Agent-based modeling workshop

Matias Quintana

2022-03-24

buds lab
building and urban data science

Disclaimer

- All content here is my personal understanding.
- Main intention is to open up the floor for discussion and trade concepts and ideas.
- Some code here is still under development so be patient and donate some coffee if you can. Nespresso red pods are suggested but Kopi O Kosong Peng does the job (UTown >>> Techno Edge).

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Agent-based modeling (ABM)

- Agent = Computational mechanism that exhibits a high degree of **autonomy**, performing **actions** in its environment based on **information** (sensors, feedback) received from the **environment**.

Panait, L., & Luke, S. (2005). Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, *11*(3), 387–434. https://doi.org/10.1007/s10458-005-2631-2
https://www.quora.com/What-is-the-relationship-between-agent-based-modeling-and-reinforcement-learning

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Agent-based modeling (ABM)

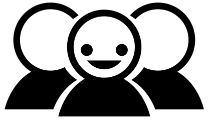
- Agent = Computational mechanism that exhibits a high degree of **autonomy**, performing **actions** in its environment based on **information** (sensors, feedback) received from the **environment**.
    - Obvious “agent” for us?

Panait, L., & Luke, S. (2005). Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, *11*(3), 387–434. https://doi.org/10.1007/s10458-005-2631-2
https://www.quora.com/What-is-the-relationship-between-agent-based-modeling-and-reinforcement-learning

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Agent-based modeling (ABM)

- Agent = Computational mechanism that exhibits a high degree of **autonomy**, performing **actions** in its environment based on **information** (sensors, feedback) received from the **environment**.
  - Obvious “agent” for us?

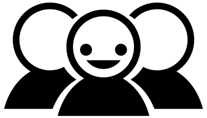


Panait, L., & Luke, S. (2005). Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, *11*(3), 387–434. https://doi.org/10.1007/s10458-005-2631-2
https://www.quora.com/What-is-the-relationship-between-agent-based-modeling-and-reinforcement-learning

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Agent-based modeling (ABM)

- Agent = Computational mechanism that exhibits a high degree of **autonomy**, performing **actions** in its environment based on **information** (sensors, feedback) received from the **environment**.
  - Obvious "agent" for us?



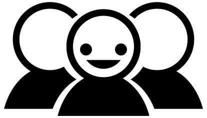
- But what for? How are they useful?

Panait, L., & Luke, S. (2005). Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, *11*(3), 387–434. https://doi.org/10.1007/s10458-005-2631-2
https://www.quora.com/What-is-the-relationship-between-agent-based-modeling-and-reinforcement-learning

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Agent-based modeling (ABM)

- Agent = Computational mechanism that exhibits a high degree of **autonomy**, performing **actions** in its environment based on **information** (sensors, feedback) received from the **environment**.
  - Obvious “agent” for us?



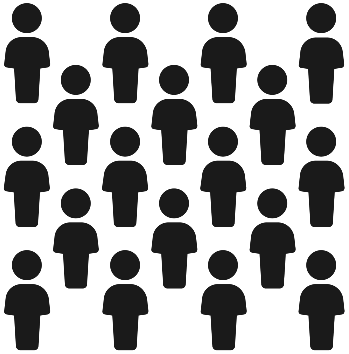
- But what for? How are they useful?
  - Helps us understand the effect of “multiple agents” **interacting** with **each other** and with their **environment**.

Panait, L., & Luke, S. (2005). Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, *11*(3), 387–434. https://doi.org/10.1007/s10458-005-2631-2
https://www.quora.com/What-is-the-relationship-between-agent-based-modeling-and-reinforcement-learning

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Examples

- Simulate occupancy for HVAC design and sizing. (**space-based approach**)



Impact of **aggregated** occupant behavior in a space of a certain category.

Erickson, V. L., Lin, Y., Kamthe, A., Brahme, R., Surana, A., Cerpa, A. E., Sohn, M. D., & Narayanan, S. (2009). Energy efficient building environment control strategies using real-time occupancy measurements. *BUILDSYS 2009 - Proceedings of the 1st ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings, Held in Conjunction with ACM SenSys 2009*, 19–24. https://doi.org/10.1145/1810279.1810284

Li, Z., Heo, Y., & Augenbroe, G. (2009). HVAC DESIGN INFORMED BY ORGANIZATIONAL SIMULATION Georgia Institute Of Technology , Atlanta , United States. *Ashrae Standard*, 2198–2203.

