Lab meeting

Robust representations for olfactory-spatial association learning

---

Andrea Pierré

March 11, 2025

Outline

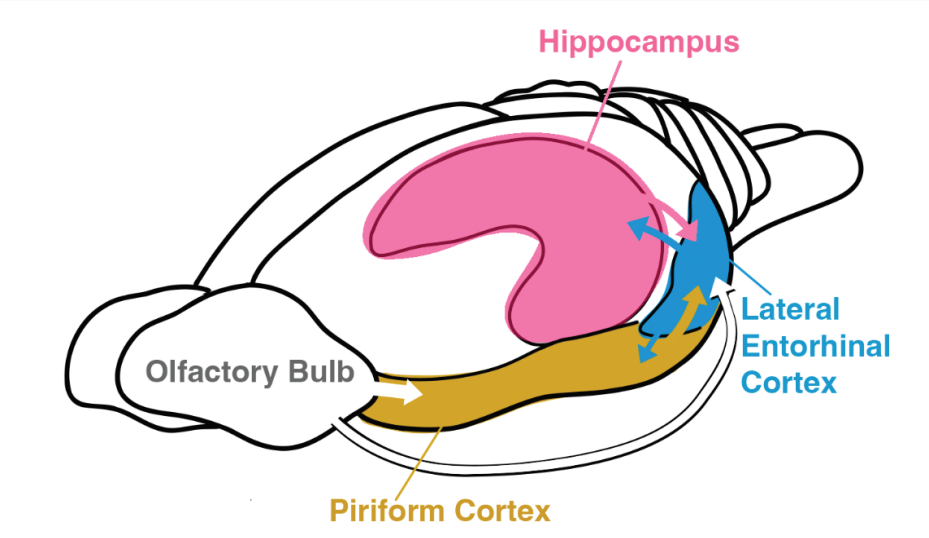
1. Project recap
2. Modeling & Simulation
3. What does the network learn?
4. Conclusion

Outline

1. Project recap
2. Modeling & Simulation
3. What does the network learn?
4. Conclusion

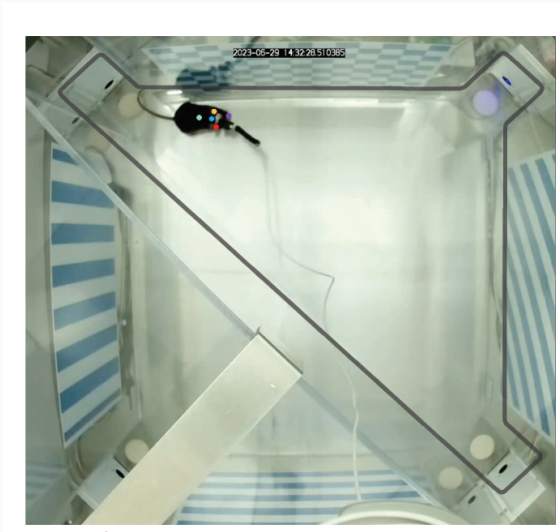
The LEC is key to sensory associations and spatial memory

- Piriform Cortex encodes olfactory information
- Hippocampus encodes spatial information
- Lateral Entorhinal Cortex (LEC) encodes both olfactory & spatial information

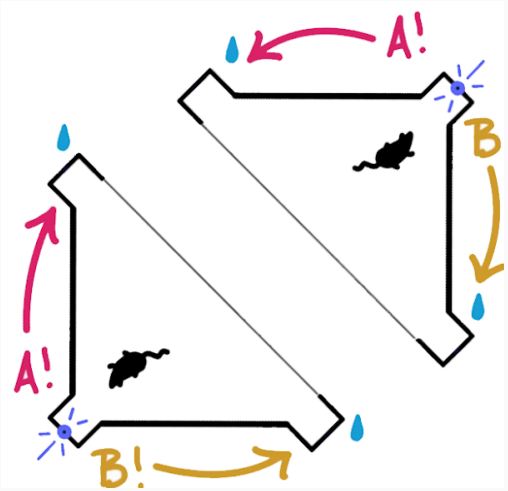


Poo et al., 2022
Bitzenhofer et al., 2022
Lee et al., 2021

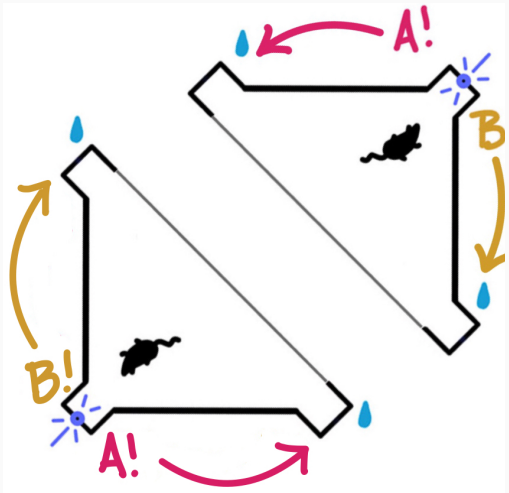
Half triangle task for olfactory-spatial association learning



