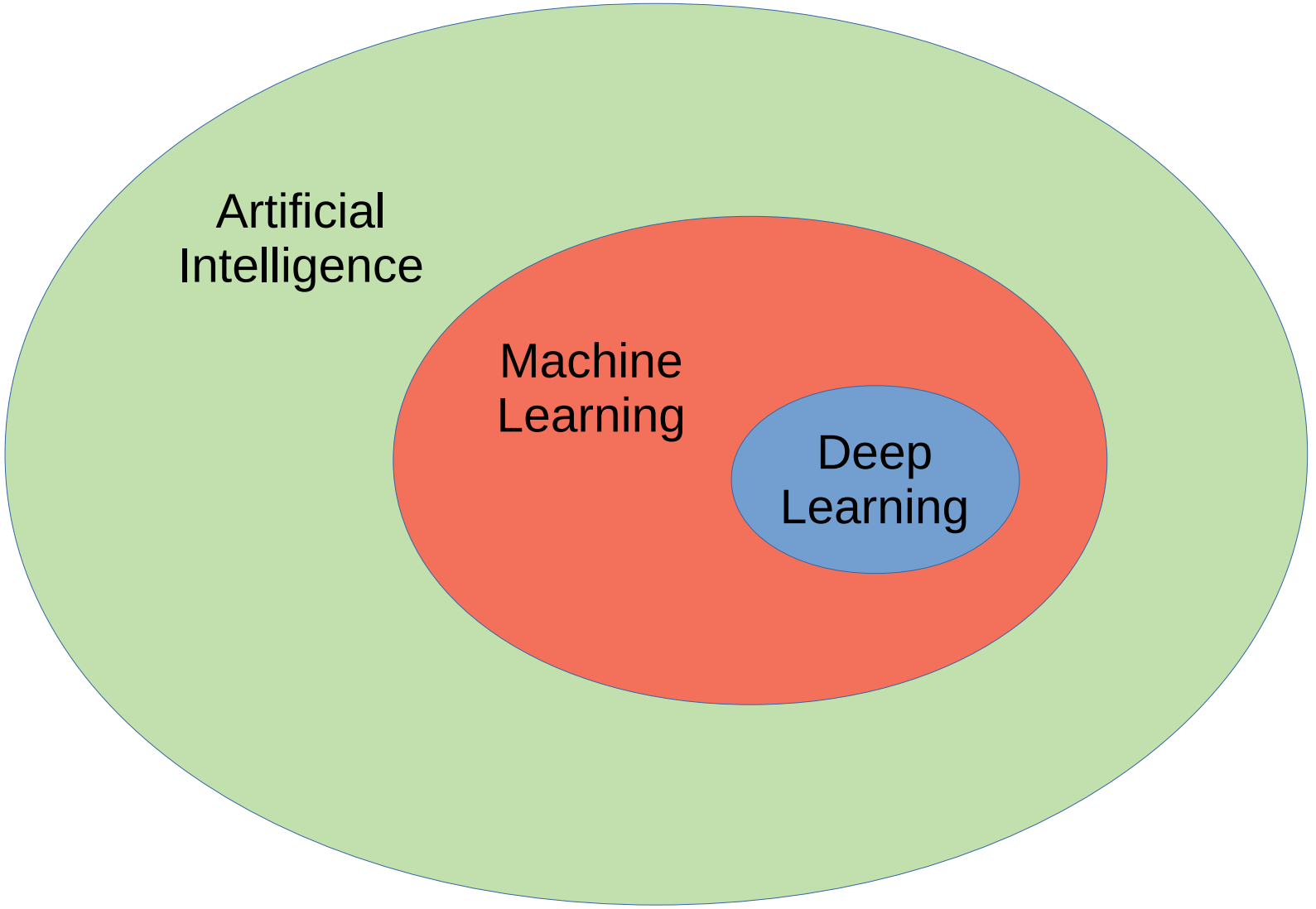


Source Code:

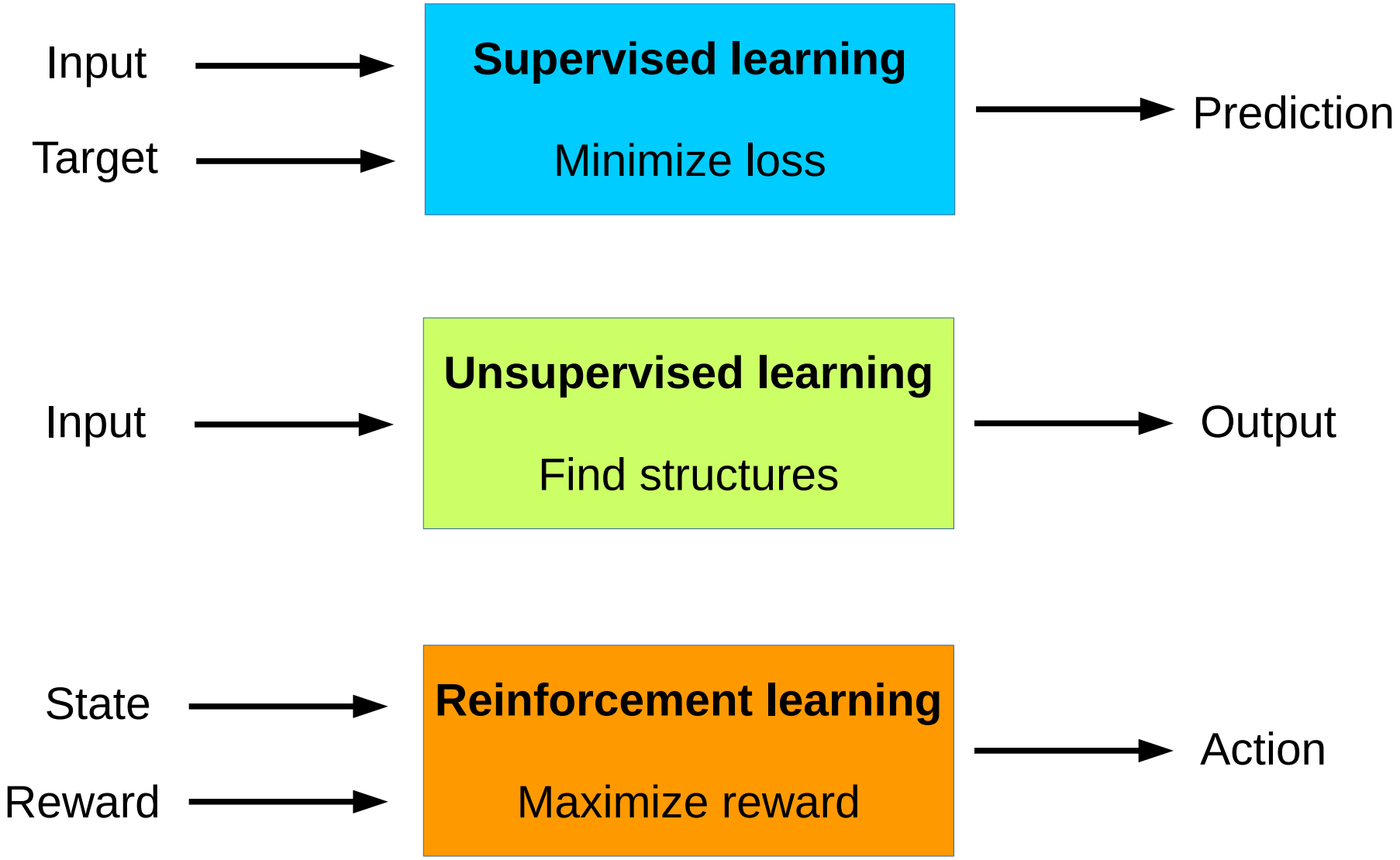
https://github.com/SteffenBauer/Deep_RL

- **Part 1: Artificial Intelligence**
- **Part 2: Examples**
- **Part 3: Reinforcement Learning**
- **Part 4: Limitations**