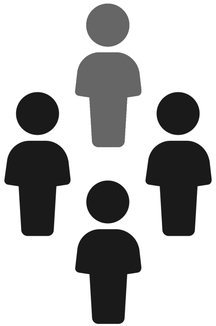
Mosteiro-Romero, M., Hischier, I., Fonseca, J. A., & Schlueter, A. (2020). A novel population-based occupancy modeling approach for district-scale simulations compared to standard-based methods. *Building and Environment*, *181*, 107084. https://doi.org/10.1016/j.buildenv.2020.107084

Happle, G., Fonseca, J. A., & Schlueter, A. (2018). A review on occupant behavior in urban building energy models. *Energy and Buildings*, *174*, 276–292. https://doi.org/10.1016/j.enbuild.2018.06.030

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Examples

- Simulate occupancy at the urban-level for energy demand (**person-based approach**)



Impact of the presence of a **specific person** in a certain space

Erickson, V. L., Lin, Y., Kamthe, A., Brahme, R., Surana, A., Cerpa, A. E., Sohn, M. D., & Narayanan, S. (2009). Energy efficient building environment control strategies using real-time occupancy measurements. *BUILDSYS 2009 - Proceedings of the 1st ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings, Held in Conjunction with ACM SenSys 2009*, 19–24. https://doi.org/10.1145/1810279.1810284

Li, Z., Heo, Y., & Augenbroe, G. (2009). HVAC DESIGN INFORMED BY ORGANIZATIONAL SIMULATION Georgia Institute Of Technology , Atlanta , United States. *Ashrae Standard*, 2198–2203.

Mosteiro-Romero, M., Hischier, I., Fonseca, J. A., & Schlueter, A. (2020). A novel population-based occupancy modeling approach for district-scale simulations compared to standard-based methods. *Building and Environment*, *181*, 107084. https://doi.org/10.1016/j.buildenv.2020.107084

Happle, G., Fonseca, J. A., & Schlueter, A. (2018). A review on occupant behavior in urban building energy models. *Energy and Buildings*, *174*, 276–292. https://doi.org/10.1016/j.enbuild.2018.06.030

Reinforcement Learning (RL)

- Framework where an *agent* learns to perform **actions** in an **environment** (that has **states**) so as to maximize a **reward**.

- Main components
  - Environment (problem to be solved)
  - Agent (entity that learns an algorithm)

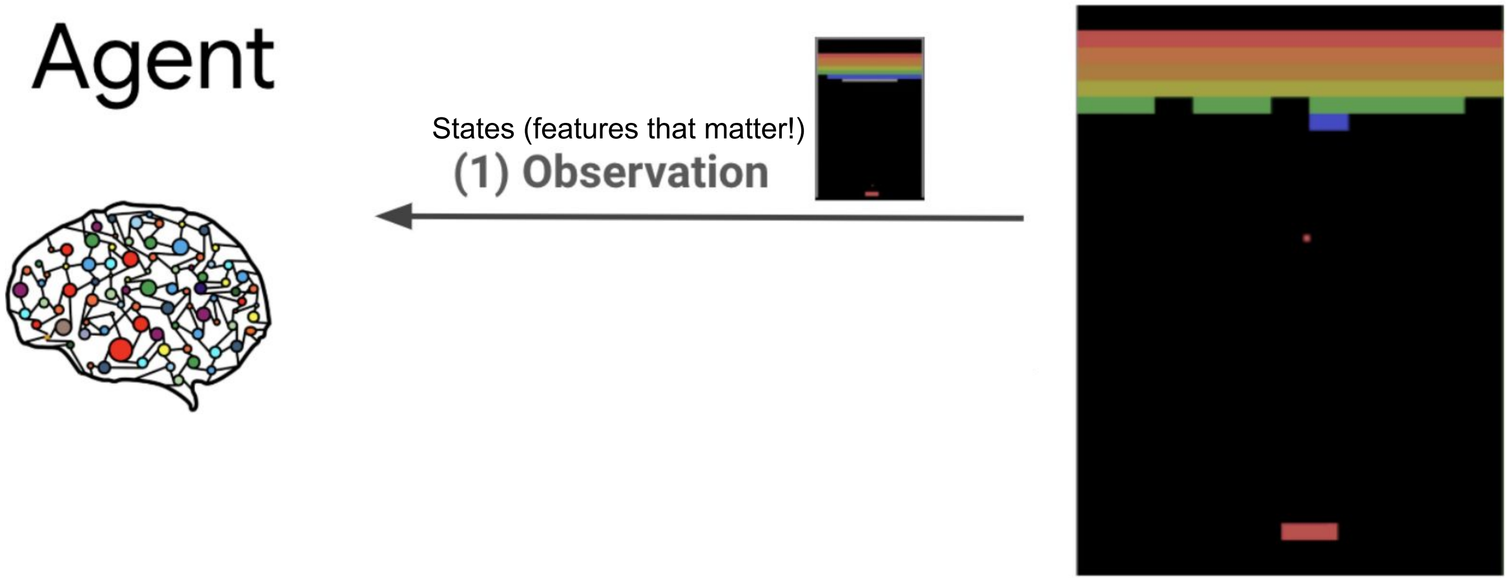
https://www.tensorflow.org/agents/tutorials/0_intro_rl

Reinforcement Learning (RL)

- How exactly does it work?