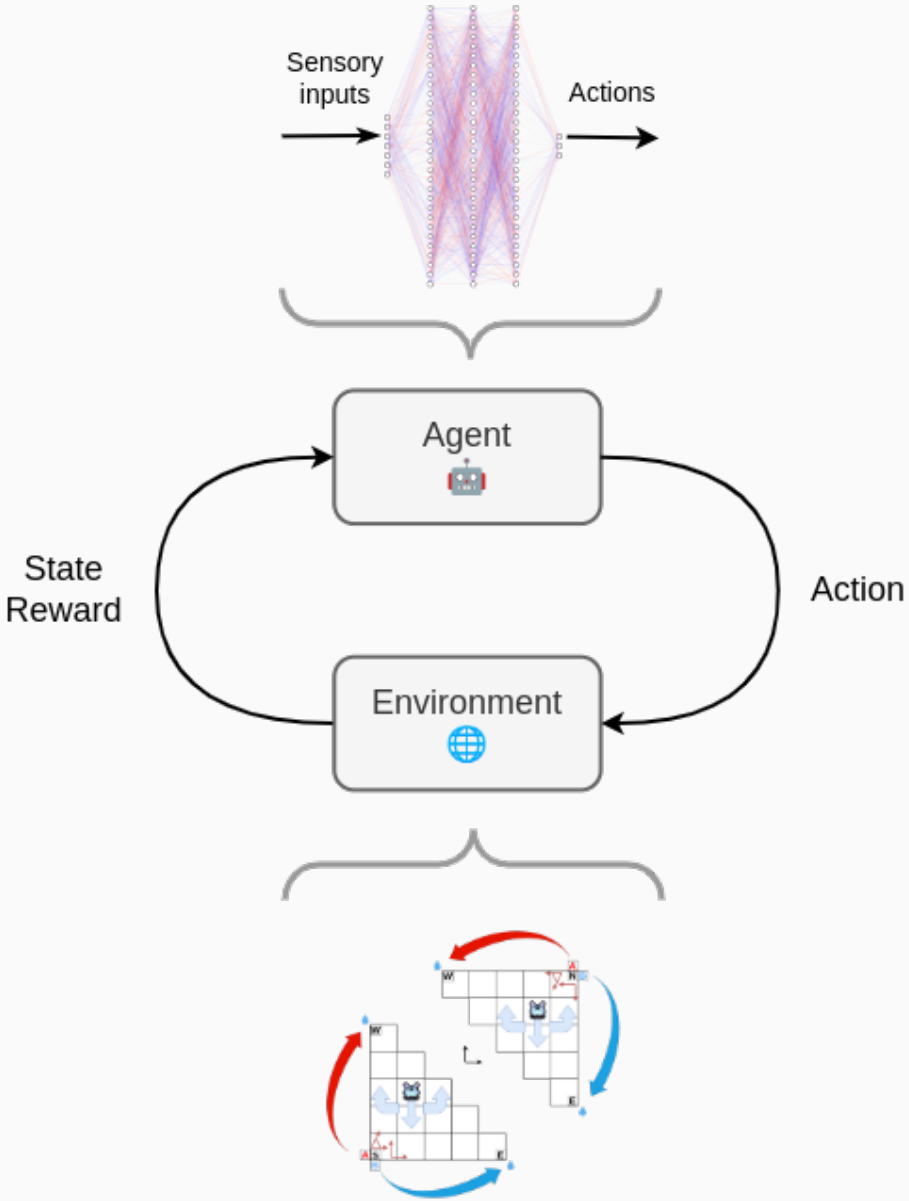
East/West task



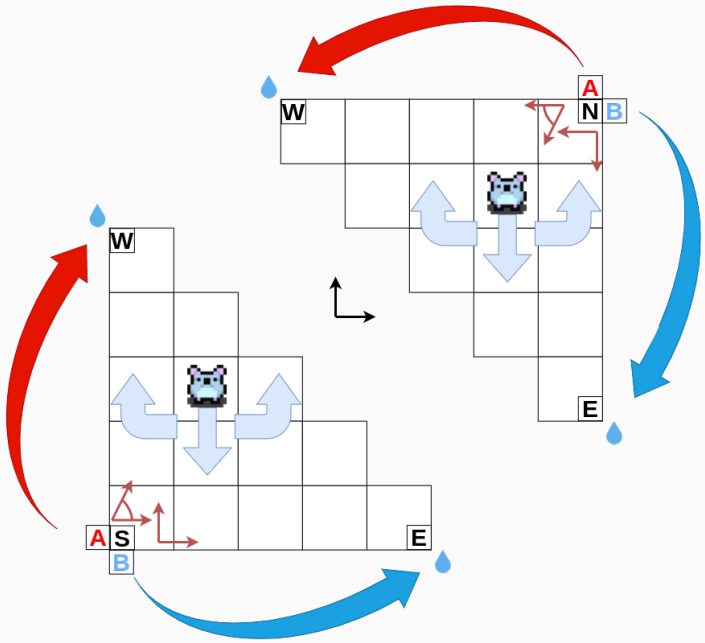
Left/Right task



Deep Reinforcement Learning model



The environment



- 3 actions: $\Leftarrow$ $\Uparrow$ $\Rightarrow$
- Duplicated coordinates inputs:
  - Cartesian coordinates from north & south port
  - Polar coordinates from north & south port

Questions & Hypothesis

Questions

- What function does the network learn?
- How the constrains of the task affect learning & the representations learned?
- How do the representations learned compare between the *in vivo* and the *in silico* neurons?

Hypothesis

- The network will use the most efficient coordinate information based on the task
- The structure of the network's weights will reflect this prioritization of information

Questions & Hypothesis

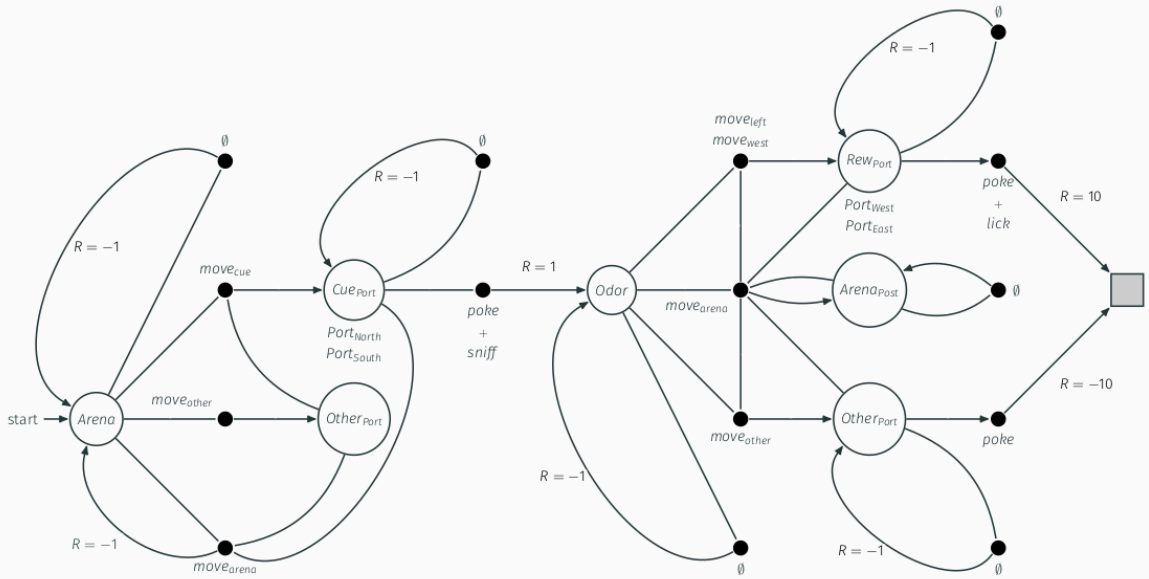
Questions

- What function does the network learn?
- How the constrains of the task affect learning & the representations learned?
- How do the representations learned compare between the *in vivo* and the *in silico* neurons?

