Real-life reinforcement learning

Julien Simon
Global Evangelist, AI & Machine Learning, AWS
@julsimon

Types of Machine Learning

Supervised learning

- Run an algorithm on a labeled data set.
- The model learns how to correctly predict the right answer.
- Regression and classification are examples of supervised learning.

Unsupervised learning

- Run an algorithm on an unlabeled data set.
- The model learns patterns and organizes samples accordingly.
- Clustering and topic modeling are examples of unsupervised learning.

Building a dataset is not always an option

- Large, complex problems
- Uncertain, dynamic environments
- Continuous learning

- Supply chain management
- HVAC systems
- Industrial robotics
- Autonomous vehicles
- Portfolio management
- Oil exploration
- etc.

aws

Building a dataset is not always an option

Large, complex problems

Uncertain, dynamic environments

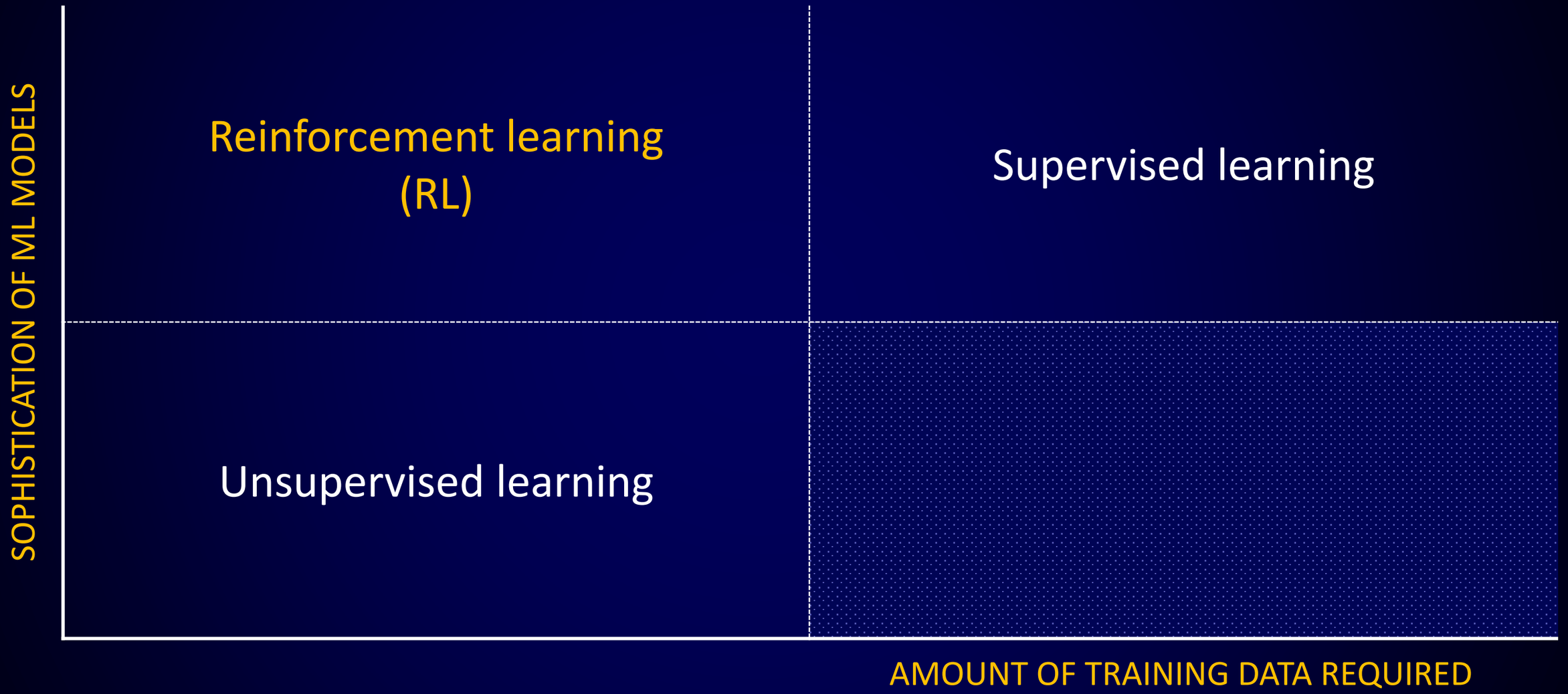
Continuous learning

Supply chain management, HVAC systems, industrial robotics, autonomous vehicles, portfolio management, oil exploration, etc.

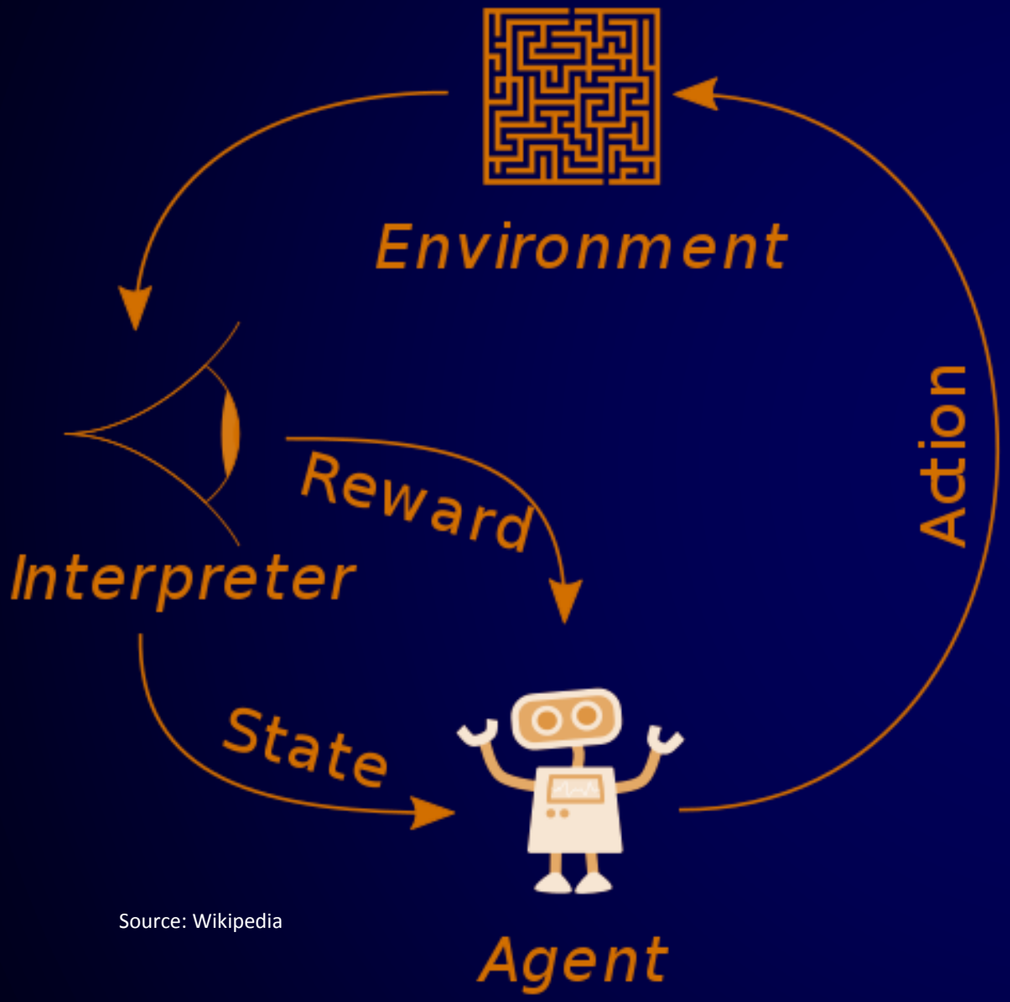
aws

Learning without any data: we've all done it!