At every simulation step …

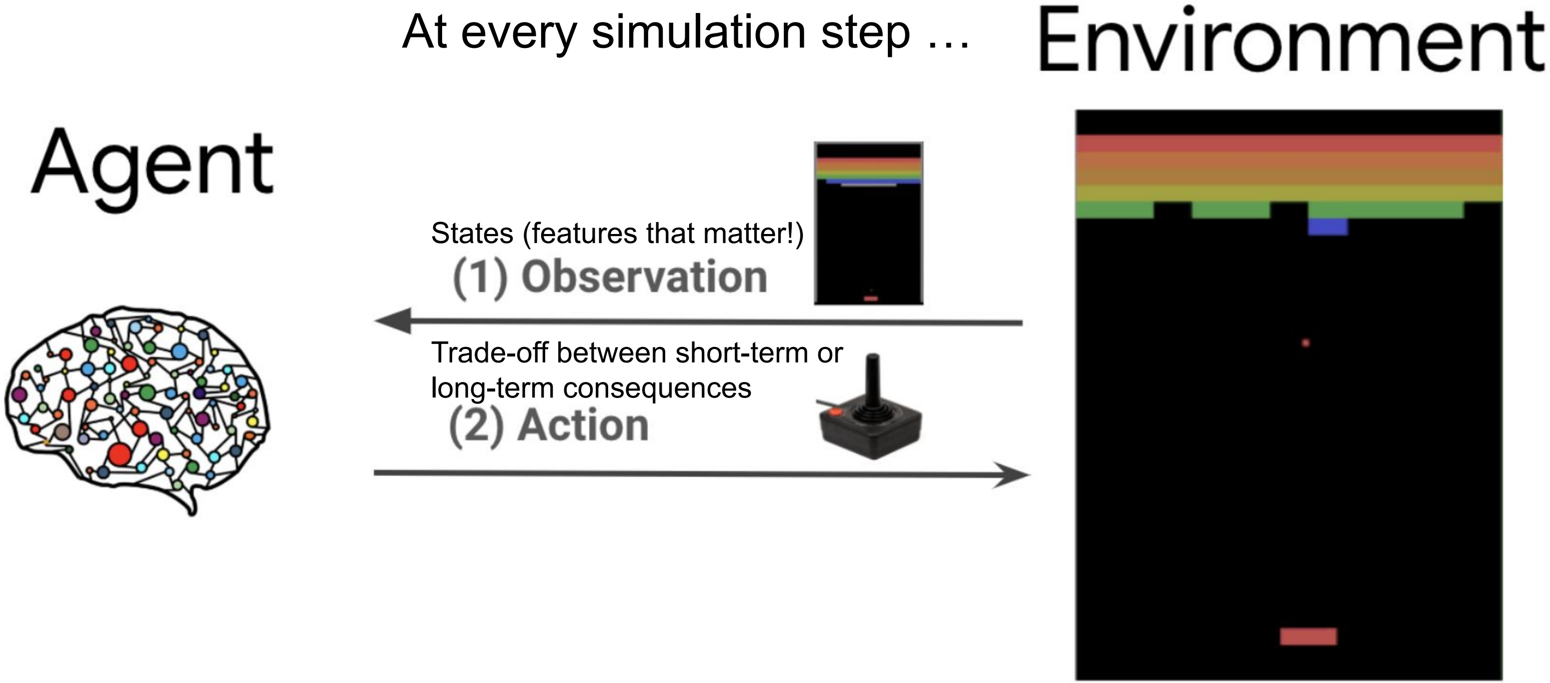
Environment



Reinforcement Learning (RL)

- How exactly does it work?