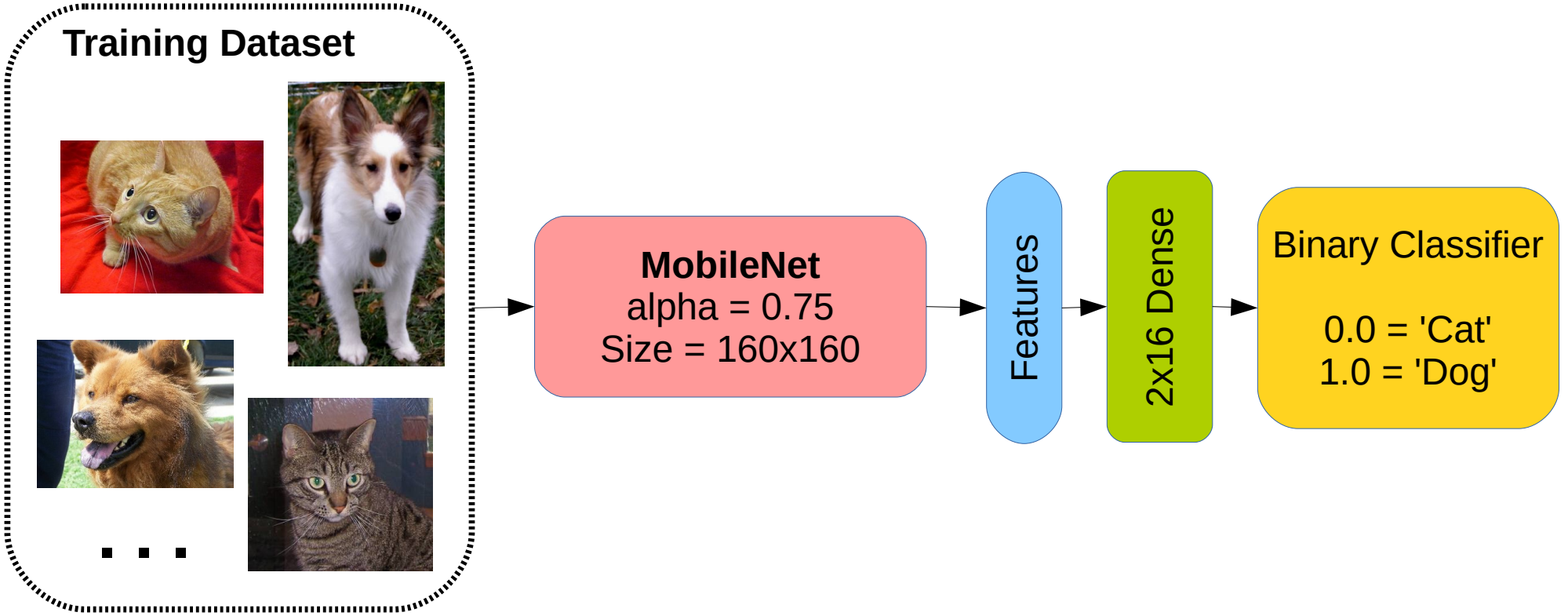
Artificial Intelligence, Machine Learning & Deep Learning



Learning methods in AI



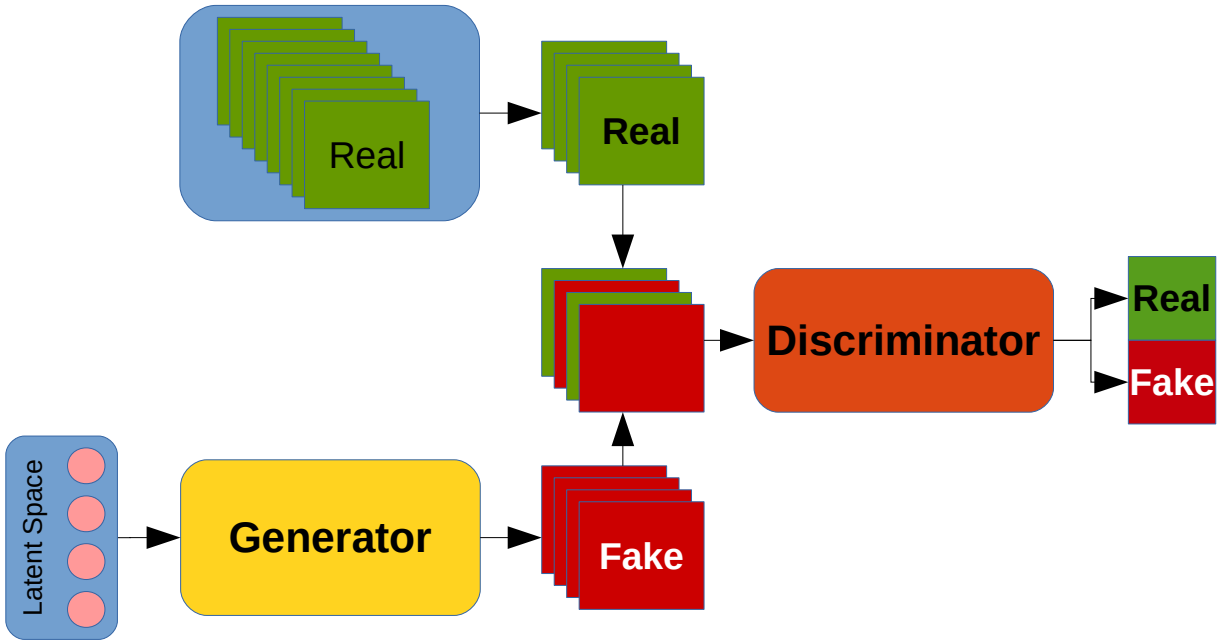
AI Talk 1: Supervised learning “Dogs versus Cats”

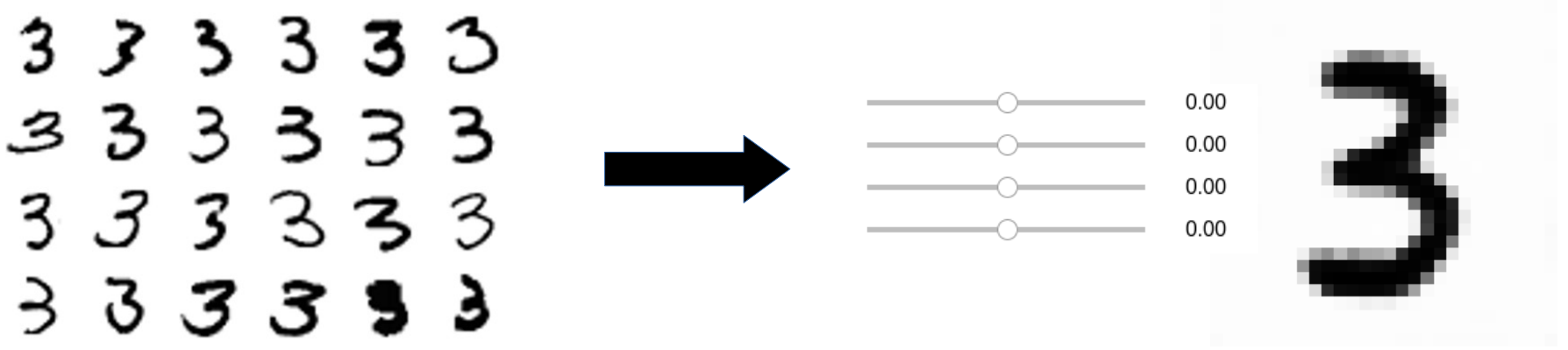


Inference

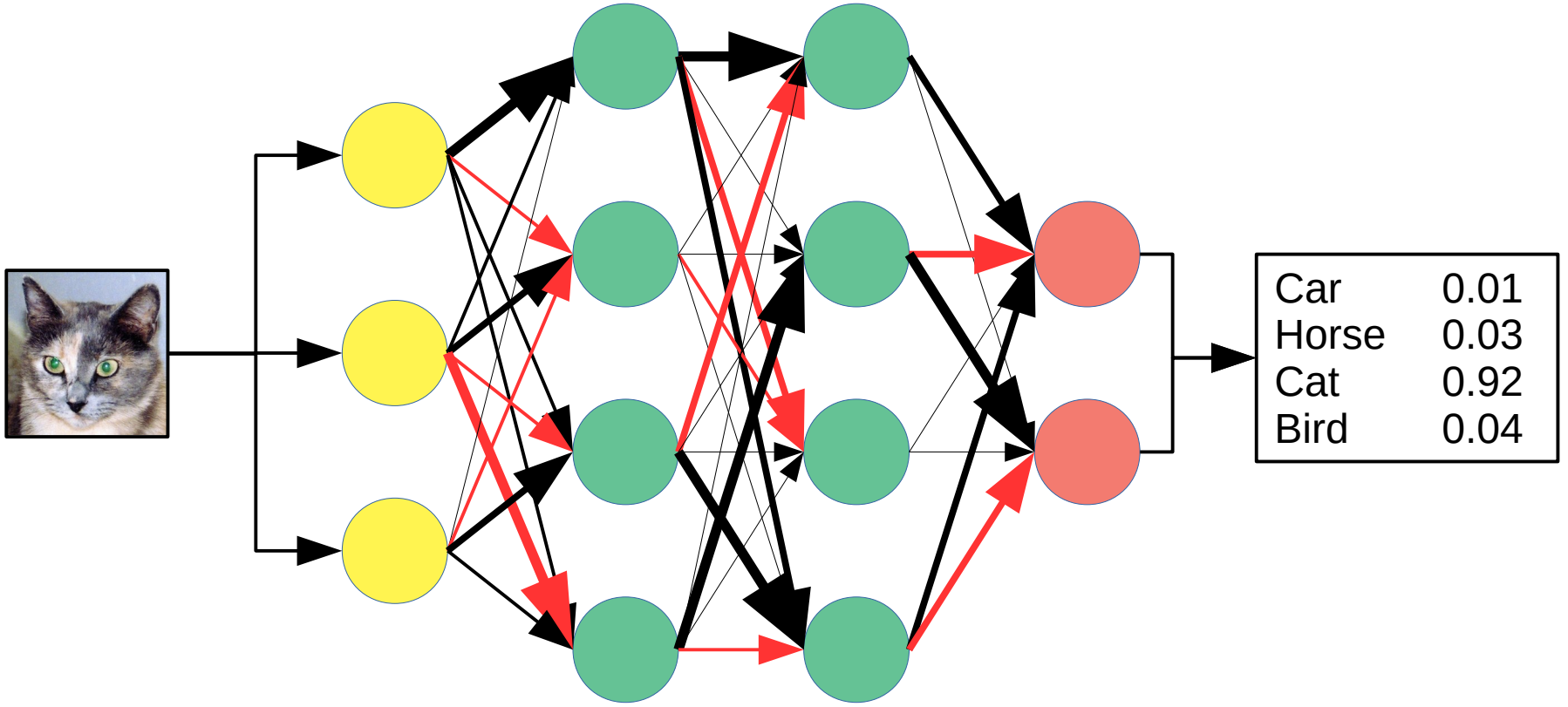
```
------------------------
 Class        | Score
------------------------
 Dog          |  0.01%
 Cat          | 99.99%
------------------------
 Result:        Cat
------------------------
```