Hypothesis

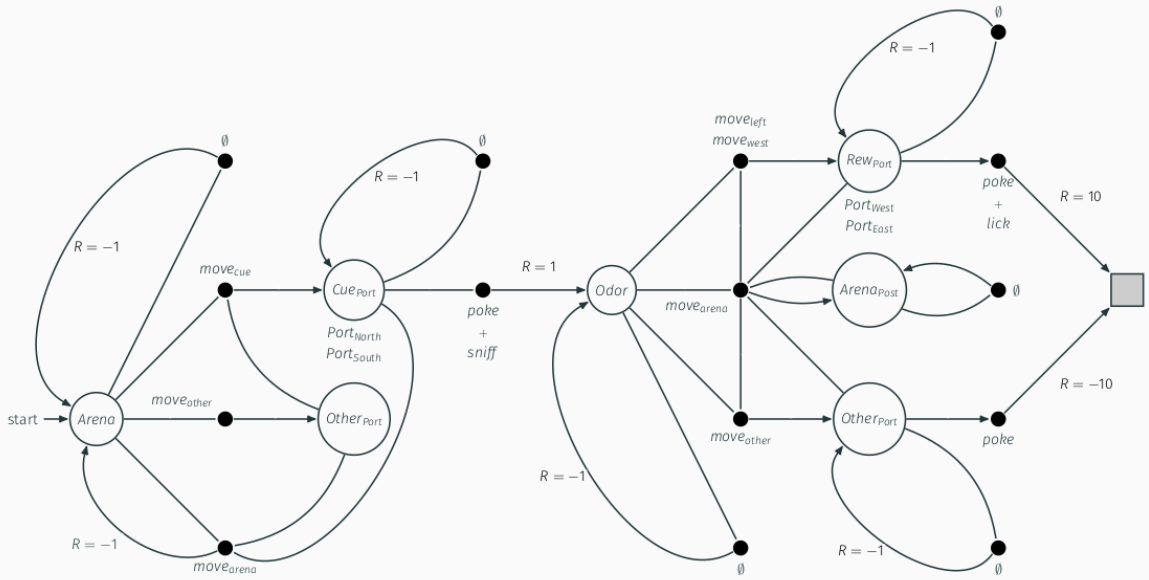
- The network will use the most efficient coordinate information based on the task
- The structure of the network's weights will reflect this prioritization of information

Looking back. . .



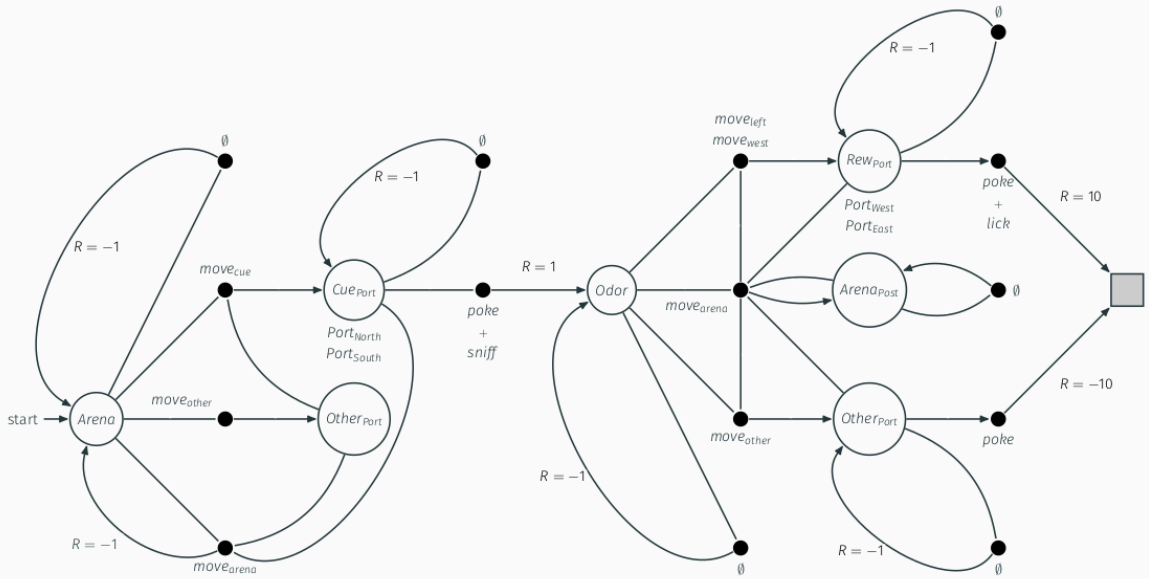
1. First step trying to define Olivia's experiment as a Markov Decision Process (MDP) in Julia
2. 2D tiles with tabular RL & function approximation in Python/NumPy
3. 2D coordinate system in Python/PyTorch
4. Duplicated coordinates experiment in Python/PyTorch

Looking back. . .



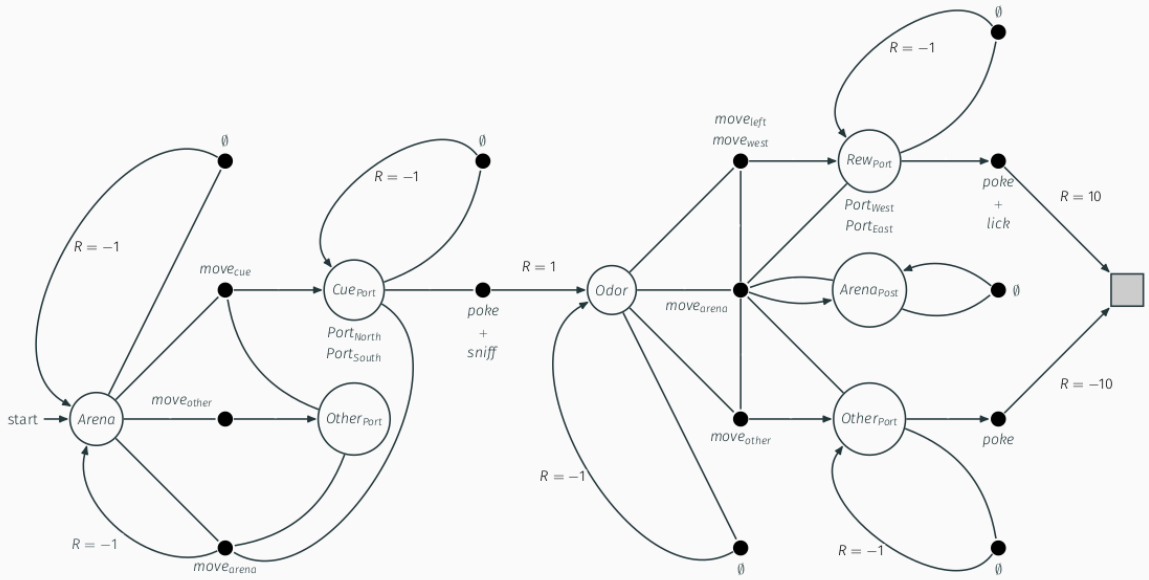
1. First step trying to define Olivia's experiment as a Markov Decision Process (MDP) in Julia
2. 2D tiles with tabular RL & function approximation in Python/NumPy
3. 2D coordinate system in Python/PyTorch
4. Duplicated coordinates experiment in Python/PyTorch

Looking back. . .