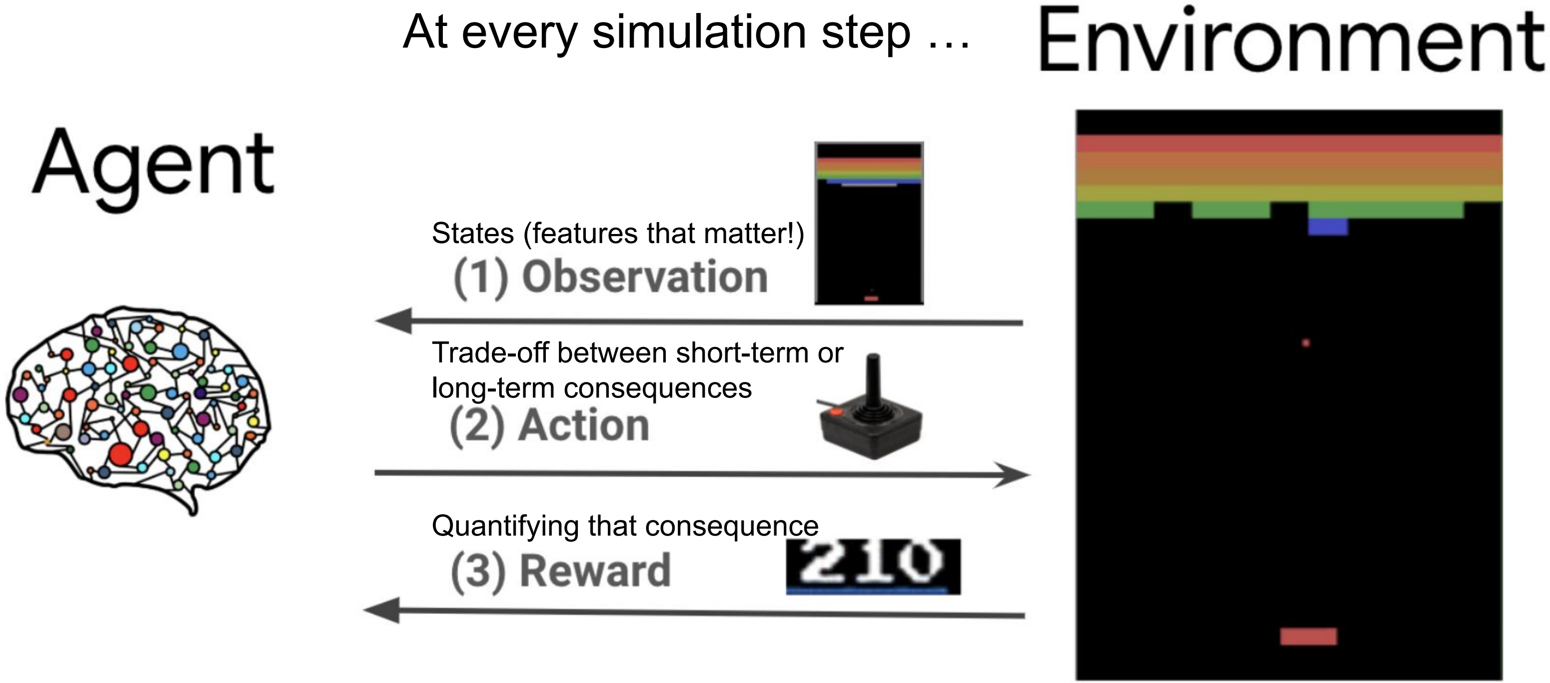
At every simulation step …



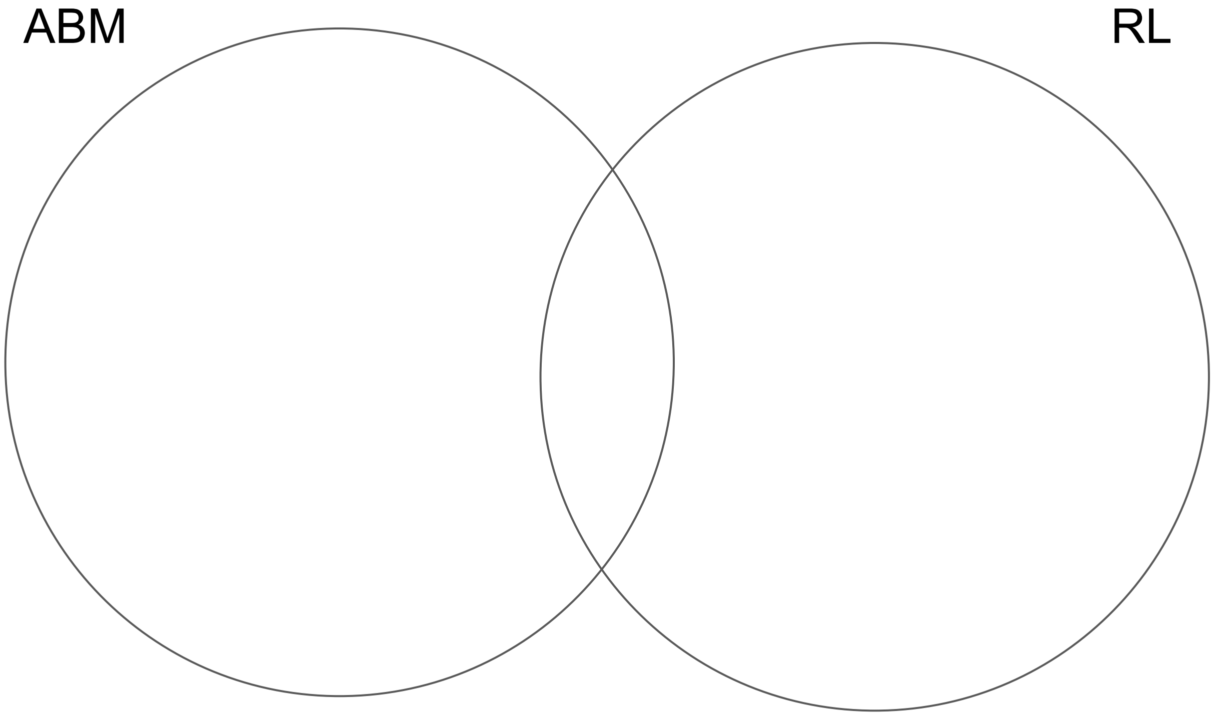
ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Reinforcement Learning (RL)