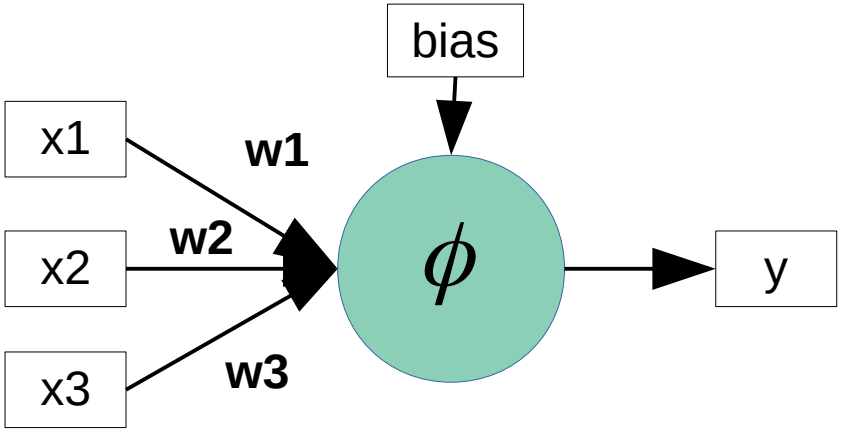
AI Talk 2: Unsupervised learning “Generative Adversarial Network”