1. First step trying to define Olivia's experiment as a Markov Decision Process (MDP) in Julia
2. 2D tiles with tabular RL & function approximation in Python/NumPy
3. 2D coordinate system in Python/PyTorch
4. Duplicated coordinates experiment in Python/PyTorch

Looking back. . .

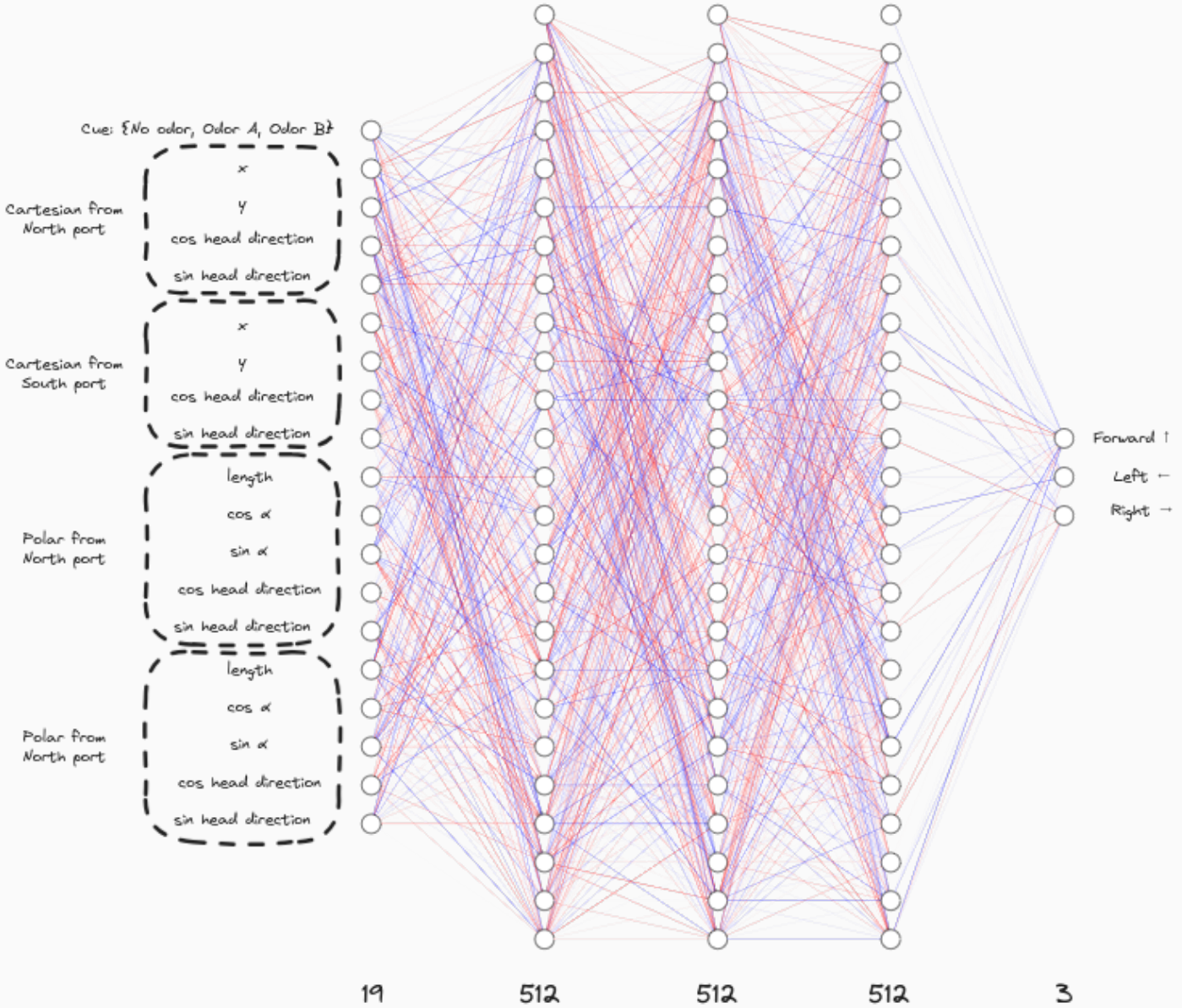


1. First step trying to define Olivia's experiment as a Markov Decision Process (MDP) in Julia
2. 2D tiles with tabular RL & function approximation in Python/NumPy
3. 2D coordinate system in Python/PyTorch
4. Duplicated coordinates experiment in Python/PyTorch

Outline

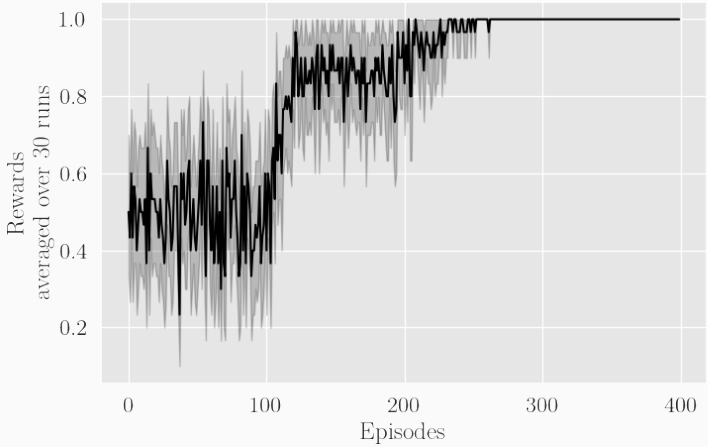
1. Project recap
2. Modeling & Simulation
3. What does the network learn?
4. Conclusion