Types of Machine Learning



Reinforcement Learning



Source: Wikipedia

An agent interacts with its environment.

The agent receives a positive or negative reward for actions that it takes: rewards are computed by a user-defined function which outputs a numeric representation of the actions that should be incentivized.

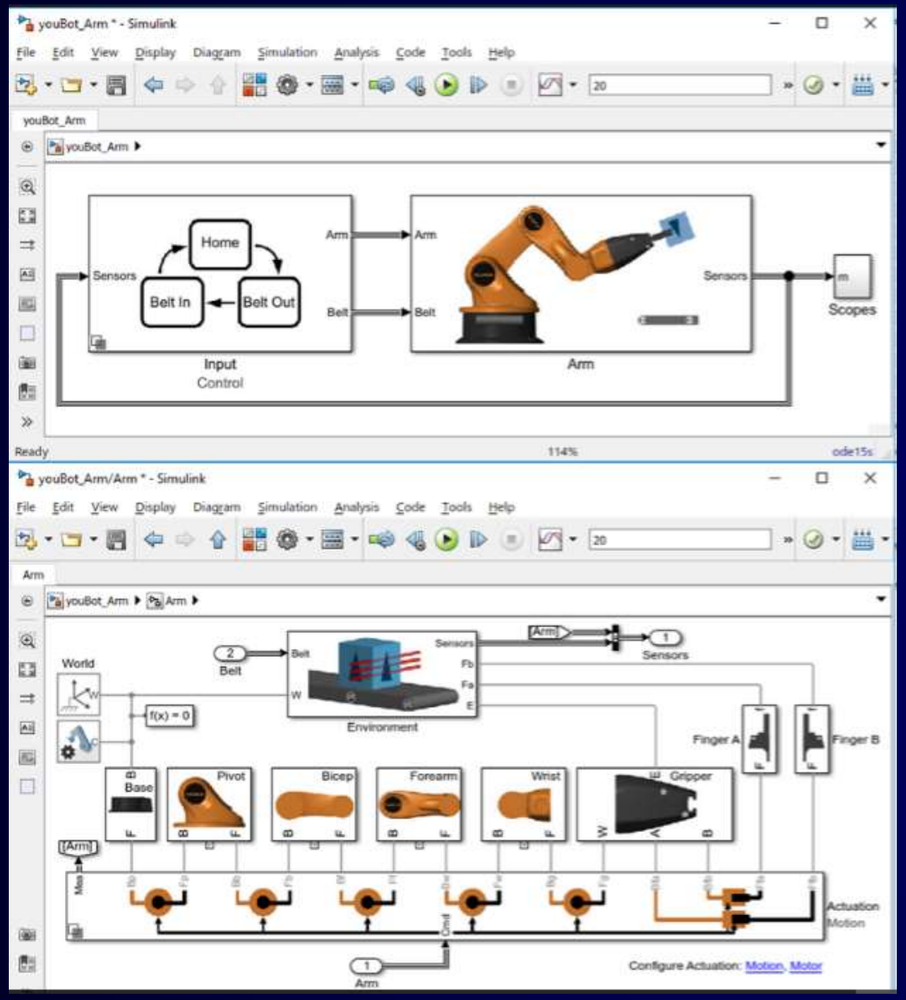
By trying to maximize the accumulation of rewards, the agent learns an optimal strategy (aka policy) for decision making.

aws

Training a RL model

1. Formulate the problem: goal, environment, state, actions, reward
2. Define the environment: real-world or simulator?
3. Define the presets
4. Write the training code and the reward function
5. Train the model

Environments can be sophisticated



Popular simulators:

OpenAI,
Roboschool,
EnergyPlus,
MATLAB,
Simulink

You can write your own RL environment

```
import gym
from gym.spaces import Discrete, Box

def __init__():
    #Initialize the RL environment

def reset(self):
    #Reset the RL environment

def step(self, action):
    #Take an action in the RL environment
    #Return the observation
    #return reward (positive or negative)

def render():
    #Render the RL environment
```


aws

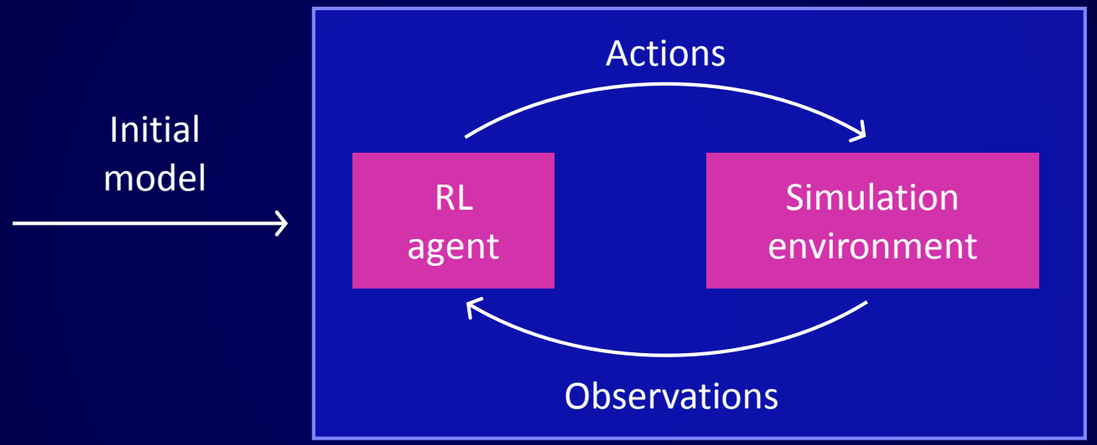
Learning to walk

https://github.com/awslabs/amazon-sagemaker-examples/tree/master/reinforcement_learning/rl_roboschool_ray