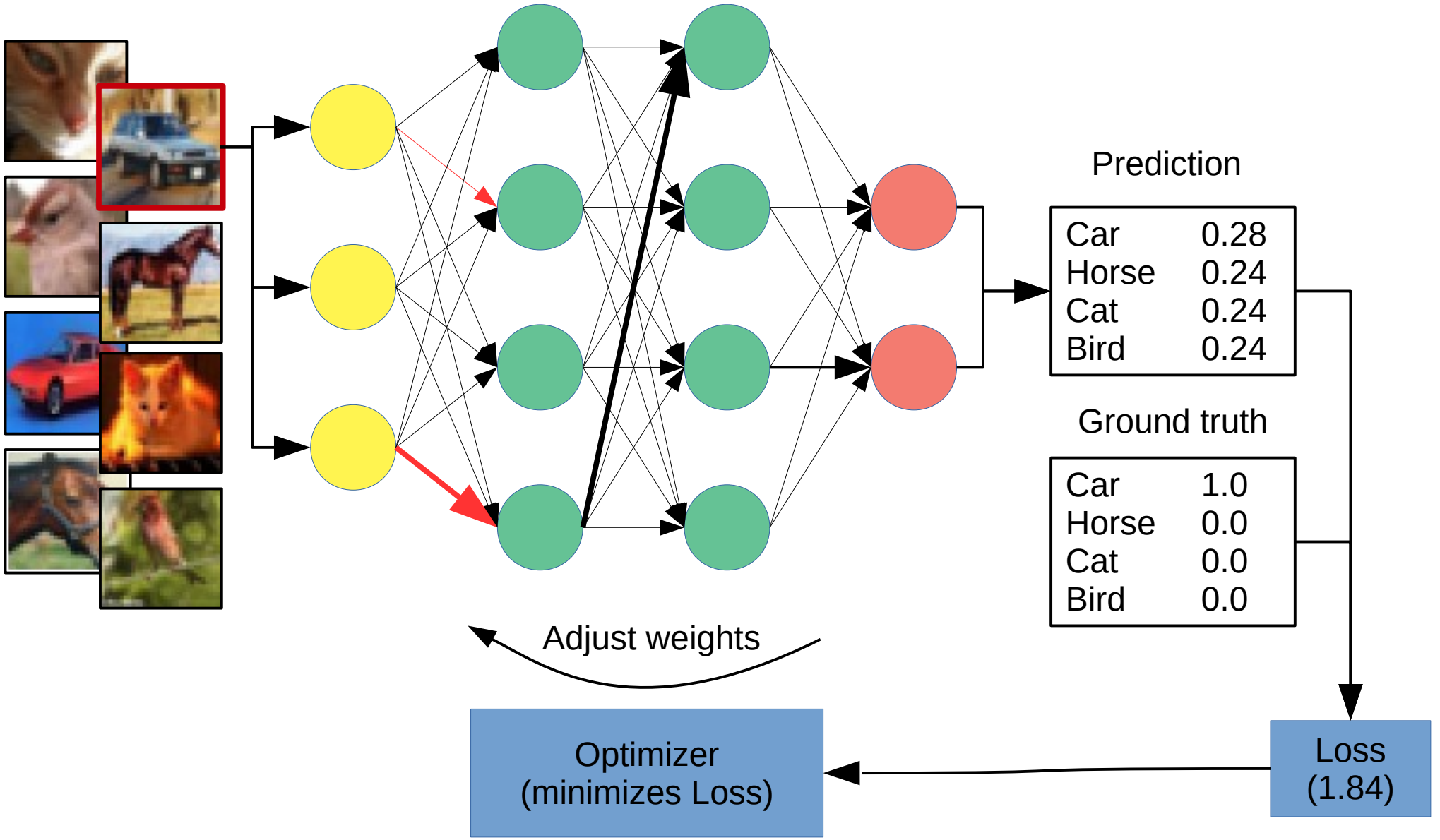
Neural networks



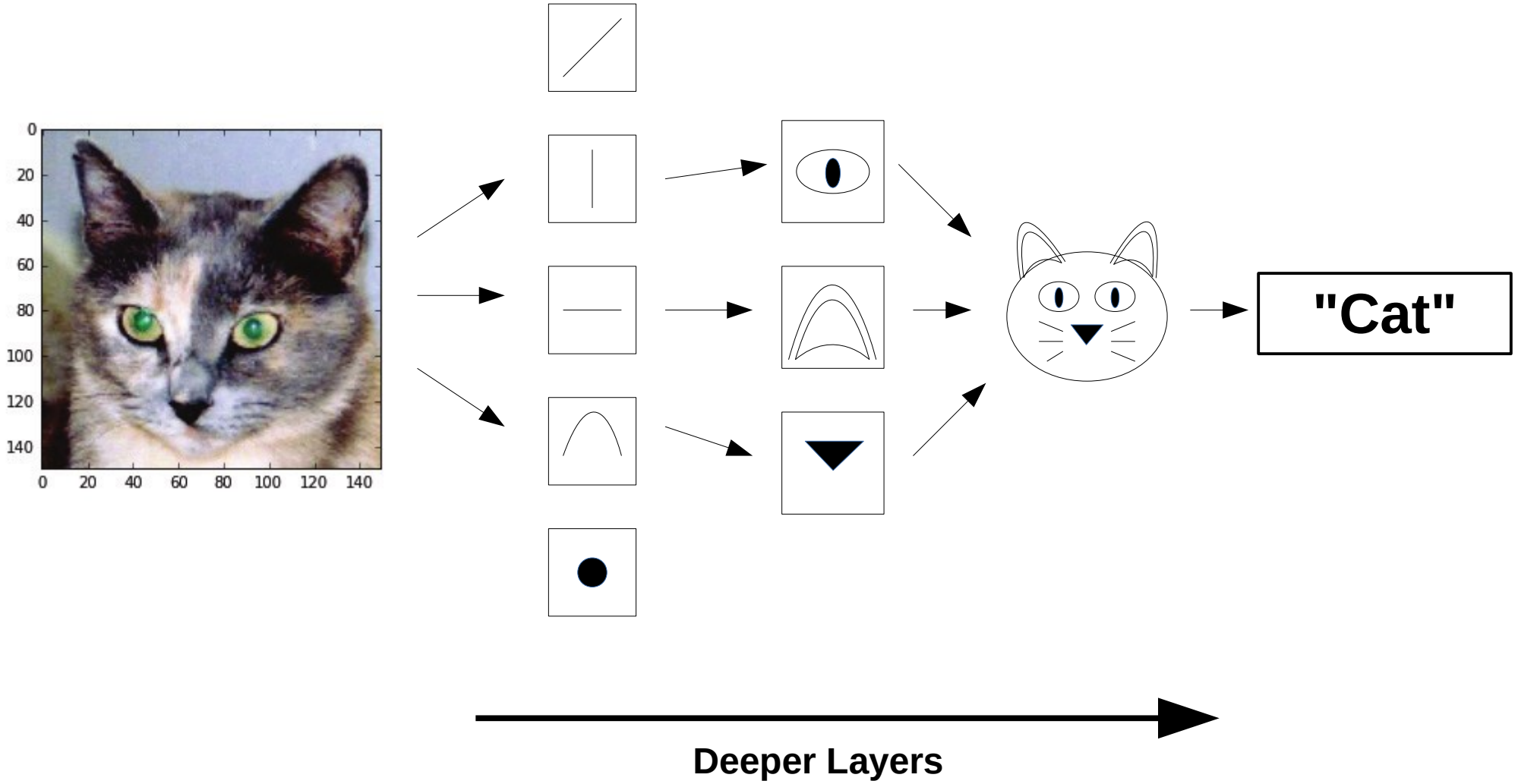


$$y = \phi\left(\left(\sum^{n} x_n \cdot w_n\right) + b\right)$$

Training a neural network



“Deep” in deep learning: Abstraction hierarchy



Examples of Reinforcement Learning

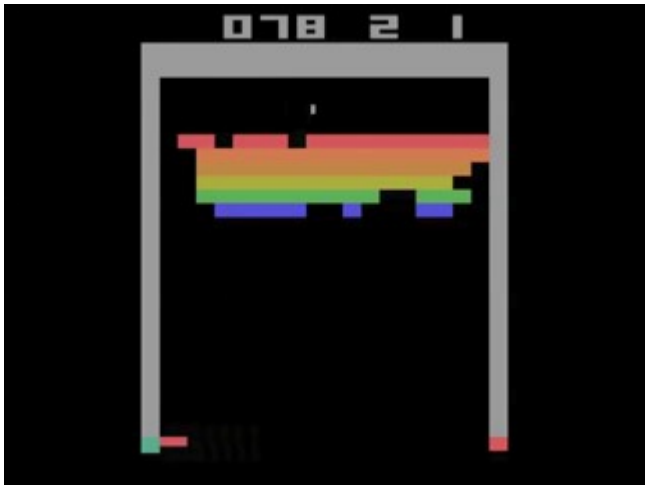
Nature 2015/2016

DeepMind 2015: Self-taught human-level videogame playing

100 Training Episodes

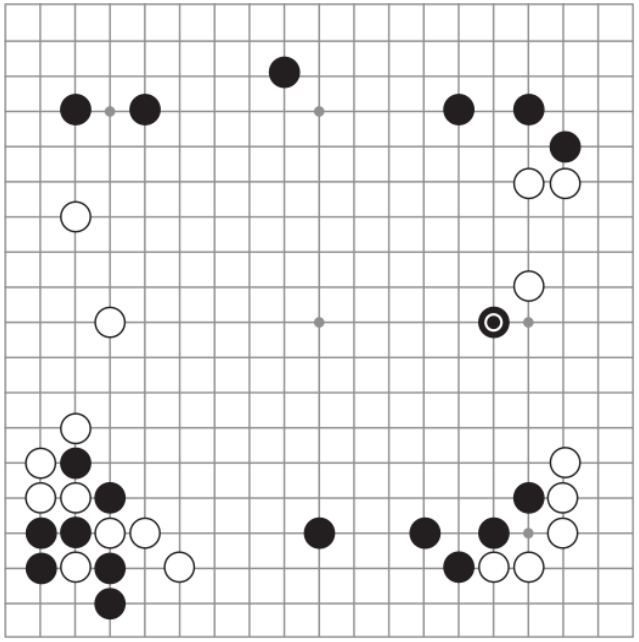
200 Training Episodes

400 Training Episodes



600 Training Episodes

DeepMind 2016: AlphaGo versus Lee Sedol 4 - 1

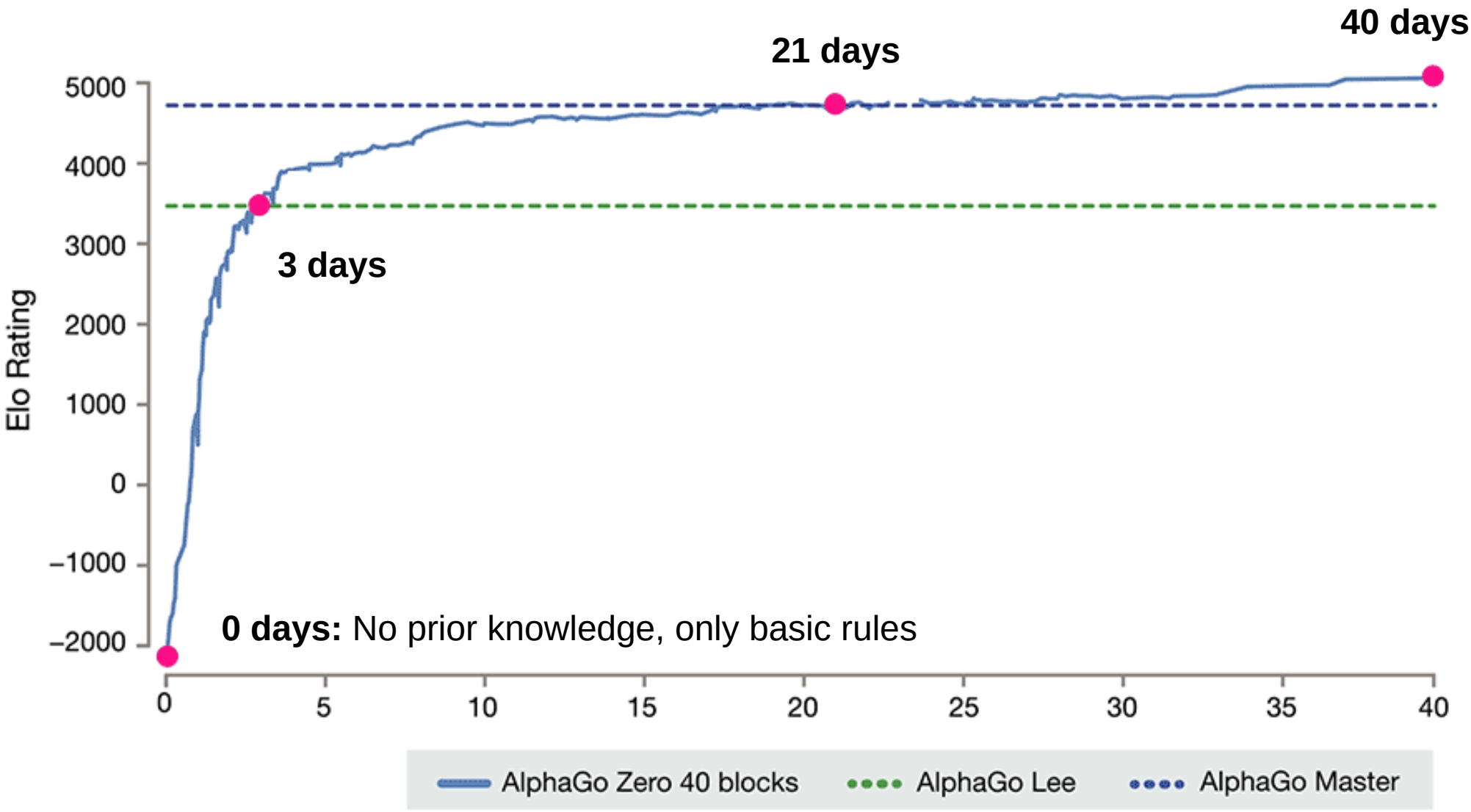


The ‘unexplainable’ move

AlphaGo versus Lee Sedol 2016

Game 2, Turn 37

DeepMind 2017: AlphaGoZero – Learning by self-play only