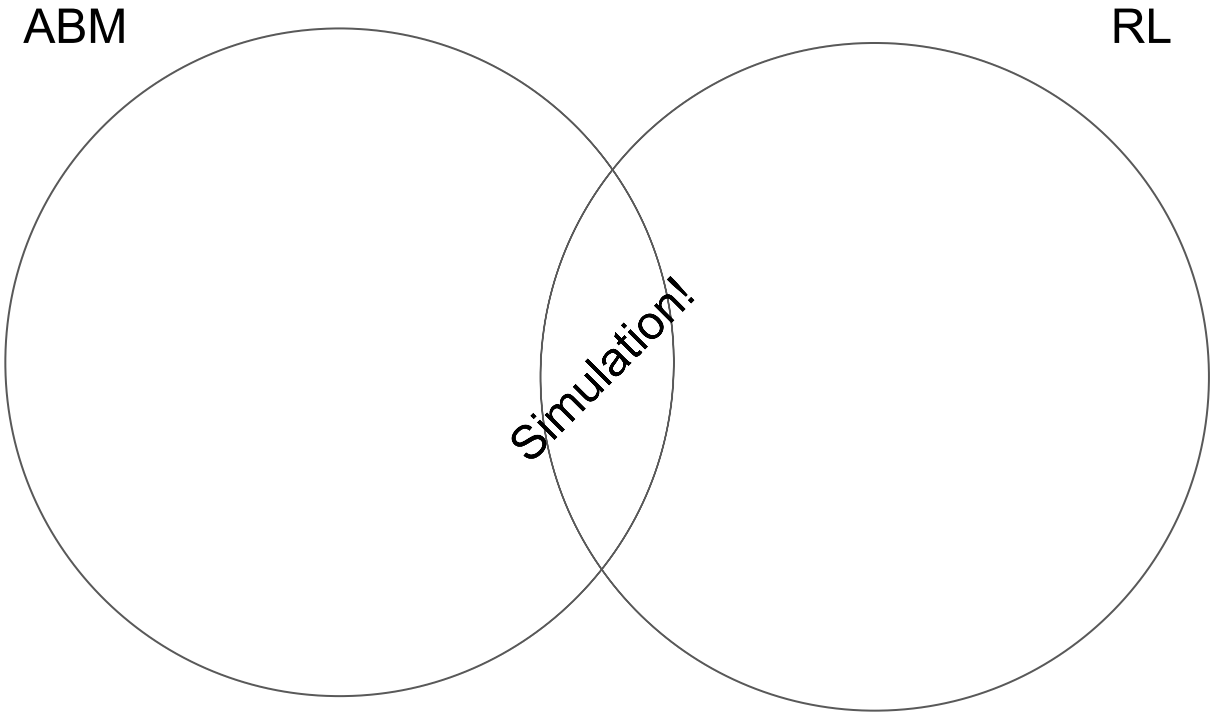
- How exactly does it work?



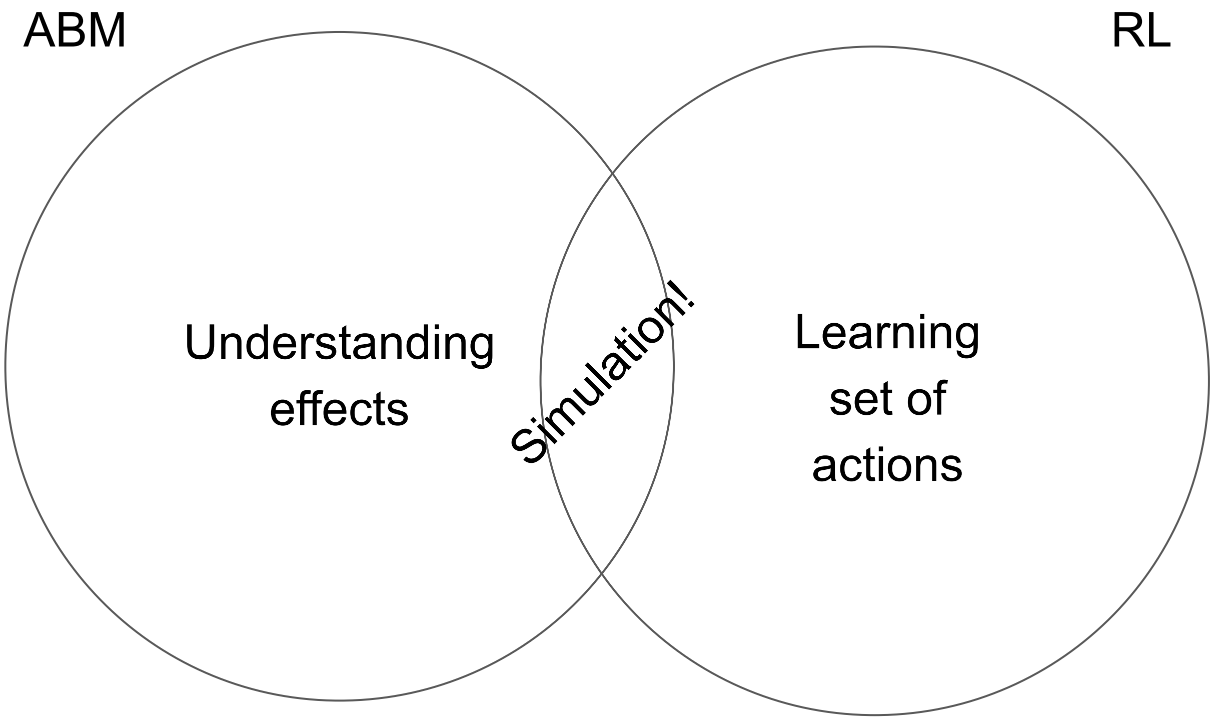
ABM vs RL



ABM vs RL



ABM vs RL


ABM
RL
Understanding effects
Simulation!
Learning set of actions

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

OpenAI gym

Open source interface to reinforcement learning tasks.