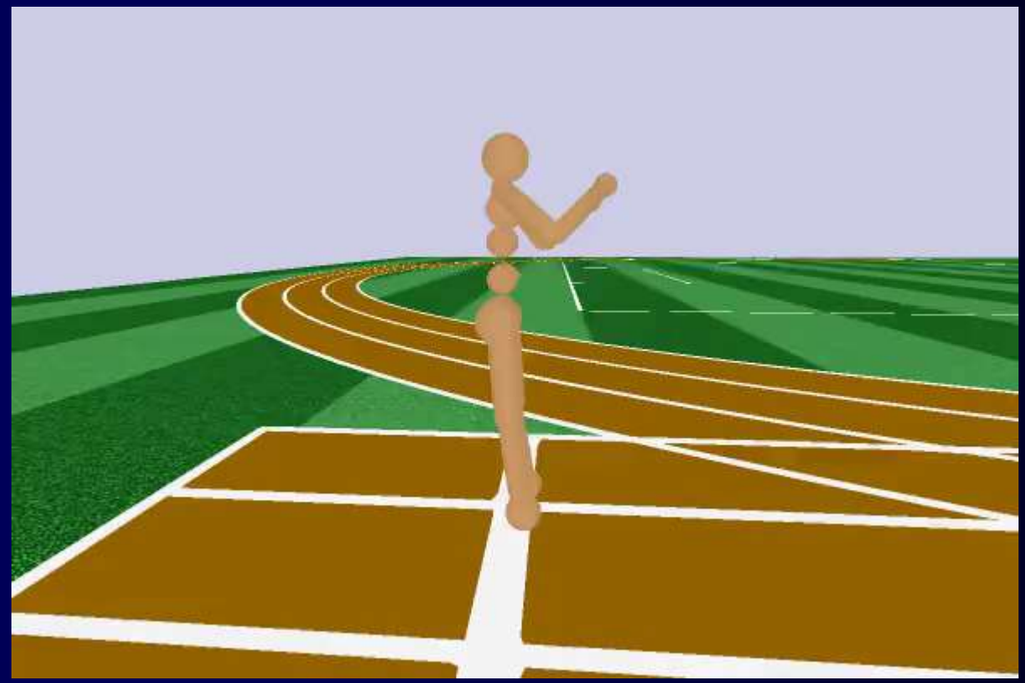
The players

At first, the agent can’t even stand up

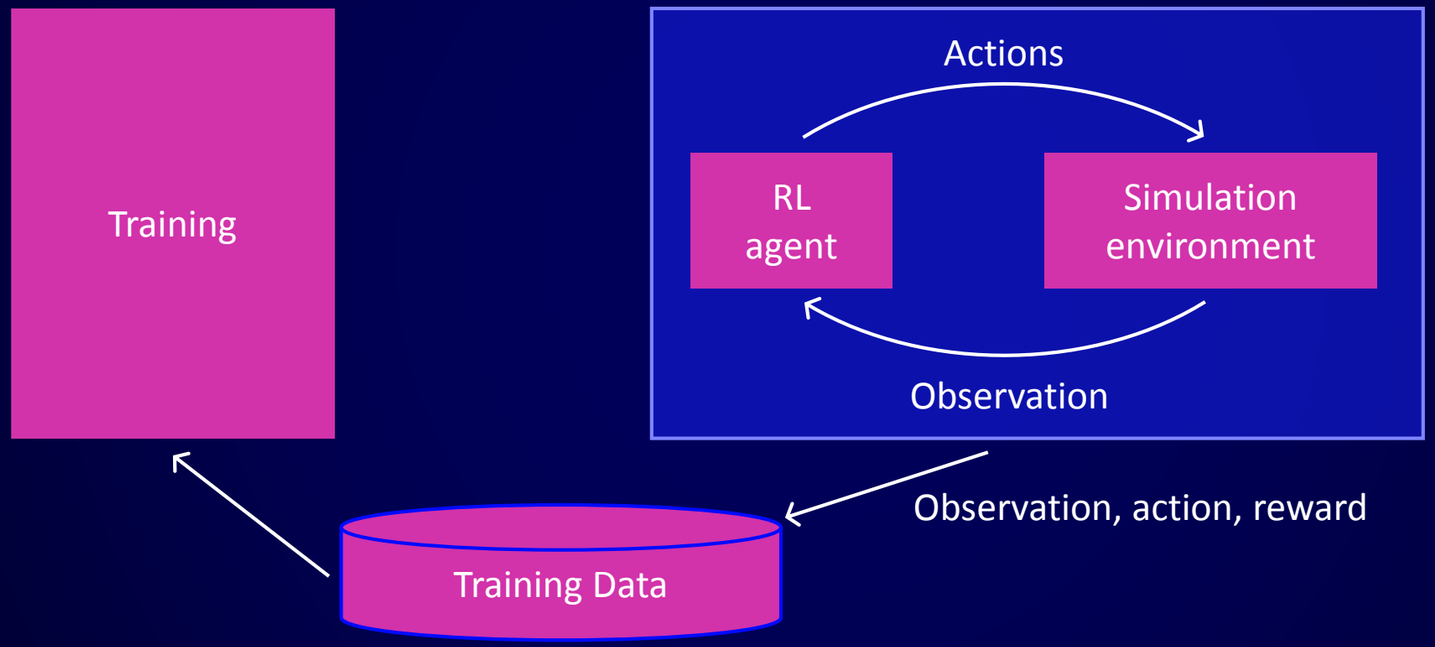
Actions and observations



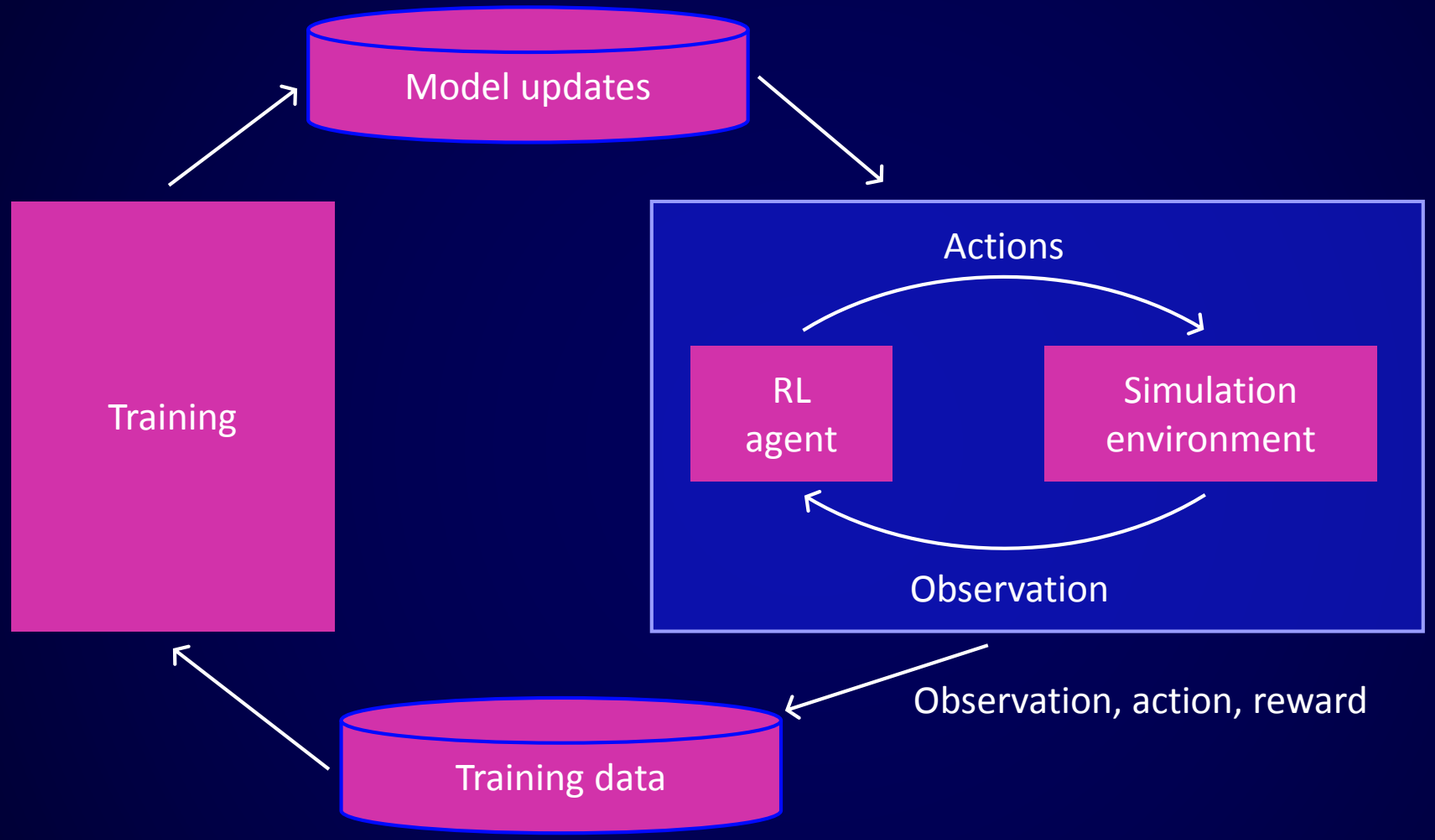
The model learns through actions and observations