https://deepmind.com/blog/article/alphago-zero-starting-scratch

DeepMind 2019: AlphaZero & Kramnik

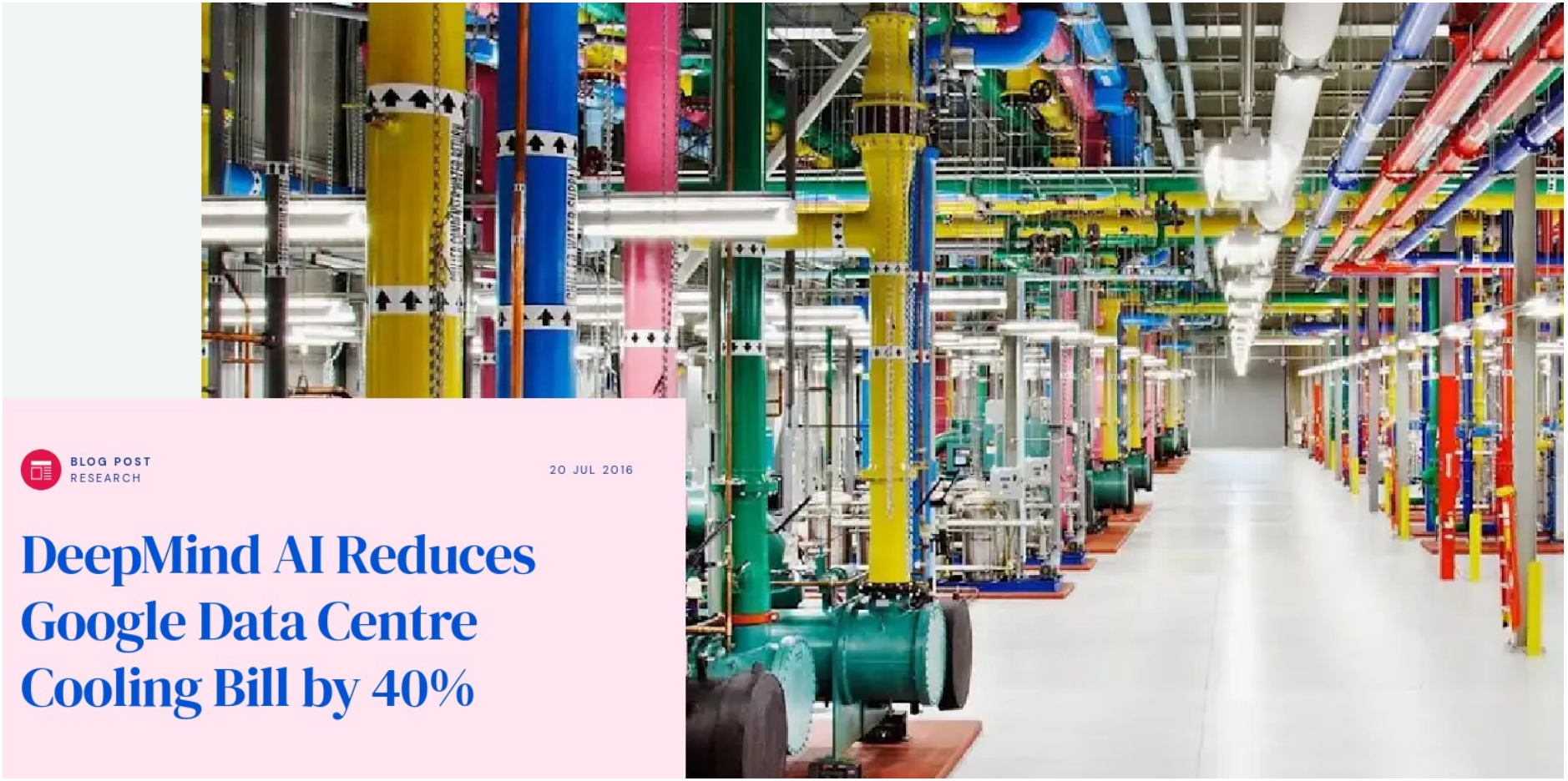
Assessing Game Balance with AlphaZero: Exploring Alternative Rule Sets in Chess

https://arxiv.org/abs/2009.04374 (15 September 2020)

Car racing & *Pit Rho* strategy recommendation system

https://medium.com/analytics-vidhya/sports-analytics-ai-in-sports-5ea645d2e0af

Adaptive datacenter cooling



https://deepmind.com/blog/article/deepmind-ai-reduces-google-data-centre-cooling-bill-40

AI documentaries (free on YouTube)

AlphaGO
The Movie

https://www.alphagomovie.com/

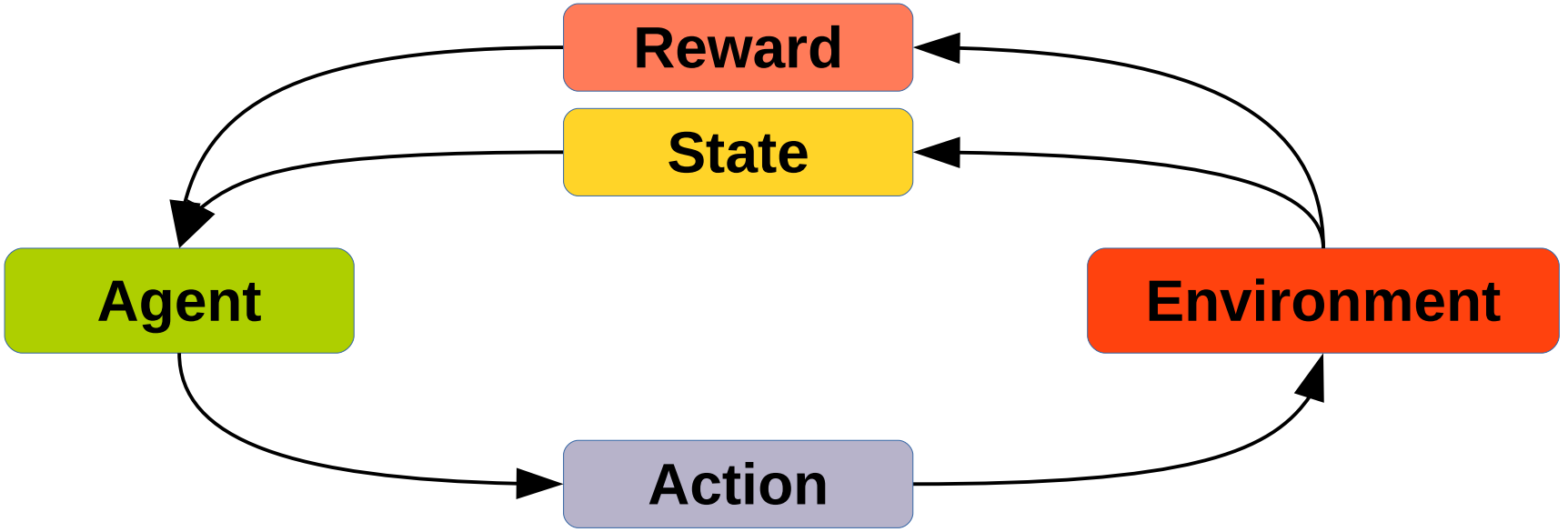
https://www.youtube.com/watch?v=WXuK6gekU1Y

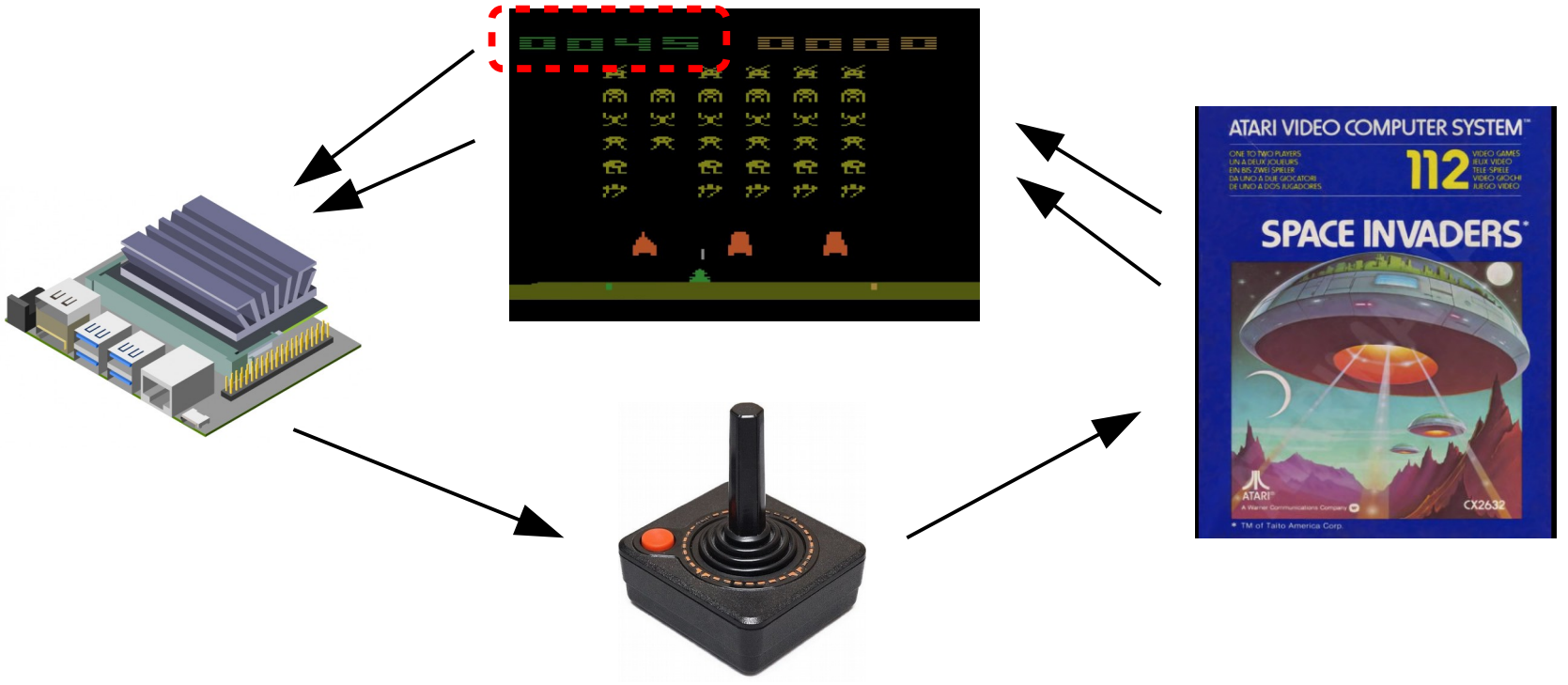
The Age of A.I.