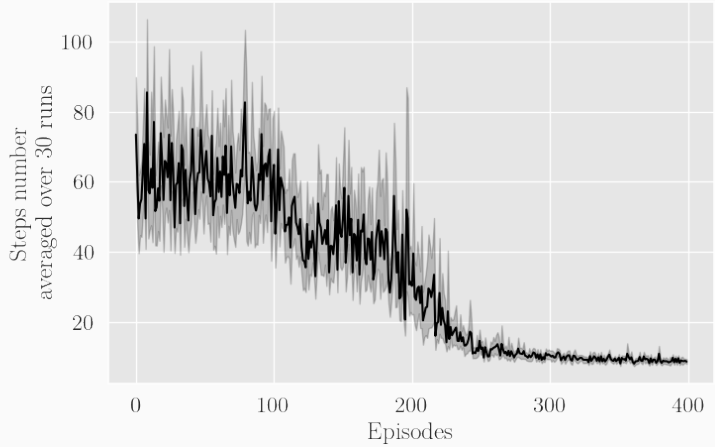
State space & network architecture



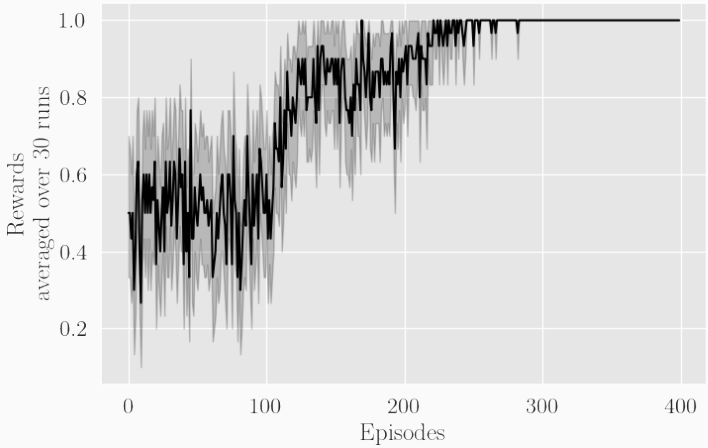
Training

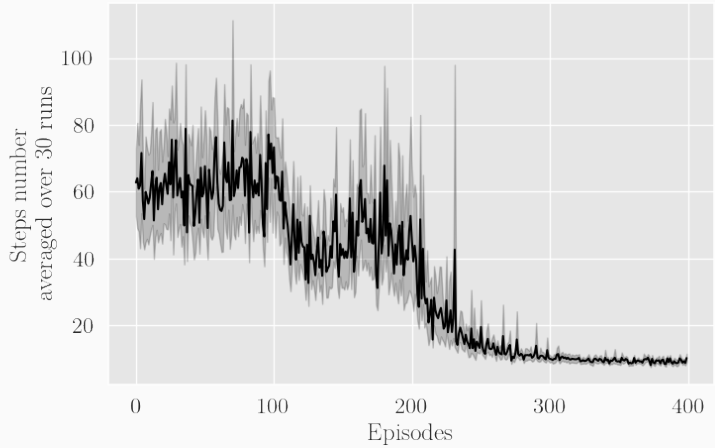
East/West



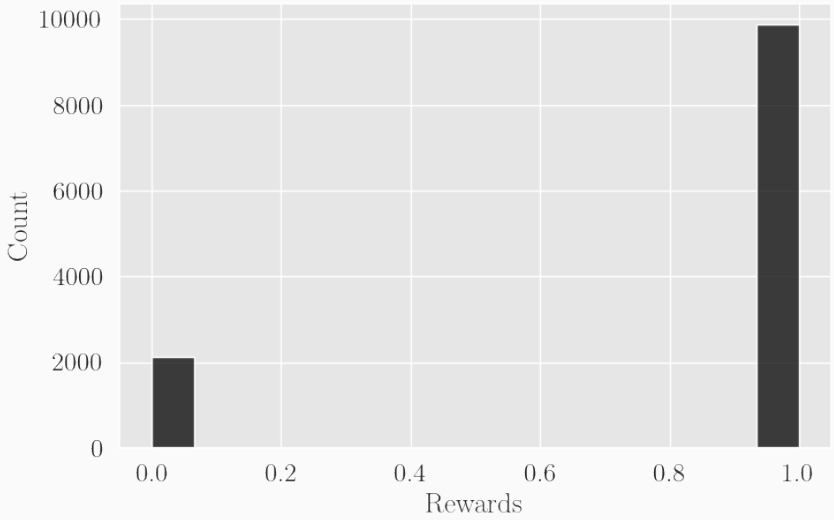


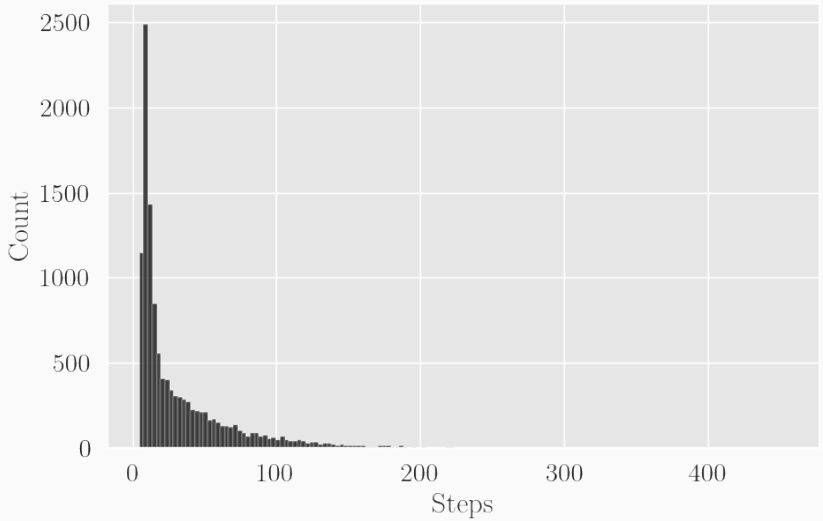
Left/Right

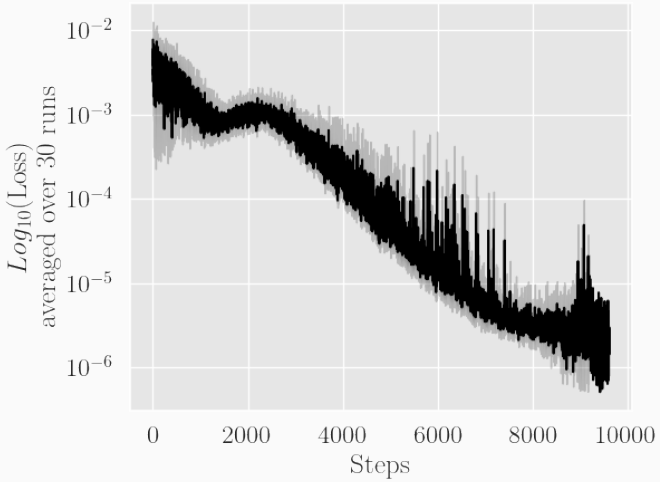