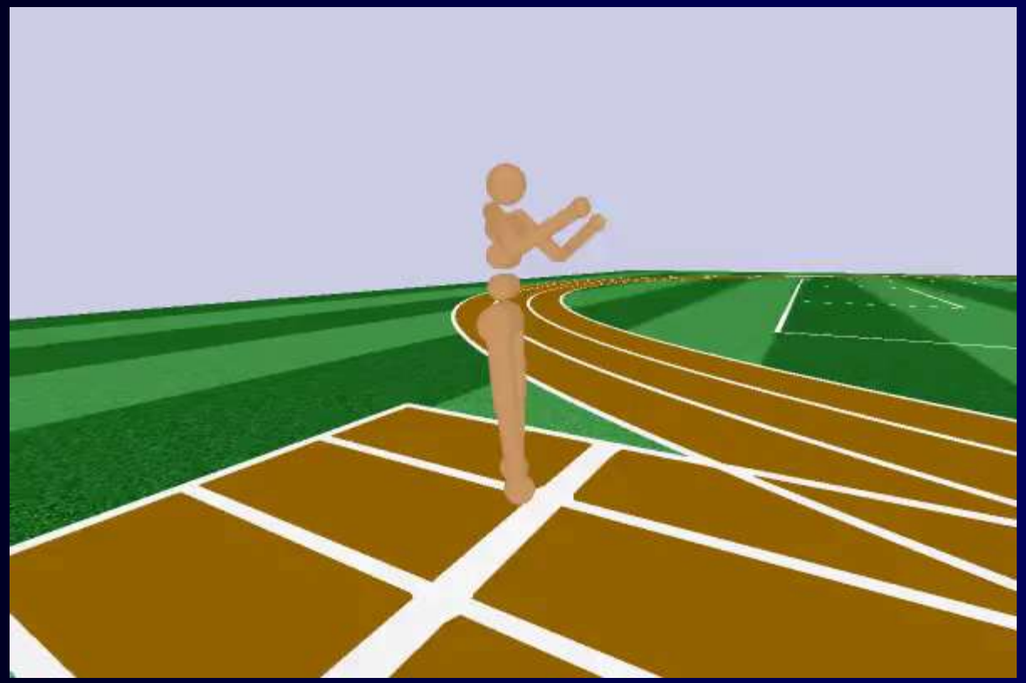
Interactions generate training data



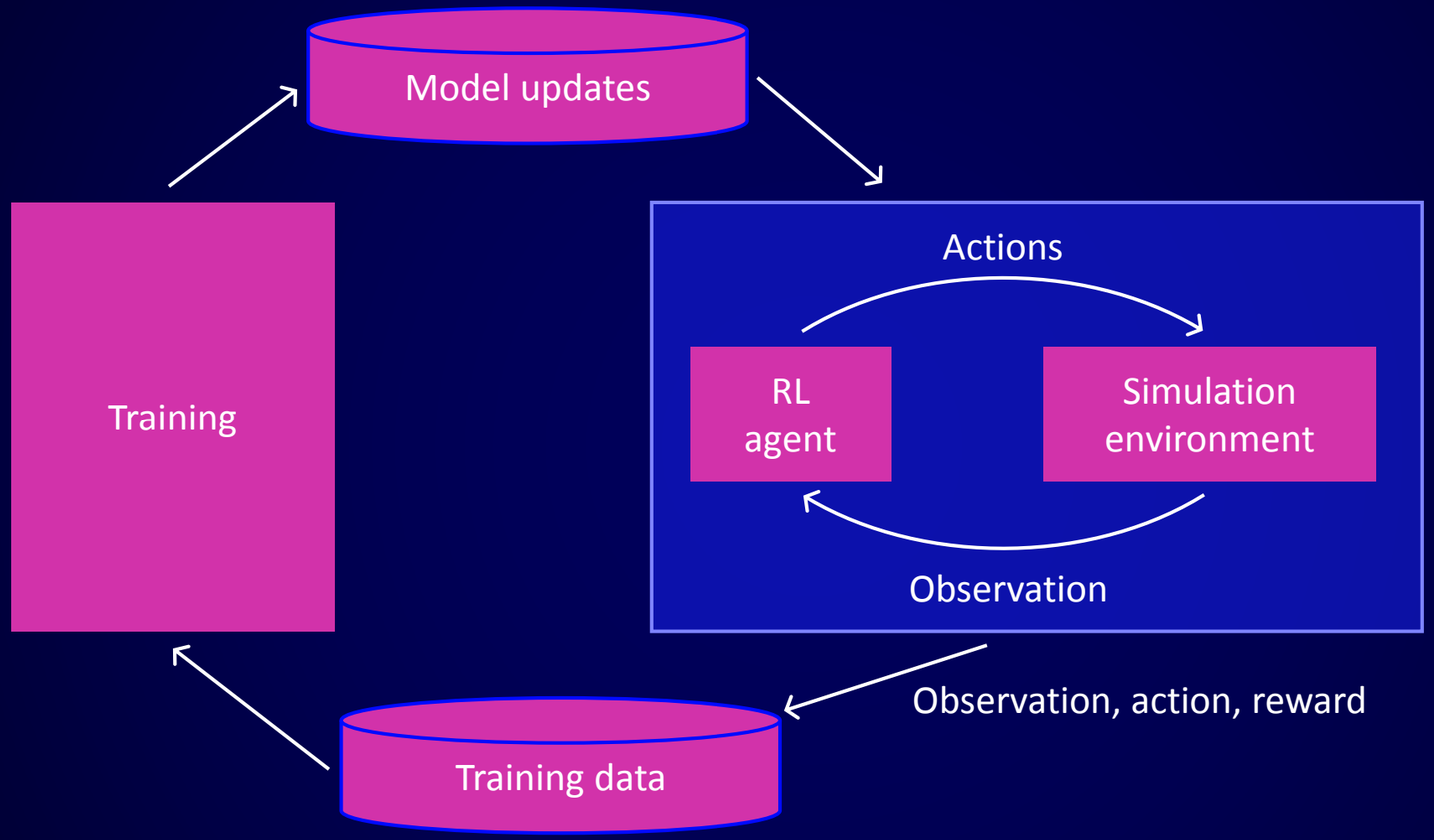
Training results in model updates