The gym library provides an easy-to-use suite of reinforcement learning tasks.

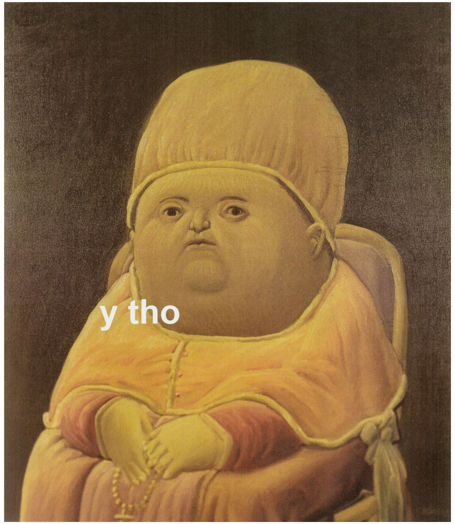
https://gym.openai.com/

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

OpenAI gym

Open source interface to reinforcement learning tasks.

The gym library provides an easy-to-use suite of reinforcement learning tasks.



https://gym.openai.com/

[OpenAI gym](https://gym.openai.com/docs/)

RL it's pretty damn cool:

- RL is **very general**, encompassing all problems that involve making a **sequence of decisions**
- RL algorithms have started to achieve **good results** in many **difficult environments.**

BUT! Research slowed down because …

- There's a need for **better benchmarks**
- Lack of **standardization of environments** used in publications

https://gym.openai.com/docs/

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Here is where “gym” comes into play

(**Pokemon gen1 is the best gen, don’t come at me**)

https://gym.openai.com/docs/

Overall structure of an openAI gym project

```
import gym
from gym import spaces

class CustomEnv(gym.Env):
  """Custom Environment that follows gym interface"""
  metadata = {'render.modes': ['human']}

  def __init__(self, arg1, arg2, ...):
    super(CustomEnv, self).__init__()    # Define action and observation space
    # They must be gym.spaces objects    # Example when using discrete actions:
    self.action_space = spaces.Discrete(N_DISCRETE_ACTIONS)    # Example for using image as input:
    self.observation_space = spaces.Box(low=0, high=255, shape=
                    (HEIGHT, WIDTH, N_CHANNELS), dtype=np.uint8)

  def step(self, action):
    # Execute one time step within the environment
    ...  def reset(self):
    # Reset the state of the environment to an initial state
    ...  def render(self, mode='human', close=False):
    # Render the environment to the screen
    ...
```

https://towardsdatascience.com/creating-a-custom-openai-gym-environment-for-stock-trading-be532be3910e

Hands on !

1. Clone the github repository https://github.com/buds-lab/abm-demo
2. Install the environments
   a. `conda env create --file environment.yaml`
   b. `pip install -r requirements.txt`
3. Let's just launch the environment doing some random actions. From `openai-gym/` Let's run `cartpole_example_1.py`
4. Now let's observe the environment's state at each time step, still with random actions though. From `openai-gym/` Let's run `cartpole_example_2.py`
5. But we want to LEARN! From `openai-gym/` Let's run `cartpole_example_3.py`

https://gym.openai.com/docs/

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Cool learning and all, but what about the more traditional ABM? What about CitySim? MATSim?

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Cool learning and all, but what about the more traditional ABM? What about CitySim? MATSim?

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Overall structure of an openAI gym project

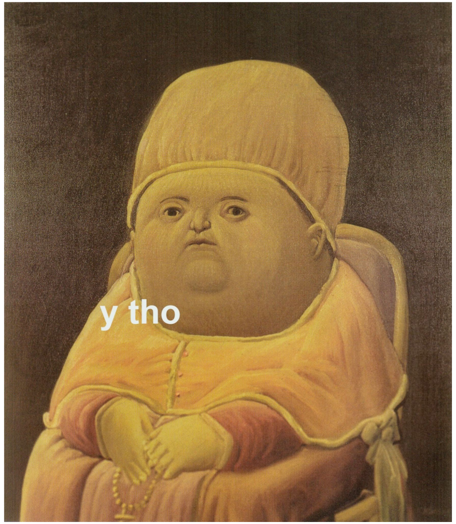
```
import gym
from gym import spaces

class CustomEnv(gym.Env):
  """Custom Environment that follows gym interface"""
  metadata = {'render.modes': ['human']}

  def __init__(self, arg1, arg2, ...):
    super(CustomEnv, self).__init__()    # Define action and observation space
    # They must be gym.spaces objects    # Example when using discrete actions:
    self.action_space = spaces.Discrete(N_DISCRETE_ACTIONS)    # Example for using image as input:
    self.observation_space = spaces.Box(low=0, high=255, shape=
                    (HEIGHT, WIDTH, N_CHANNELS), dtype=np.uint8)

  def step(self, action):
    # Execute one time step within the environment
    ...  def reset(self):
    # Reset the state of the environment to an initial state
    ...  def render(self, mode='human', close=False):
    # Render the environment to the screen
    ...
```