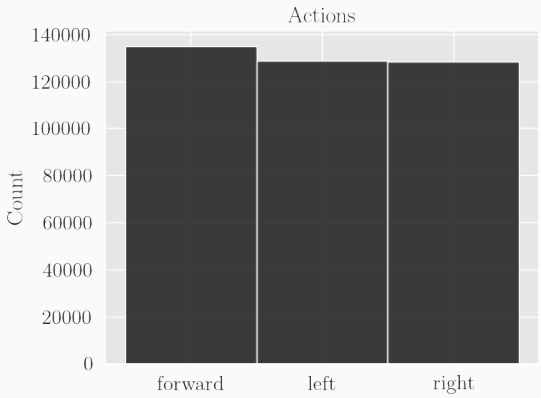


Training checks – East/West

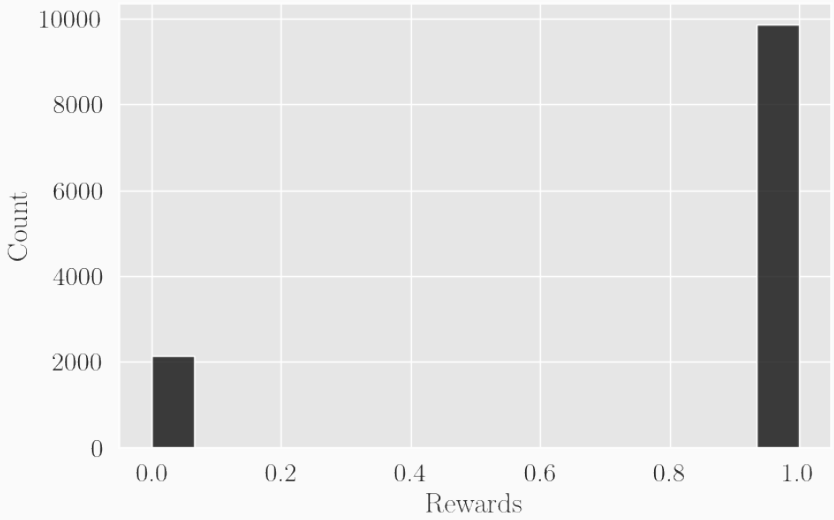


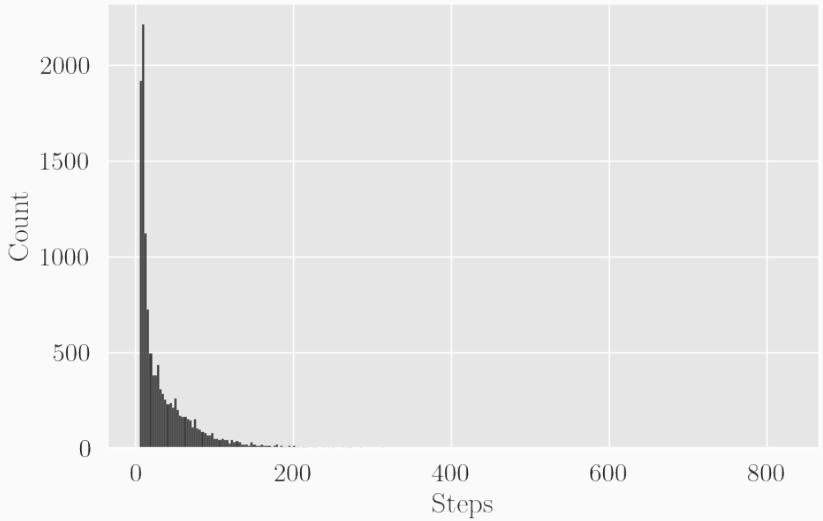


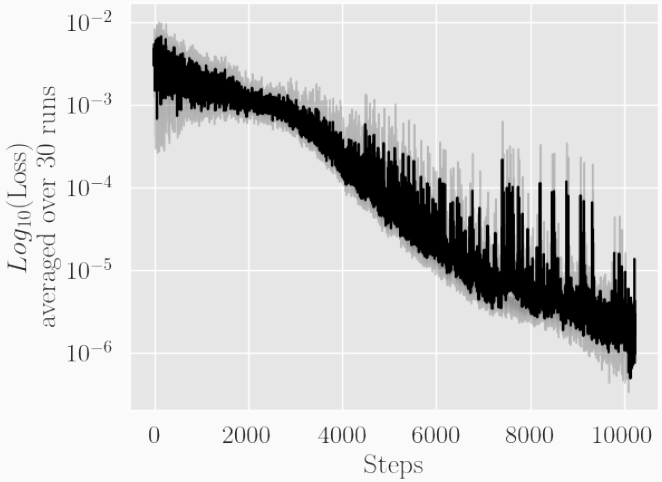


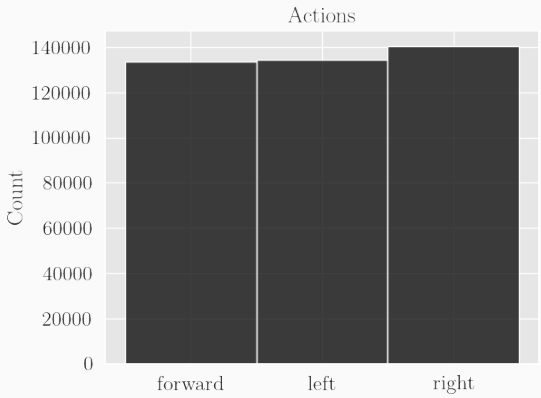


Training checks – Left/Right