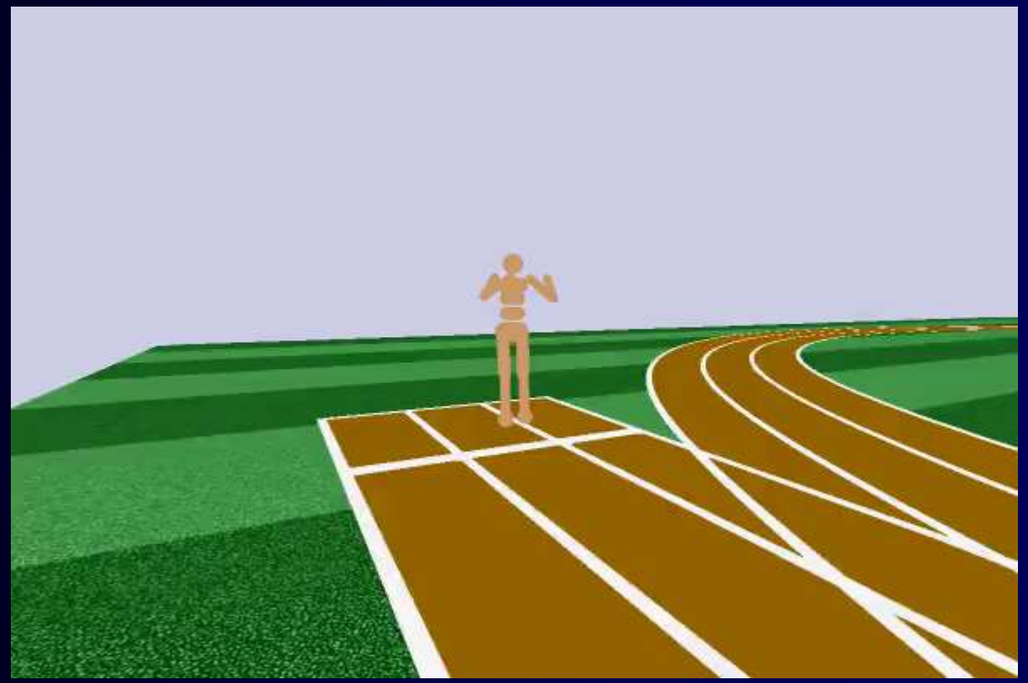
The agent learns to stand and step



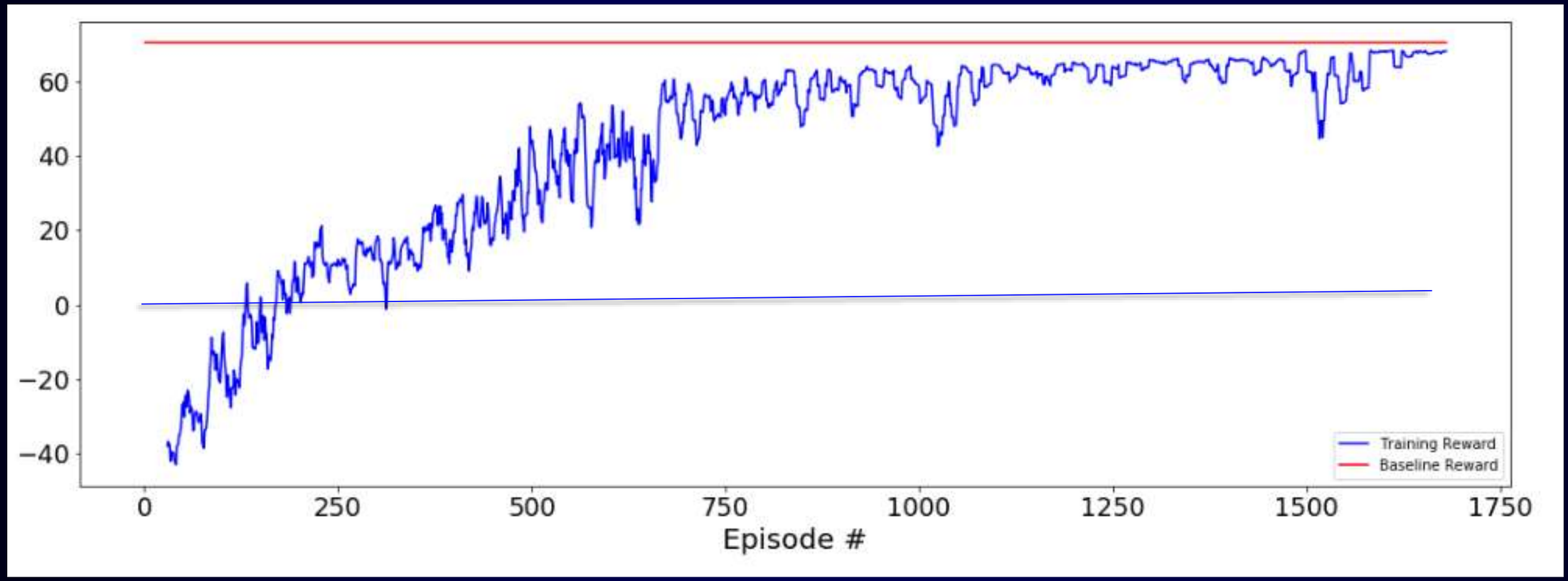
Multiple training episodes improve learning