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

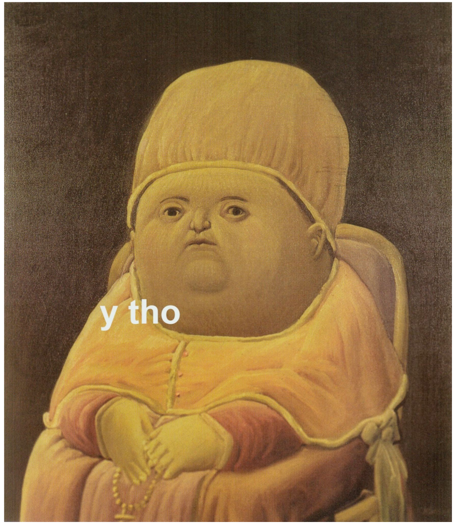
Overall structure of an openAI gym project

```
import gym
from gym import spaces

class CustomEnv(gym.Env):
  """Custom Environment that follows gym interface"""
  metadata = {'render.modes': ['human']}

  def __init__(self, arg1, arg2, ...):
    super(CustomEnv, self).__init__()    # Define action and observation space
    # They must be gym.spaces objects    # Example when using discrete actions:
    self.action_space = spaces.Discrete(N_DISCRETE_ACTIONS)    # Example for using image as input:
    self.observation_space = spaces.Box(low=0, high=255, shape=
                    (HEIGHT, WIDTH, N_CHANNELS), dtype=np.uint8)

  def step(self, action):
    # Execute one time step within the environment
    ...  def reset(self):
    # Reset the state of the environment to an initial state
    ...  def render(self, mode='human', close=False):
    # Render the environment to the screen
    ...
```

We can simply define “agents” (without learning) to perform actions at each step of the simulation!


y tho

ABM - RL - ABM vs RL - OpenAI - ComfortLearn



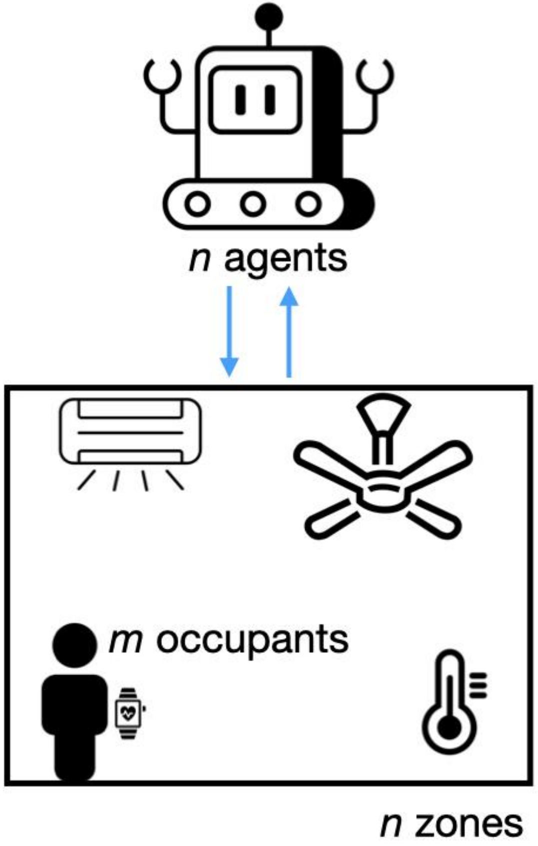
Gym environments are becoming more popular for standardised benchmarking in the built environment

- CityLearn
- GridLearn
- Gym-Eplus

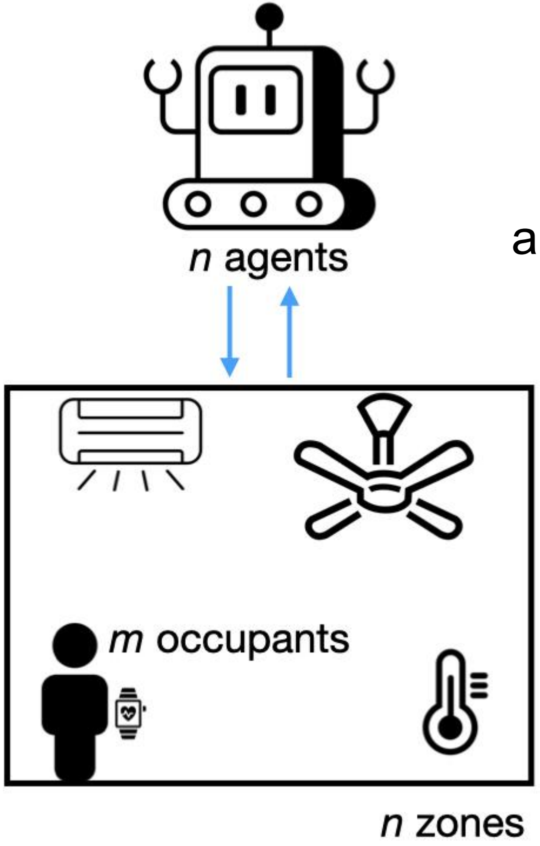
"Need for more benchmarking of control algorithms"