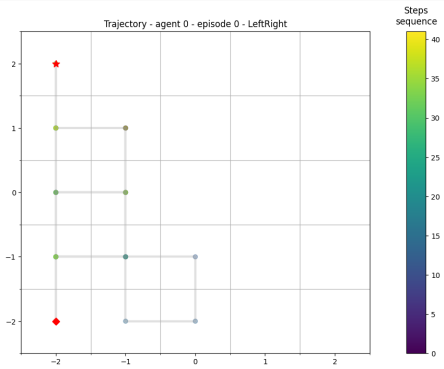


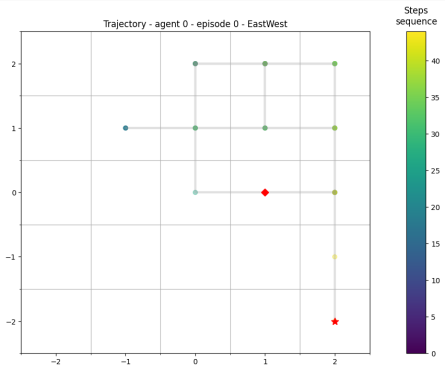


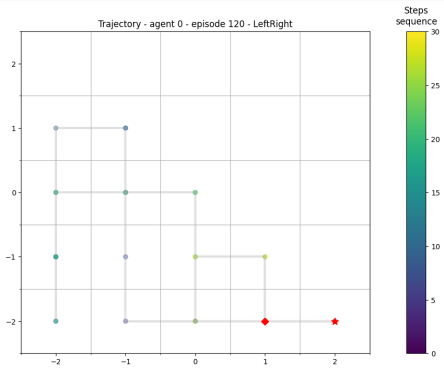


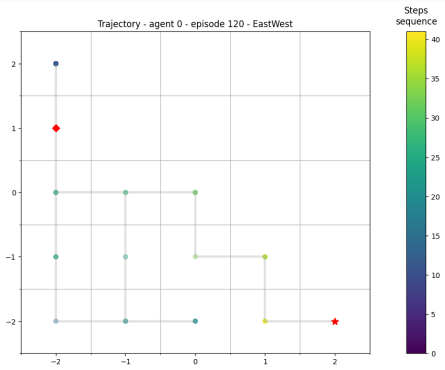


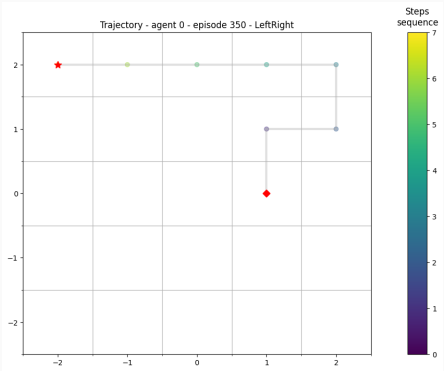
Agent behavior

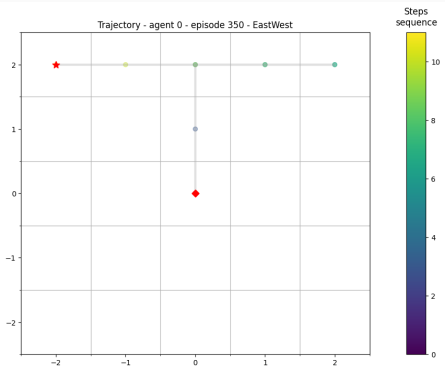








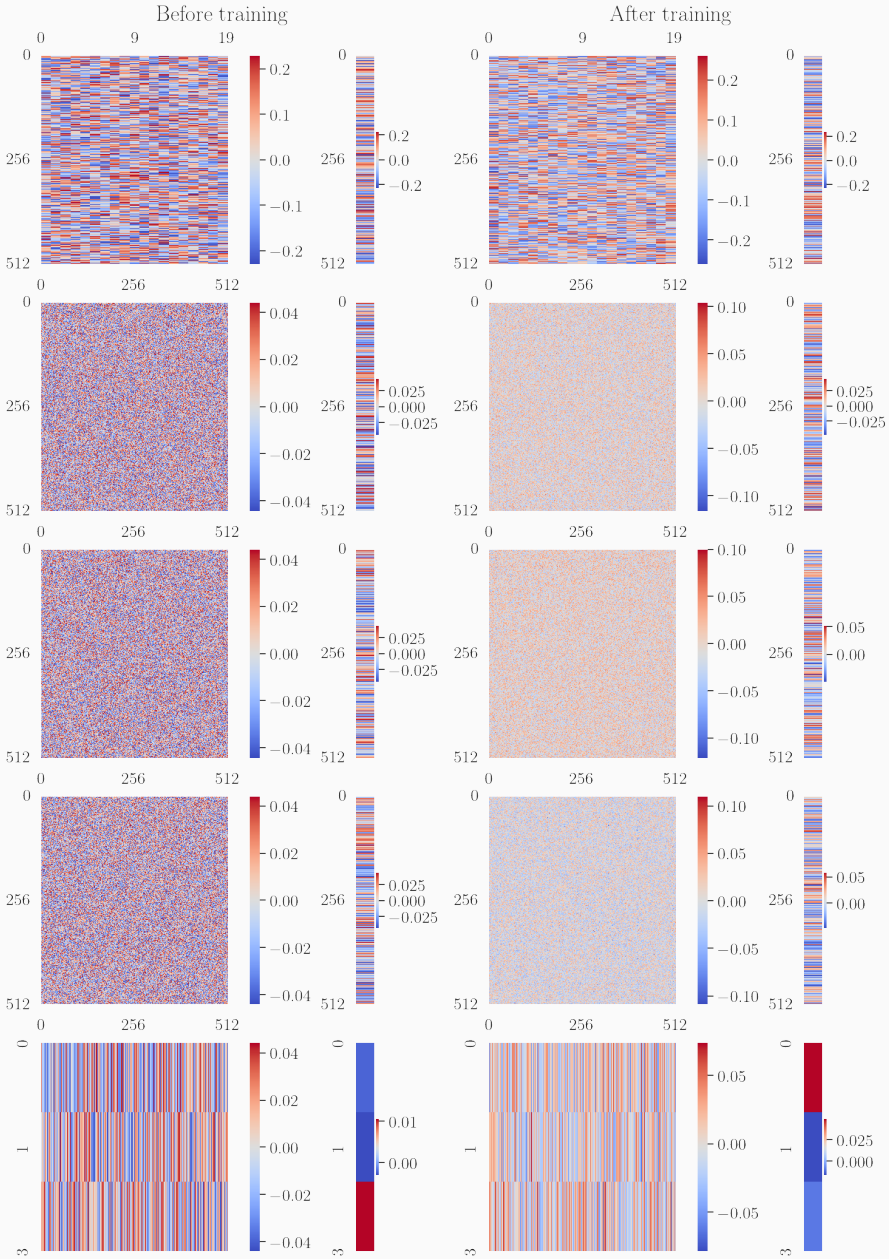