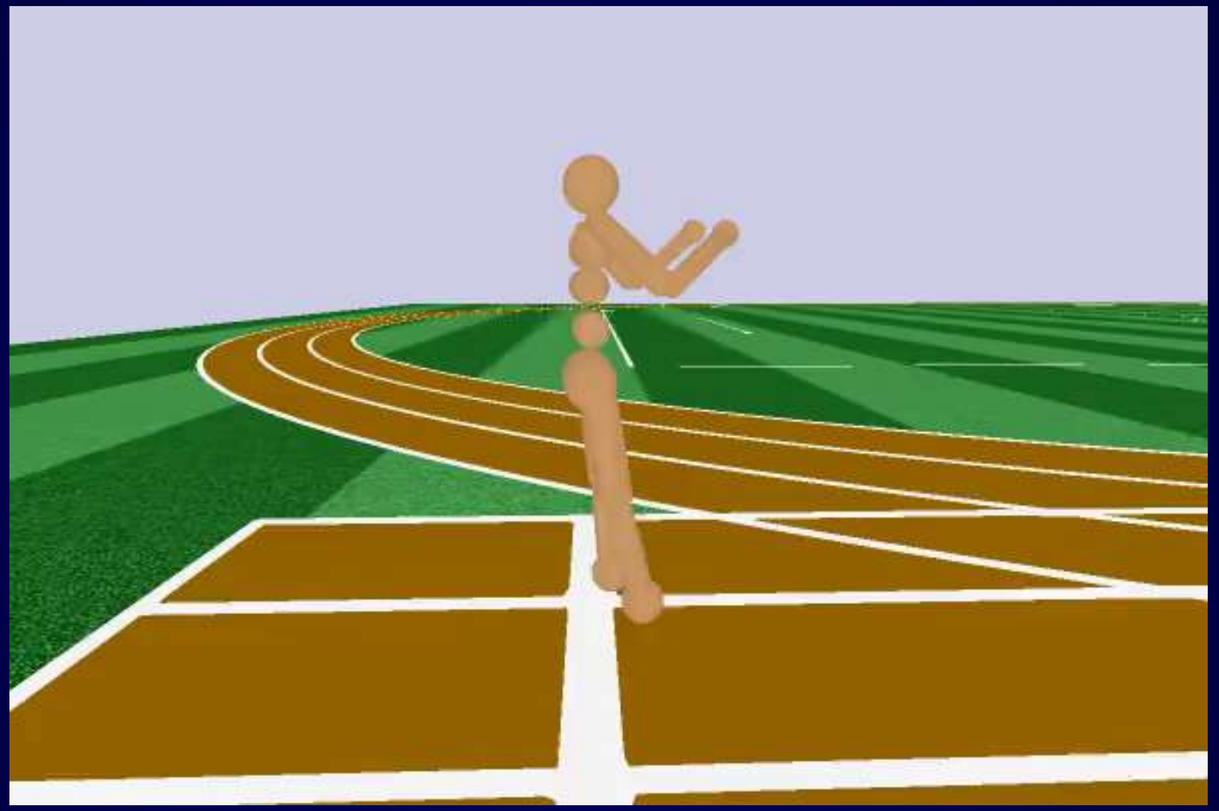
Making progress



RL agents try to maximize rewards

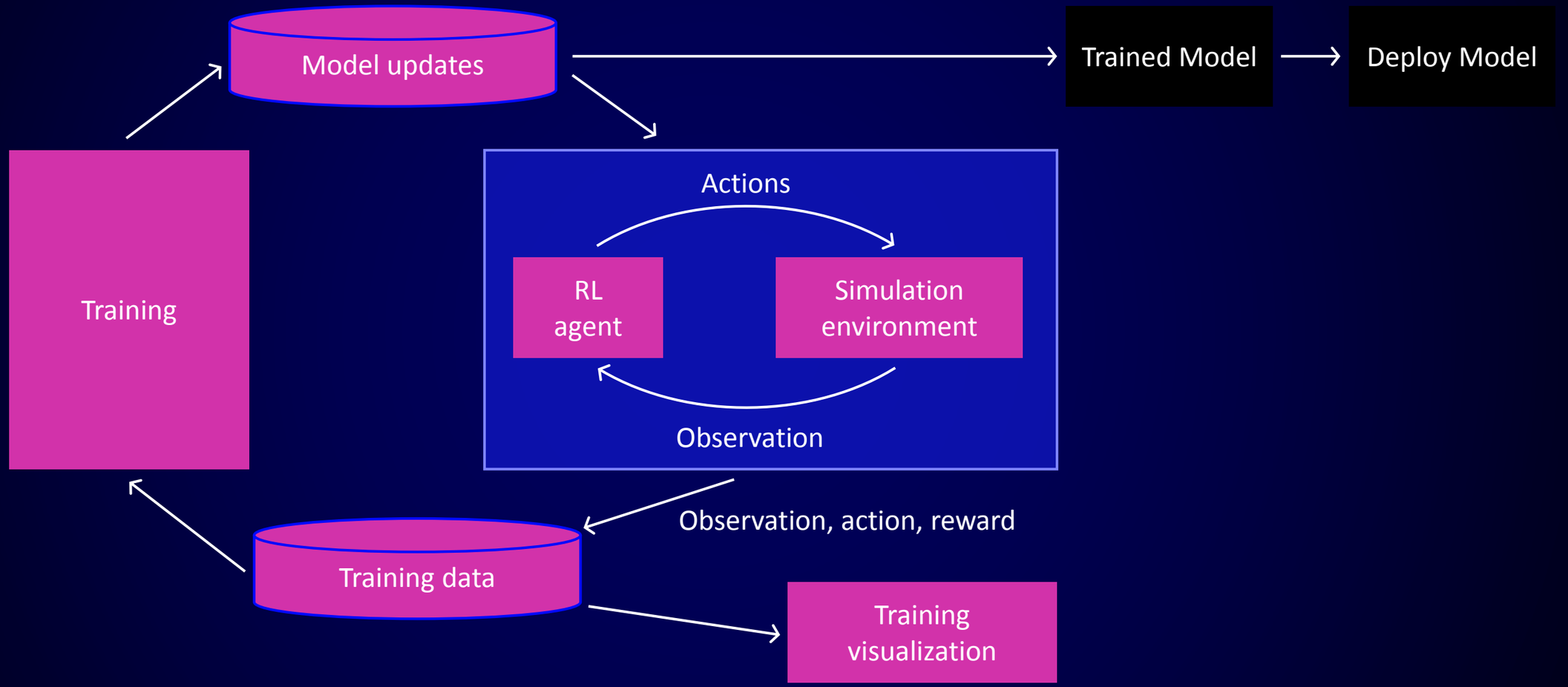


Eventually, the model learns how to walk and run



You can continue training Harry to jump obstacles, play games, dance, and more

Evaluate and deploy trained models



Customers are using RL on AWS

GE Healthcare

HONDA

SIXT

mixi

amazon

SyntheticGestalt

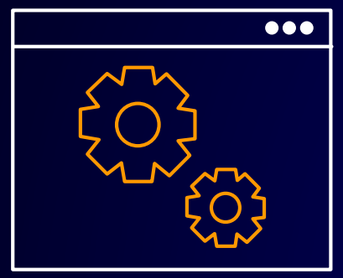
Scientific Research by Artificially Intelligent Agents