Wölfle, D., Vishwanath, A., & Schmeck, H. (2020). A Guide for the Design of Benchmark Environments for Building Energy Optimization. *BuildSys 2020 - Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 220–229. https://doi.org/10.1145/3408308.3427614

So what is ComfortLearn?



So what is ComfortLearn?

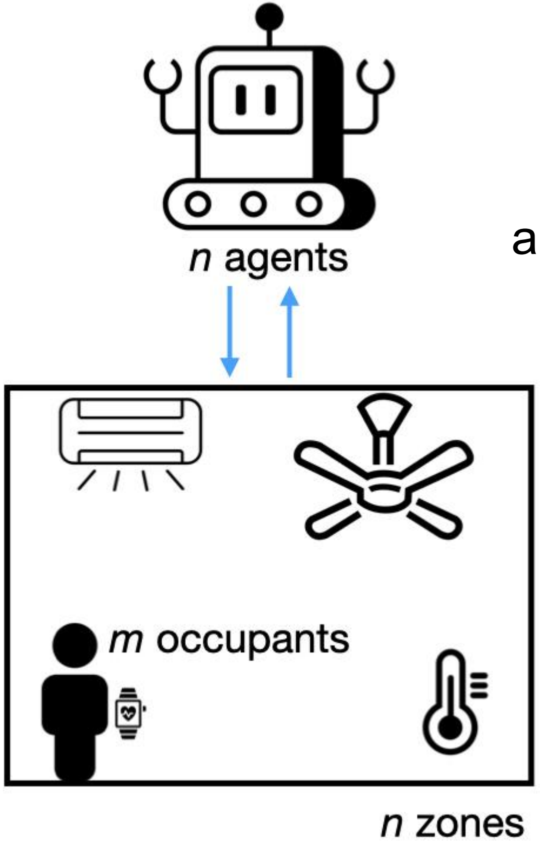


RL agent that learns set-points and occupant allocation in different zones

Agent-based occupants from real occupant longitudinal data

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

So what is ComfortLearn?



RL agent that learns set-points and occupant allocation in different zones

Agent-based occupants from real occupant longitudinal data

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Hands on (again!)!

1. From the same github repository go to `comfortlearn/`
2. The overall structure of this ABM using a gym environment is

```
# main file
import ...

###
# load zone(s) energy model or historical data from config files
###

# instantiate actual environment
env = ComfortLearn(**env_params)
observations_spaces, actions_spaces = env.get_state_action_spaces()

###
# load agent stuff, as of now it's a big TODO though
###

# start simulation
state = env.reset()
done = False

actions = agent.select_action(state)

while not done:
    next_state, reward, done = env.step(actions)
    action_next = agent.select_action(next_state)
    action = action_next
```

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Hands on (again!)!

1. From the same github repository go to `comfortlearn/`
2. The overall structure of this ABM using a gym environment is

```
# main file
import ...

###
# load zone(s) energy model or historical data from config files
###

# instantiate actual environment
env = ComfortLearn(**env_params)
observations_spaces, actions_spaces = env.get_state_action_spaces()

###
# load agent stuff, as of now it's a big TODO though
###

# start simulation
state = env.reset()
done = False

actions = agent.select_action(state)

while not done:
    next_state, reward, done = env.step(actions)
    action_next = agent.select_action(next_state)
    action = action_next
```

```
# comfortlearn file

import ...

class ComfortLearn(gym.Env):
    def __init__(args):
        # load states and actions for each zone

        # create occupants objects and load their data, this will be ABM!

        # create zone objects and load their data


    def step(self, actions):
        # where the magic happens! How does the environment changes at every time step?
        # right now, the environment just evolves based on historical data, meaning we just read
        # the next row in the dataframe

      return states, reward, done

    def reset(self):
        # let's take it from the top, shall we?
        # we restart values and conditions to the very beginning

    # auxiliary, getters, and setters functions are also used
```

ABM - RL - ABM vs RL - OpenAI - ComfortLearn

Hands on (again!)!

1. From the same github repository go to `comfortlearn/`
2. The overall structure of this ABM using a gym environment is
3. So, let's run:

```
python main.py configs/baseline_tolerance_week.yaml
```

OpenAI gyms are very versatile and a good (and promising!) option for agent-based with or without RL!

OpenAI gyms are very versatile and a good (and promising!) option for agent-based with or without RL!

(and also Vim and working in the terminal is extremely cool)