https://www.youtube.com/playlist?list=PLjq6DwYksrzz_fsWIpPcf6V7p2RNAneKc

Reinforcement Learning

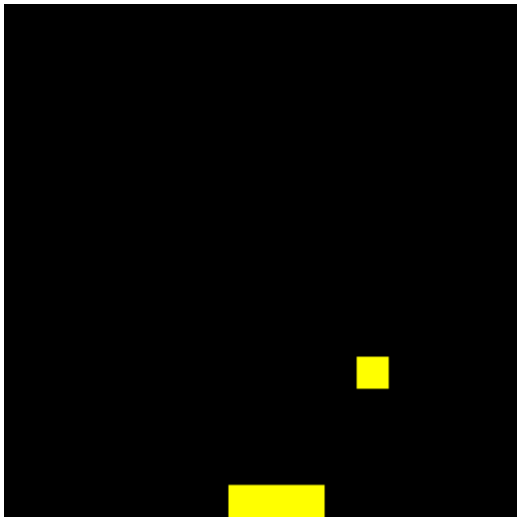
The *standard model* of Reinforcement Learning

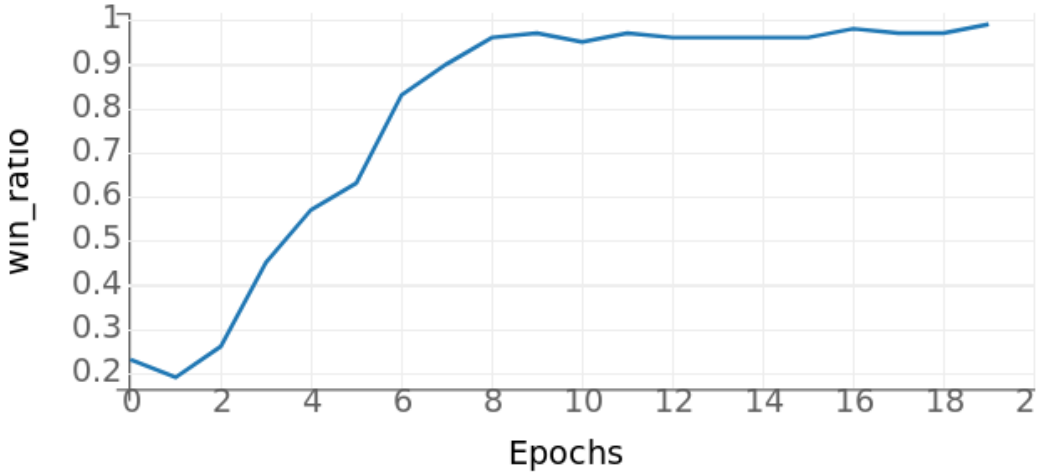




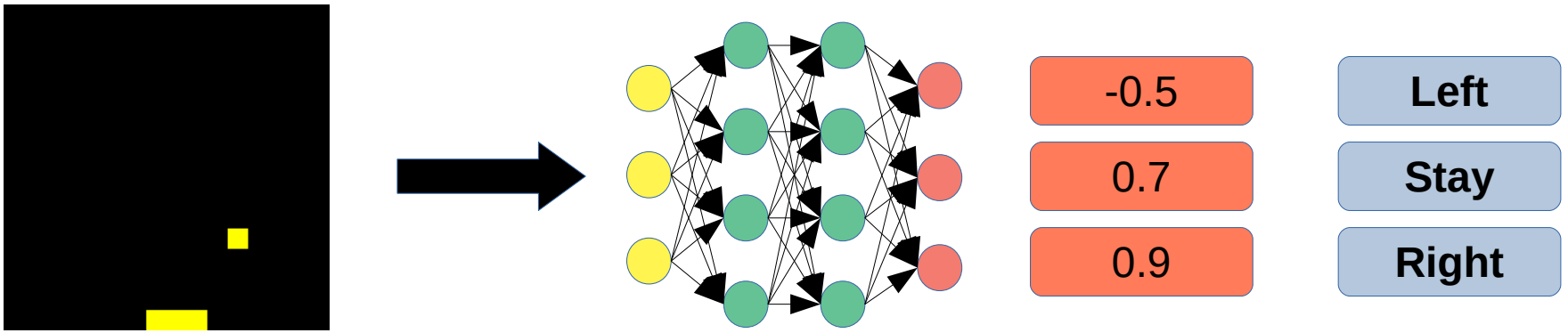
Live Demo

Catch game in 5 minutes





Q-Learning & Value networks



State

Values

Actions

expected total future reward for state/action

$$Q(S,A) \rightarrow V$$

Q-Function:

$$Q(S) \rightarrow \vec{V}_a$$

Training a value network (Q-Learning)

State | Action | State+1 | Reward | Game over

Game over:

$$Q(S,A) \leftarrow R_{final}$$

Not Game over:

$$Q(S,A) \leftarrow R + \underbrace{R_{+1} + R_{+2} + R_{+3} \ldots R_{final}}$$

$$Q(S,A) \leftarrow \underbrace{R}_{\text{Immediate Reward}} + \underbrace{Q(S_{+1})_{max}}_{\text{Future Expectations}}$$

Epsilon-Greedy / Future discount / Experience Replay

Epsilon:

“Exploitation” vs “Exploration”

Value -> Action

1-ε	**$A(V_{max})$**
ε	**A(rnd)**

Gamma:

Future discount factor

Adjust short/long term

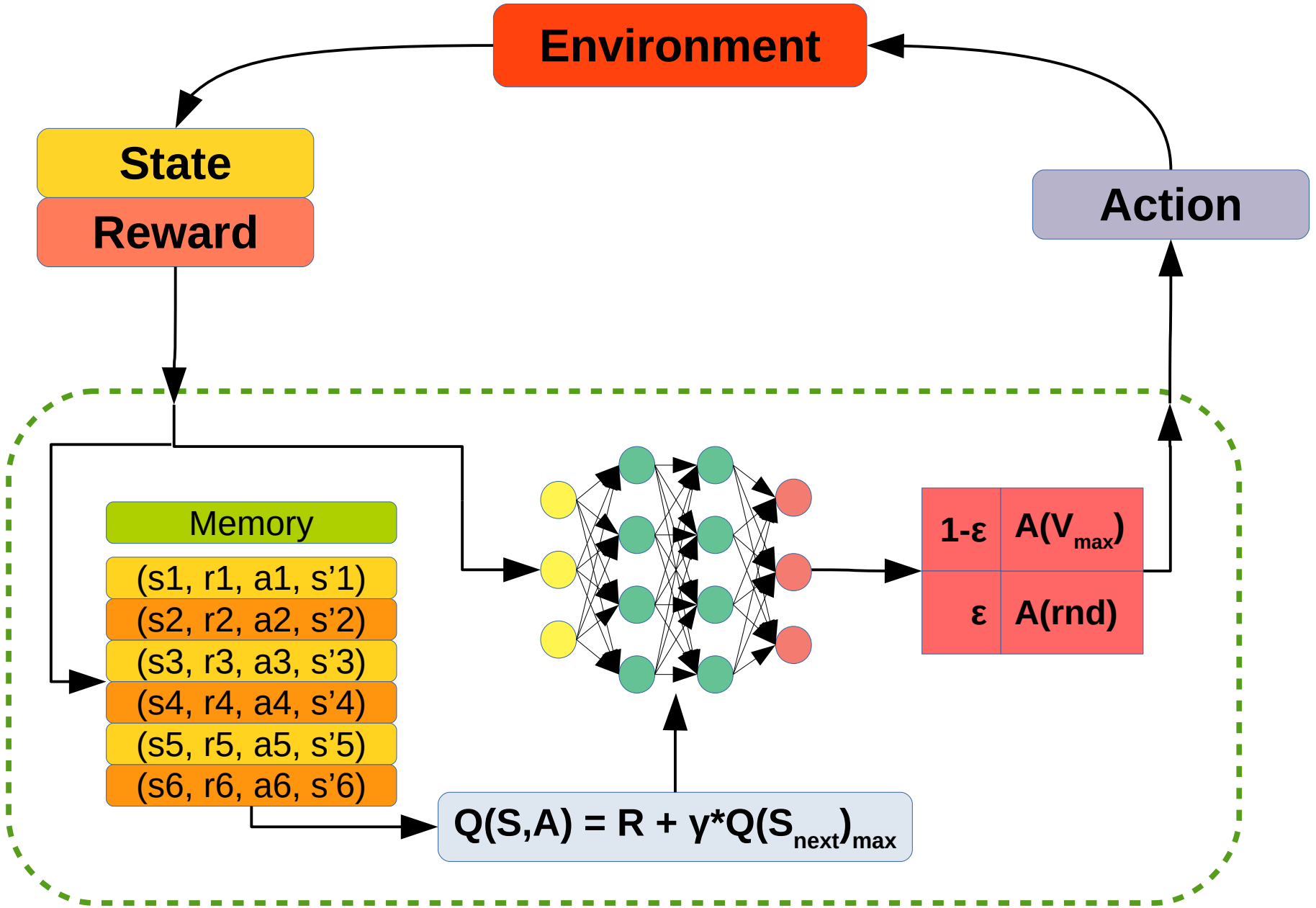
$$Q(S,A) \leftarrow R + \gamma * Q(S_{next})_{max}$$