Tradelegs

aws

Amazon SageMaker RL

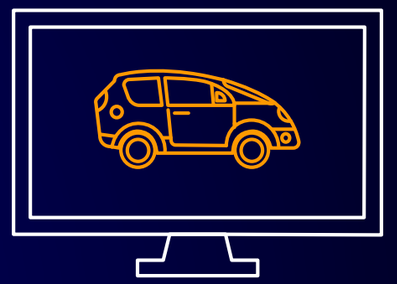
Reinforcement learning for every developer and data scientist



Fully managed

mxnet
TensorFlow
intel riselab UC Berkeley

Broad support for frameworks



Broad support for simulation environments including SimuLink and MatLab

---

KEY FEATURES

TensorFlow, Apache MXNet, Intel Coach, and Ray RL support

2D & 3D physics environments and OpenAI Gym support

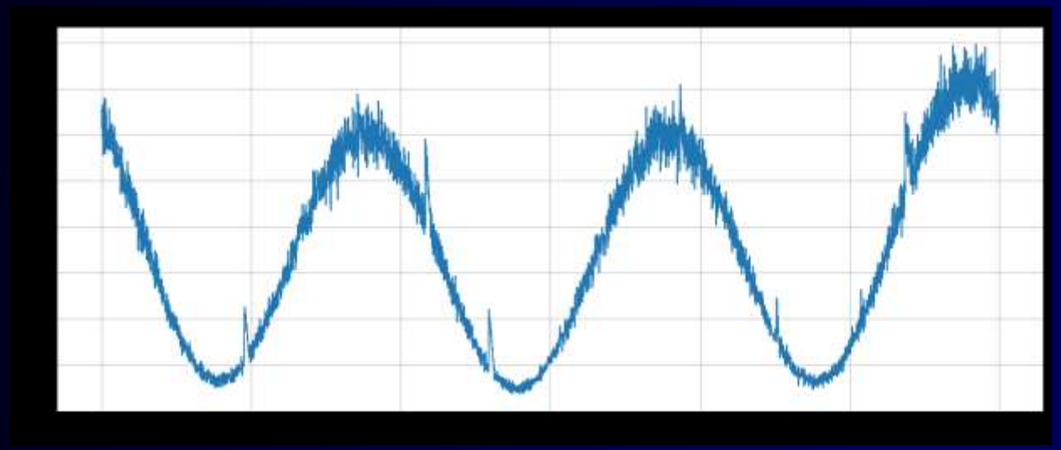
Supports Amazon Sumerian and Amazon RoboMaker

Example notebooks and tutorials

aws

Robotics

Auto Scaling



Objective	Adapt instance capacity to load profile
STATE	Current load, failed jobs, active machines
ACTION	Remove or add machines
REWARD	Positive for successful transactions High penalty for losing transactions

https://github.com/awslabs/amazon-sagemaker-examples/tree/master/reinforcement_learning/rl_predictive_autoscaling_coach_customEnv

aws

Financial portfolio management

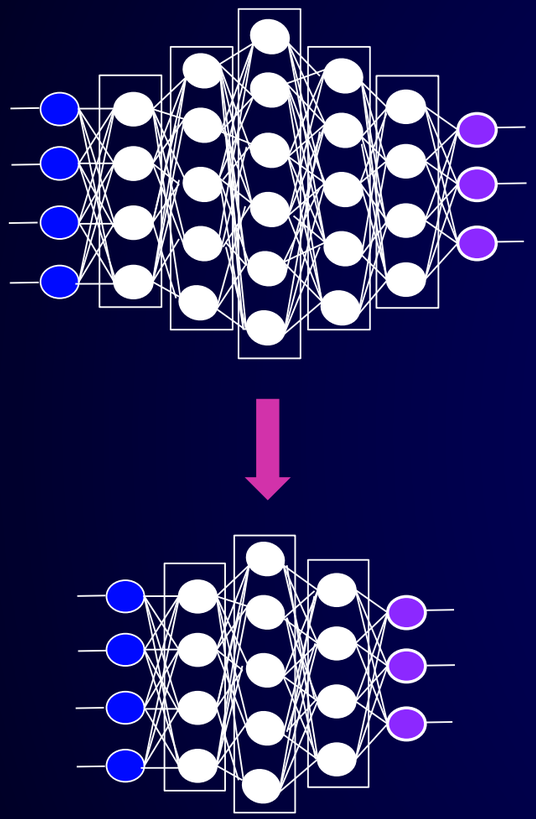
Objective	**Maximize the value of a financial portfolio**
STATE	Current stock portfolio, price history
ACTION	Buy, sell stocks
REWARD	Positive when return is positive Negative when return is negative

Jiang, Zhengyao, Dixing Xu, and Jinjun Liang
« A deep reinforcement learning framework for the financial portfolio management problem. »
arXiv:1706.10059 (2017)

https://github.com/awslabs/amazon-sagemaker-examples/tree/master/reinforcement_learning/rl_portfolio_management_coach_customEnv

aws

Compressing deep learning models



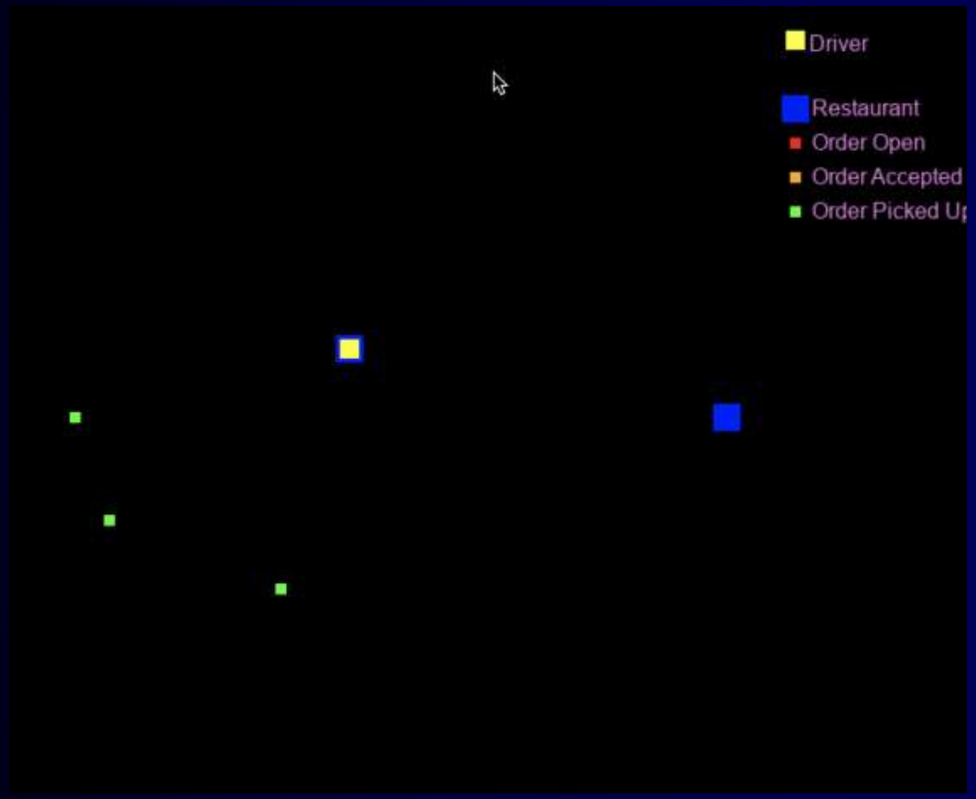
Objective	Compress model without losing accuracy
STATE	Layers
ACTION	Remove or shrink a layer
REWARD	A combination of compression ratio and accuracy.

Ashok, Anubhav, Nicholas Rhinehart, Fares Beainy, and Kris M. Kitani "N2N learning: network to network compression via policy gradient reinforcement learning." arXiv:1709.06030 (2017).

https://github.com/awslabs/amazon-sagemaker-examples/tree/master/reinforcement_learning/rl_network_compression_ray_custom

aws

Vehicle routing



Objective	Fulfill customer orders
STATE	Current location, distance from homes
ACTION	Accept, pick up, and deliver order
REWARD	Positive when we deliver on time Negative when we fail to deliver on time

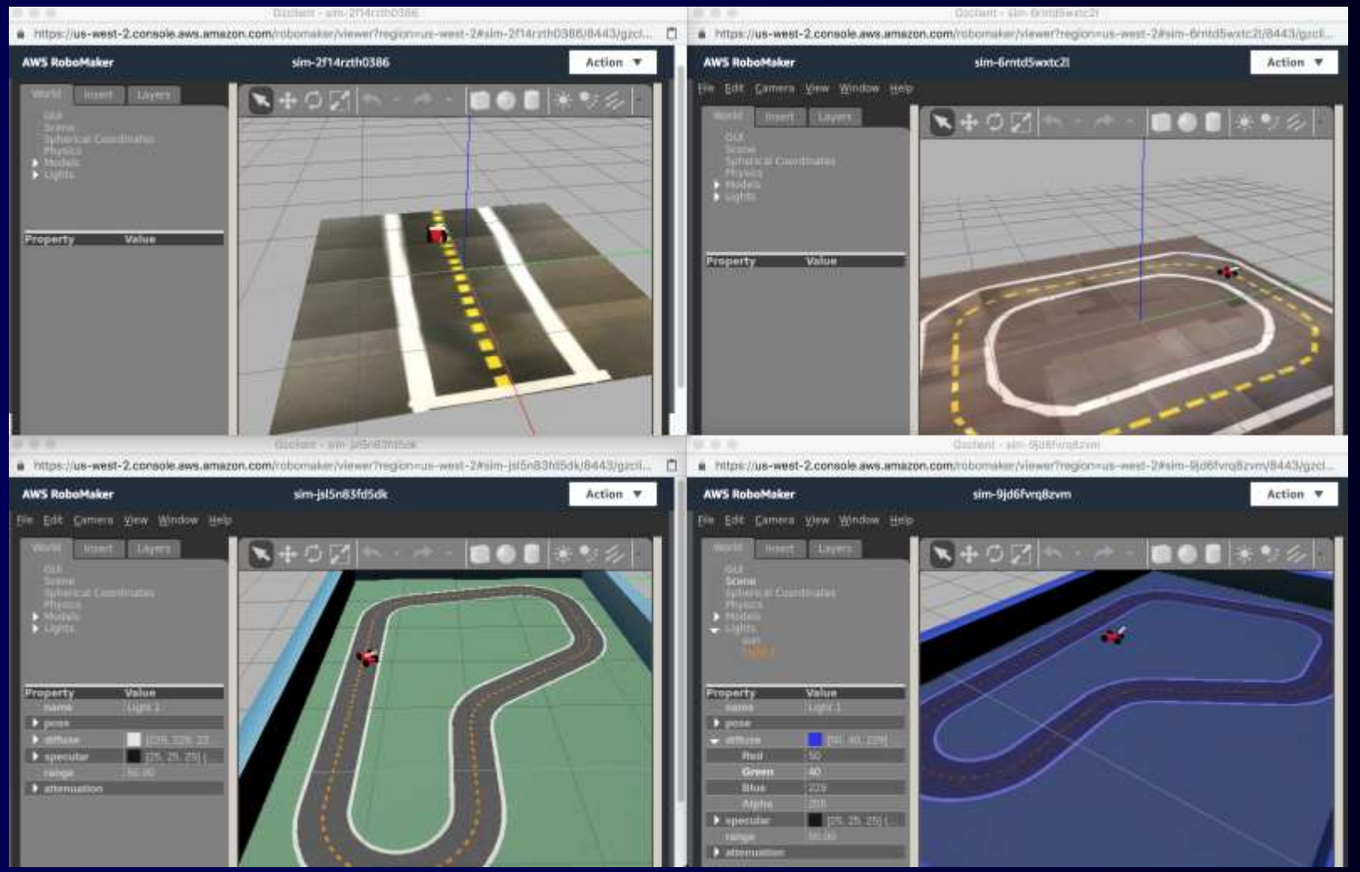
https://github.com/awslabs/amazon-sagemaker-examples/tree/master/reinforcement_learning/rl_traveling_salesman_vehicle_routing_coach

aws

Autonomous driving

AWS DeepRacer

1/18th scale autonomous vehicle



Amazon RoboMaker

Getting started

http://aws.amazon.com/free

https://ml.aws

https://aws.amazon.com/sagemaker
https://github.com/awslabs/amazon-sagemaker-examples
https://aws.amazon.com/blogs/aws/amazon-sagemaker-rl-managed-reinforcement-learning-with-amazon-sagemaker/

https://aws.amazon.com/deepracer/

https://medium.com/@julsimon

Thank you!

Julien Simon
Global Evangelist, AI & Machine Learning, AWS
@julsimon