Experience replay

Random batch training

Prevent ‘catastrophic fogetting’

Memory
$(s_1, r_1, a_1, s'_1, o_1)$
$(s_2, r_2, a_2, s'_2, o_2)$
$(s_3, r_3, a_3, s'_3, o_3)$
$(s_4, r_4, a_4, s'_4, o_4)$
$(s_5, r_5, a_5, s'_5, o_5)$
$(s_6, r_6, a_6, s'_6, o_6)$

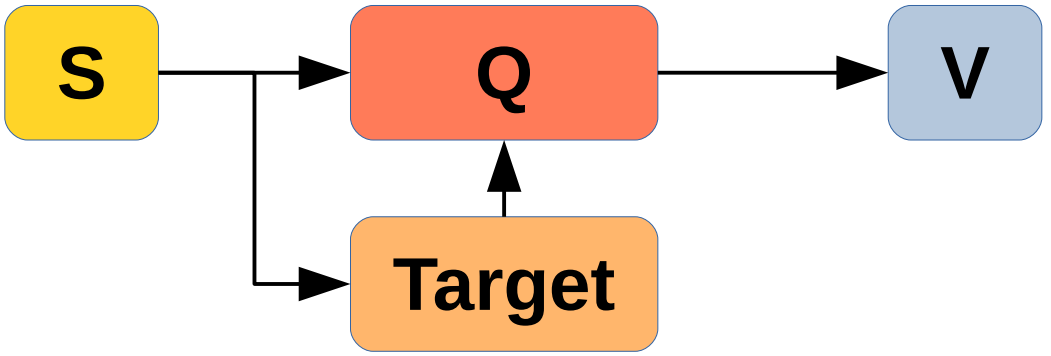

Environment
State
Reward
Action
Memory
(s1, r1, a1, s'1)
(s2, r2, a2, s'2)
(s3, r3, a3, s'3)
(s4, r4, a4, s'4)
(s5, r5, a5, s'5)
(s6, r6, a6, s'6)
1-ε
A(V max)
ε
A(rnd)
Q(S,A) = R + γ*Q(S next)max

Other RI architectures

Target network

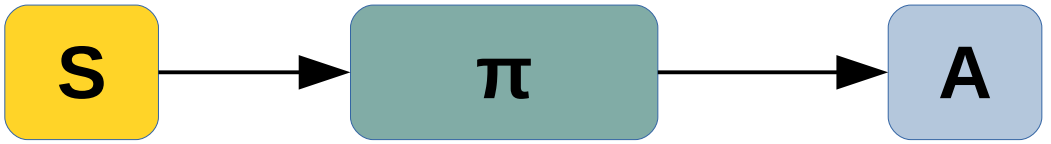
More stability

Action V – Train V



Policy network

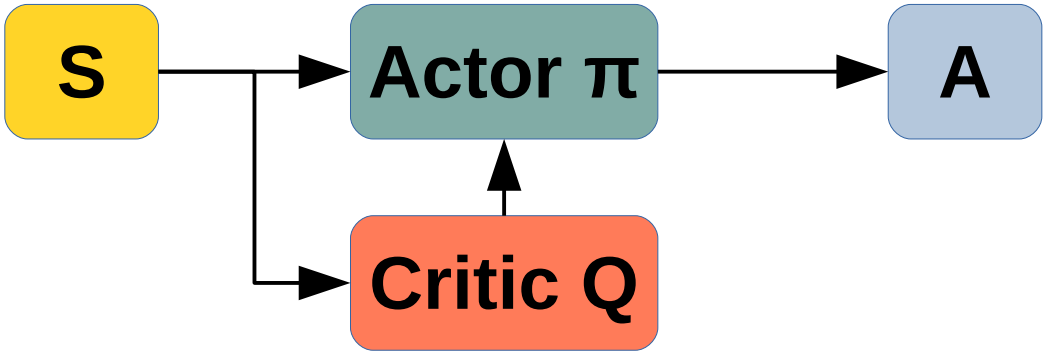
Directly learn action distribution



Actor-Critic

Even more stability

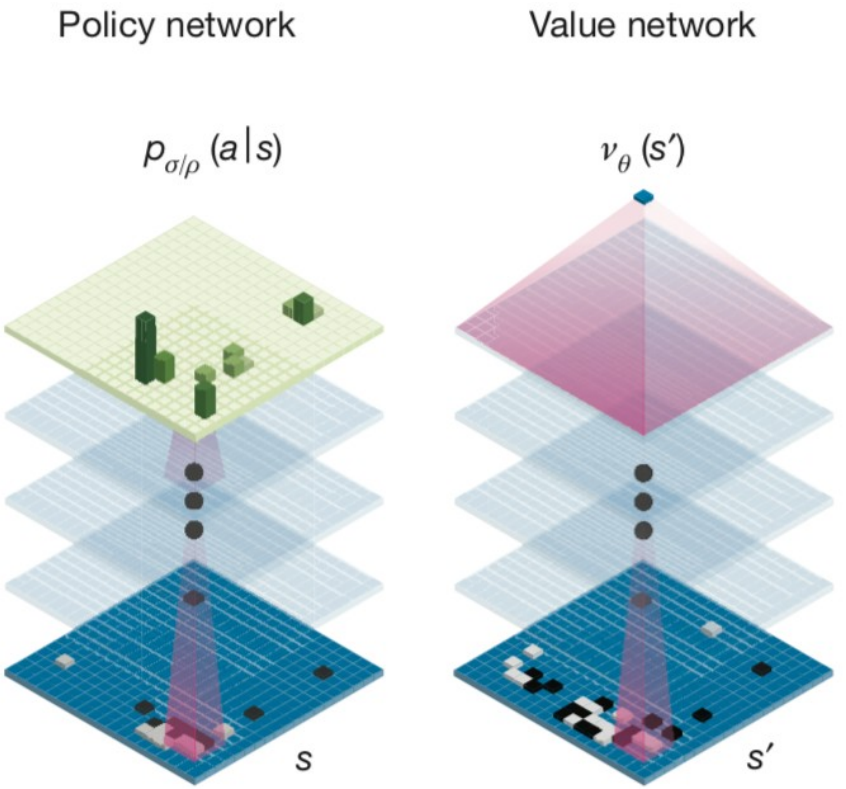
Address *'bias / variance problem'*



Actor-Critic example: AlphaGo

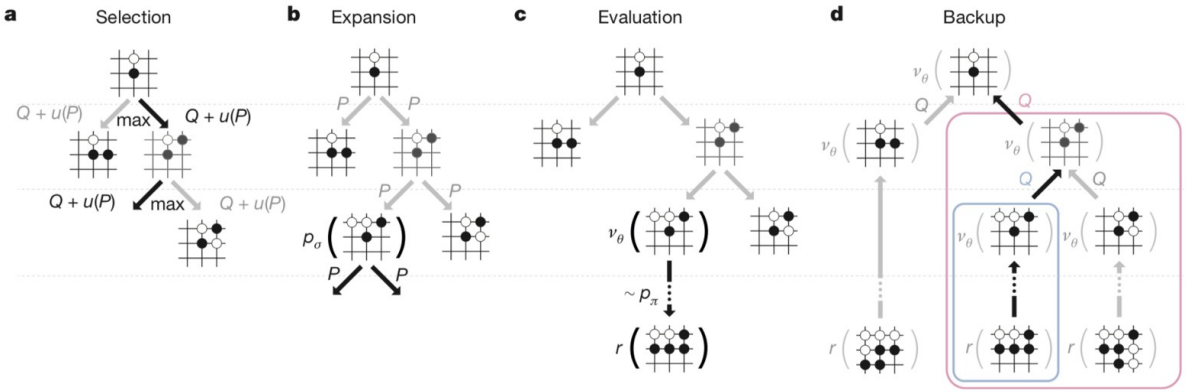
Actor-Critic:

Position evaluation



Monte Carlo Tree Search:

Look-ahead search

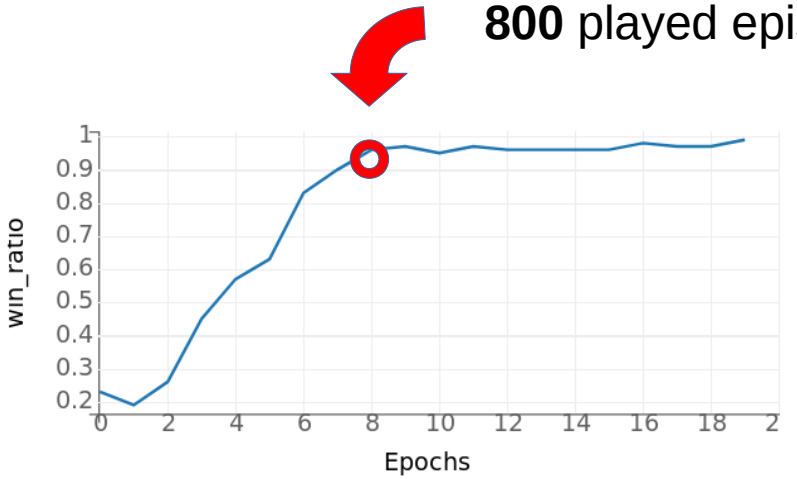


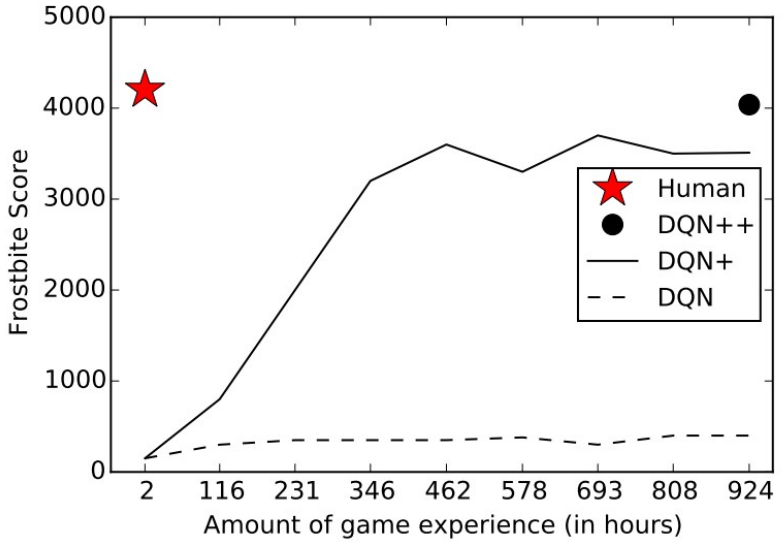
Limitations of current AI

Current AI is limited:

- **Data-Hungry** *Sample-inefficient*, needs massive amounts of data
- **Opaque** An AI's decisions are very *difficult to explain*
- **Narrow** *"One-trick Pony"*, able to learn only one very narrow task
- **Brittle** *Breaking adversarial examples* are easy to find

Sample inefficiency



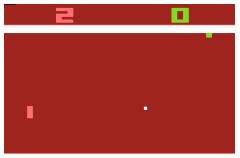


Building Machines That Learn and Think Like People (2016) https://arxiv.org/abs/1604.00289

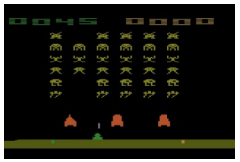
DQN vs Human performance

(Source: “Human-level control through deep reinforcement learning (2015)”)

Breakout 1327%



Pong 132%



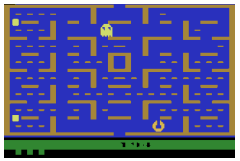
Space Invaders 121%

Expert Player 100%

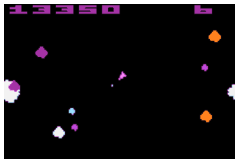
Q*Bert 78%

Average Player 75%

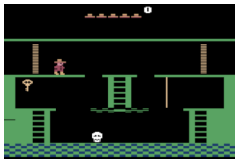
Centipede 62%



Pac-Man 13%



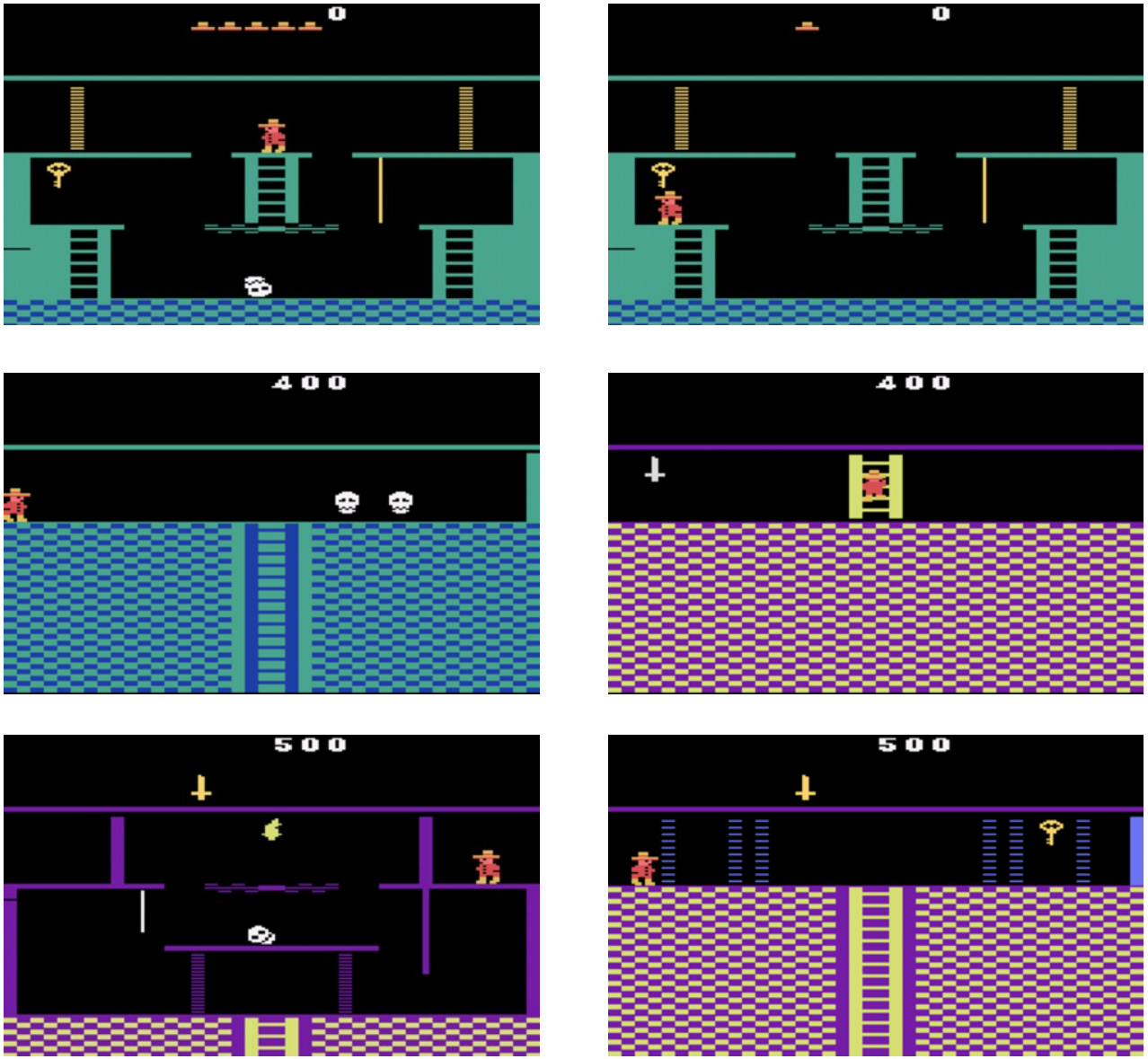
Asteroids 7%



Montezumas Revenge 0%

Random Play 0%

“Montezuma’s Revenge” - RL Holy Grail



“Naughty” AIs

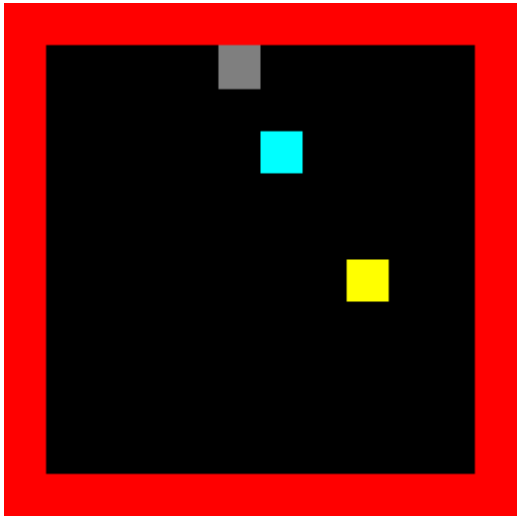
RI systems optimize their reward function – at any cost.

Suboptimal chosen rewards can lead to unexpected results.

Fruit game:

Hungry mouse must find block of cheese,
and avoid poison and death by starving.

Problem: Agent becomes suicidal when negative reward is set too high -> *“Maximizes” reward by killing itself as fast as possible.*



More examples at:

https://hackaday.com/2018/11/11/the-naughty-ais-that-gamed-the-system/