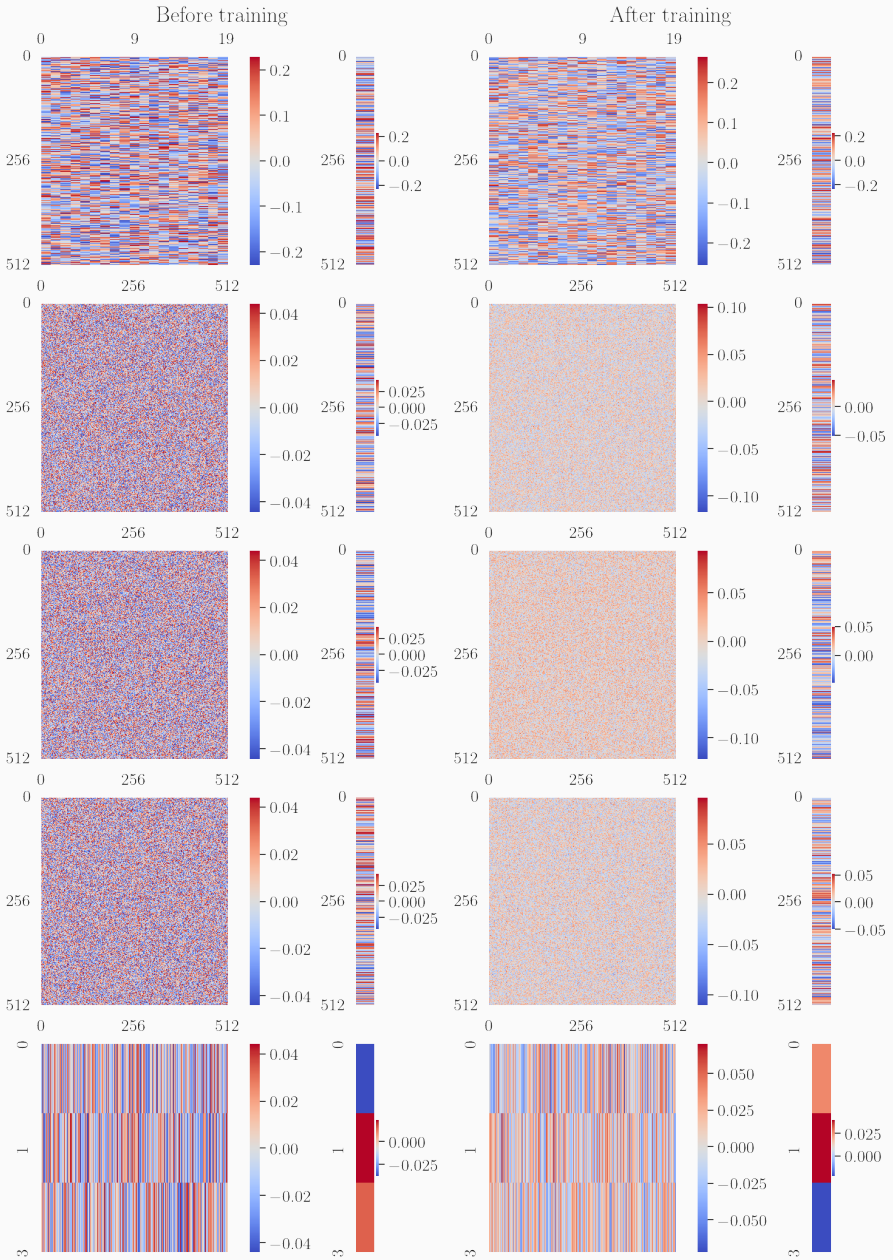
Outline

1. Project recap
2. Modeling & Simulation
3. What does the network learn?
4. Conclusion

Weights structure – East/West

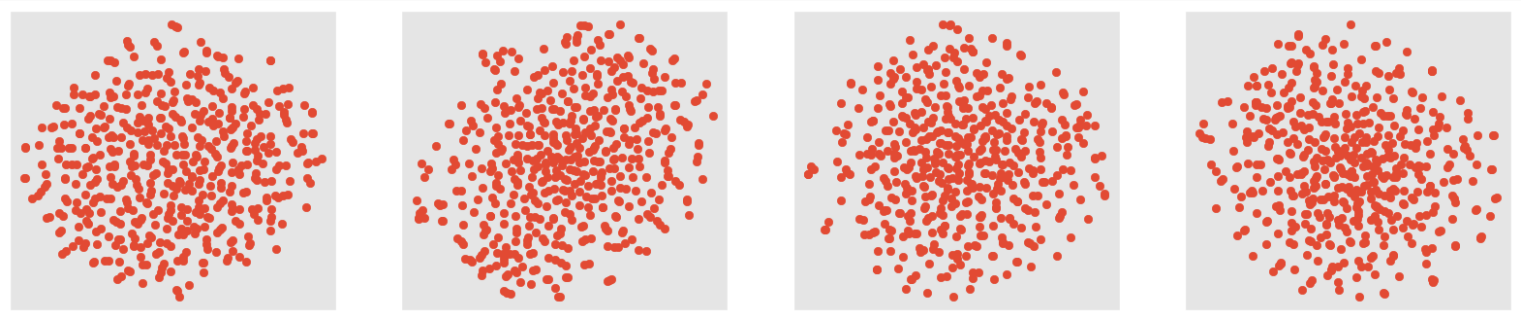


Weights structure – Left/Right

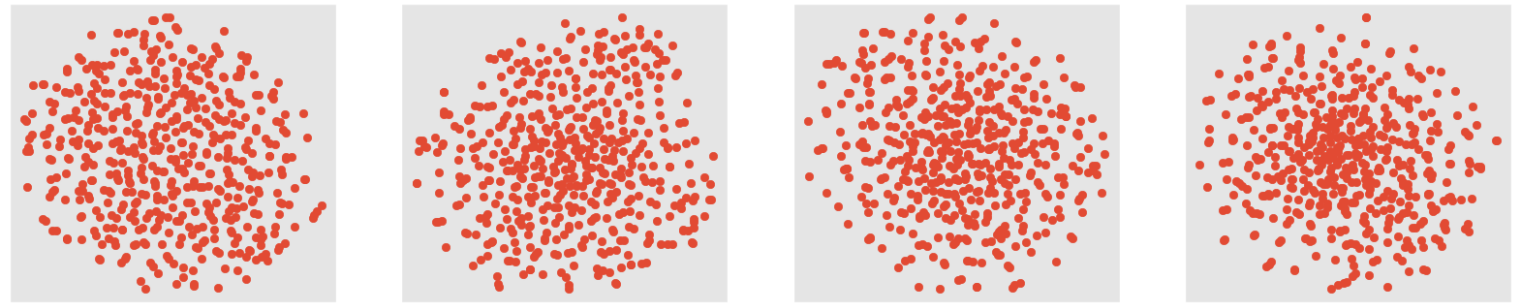


Weights clustering

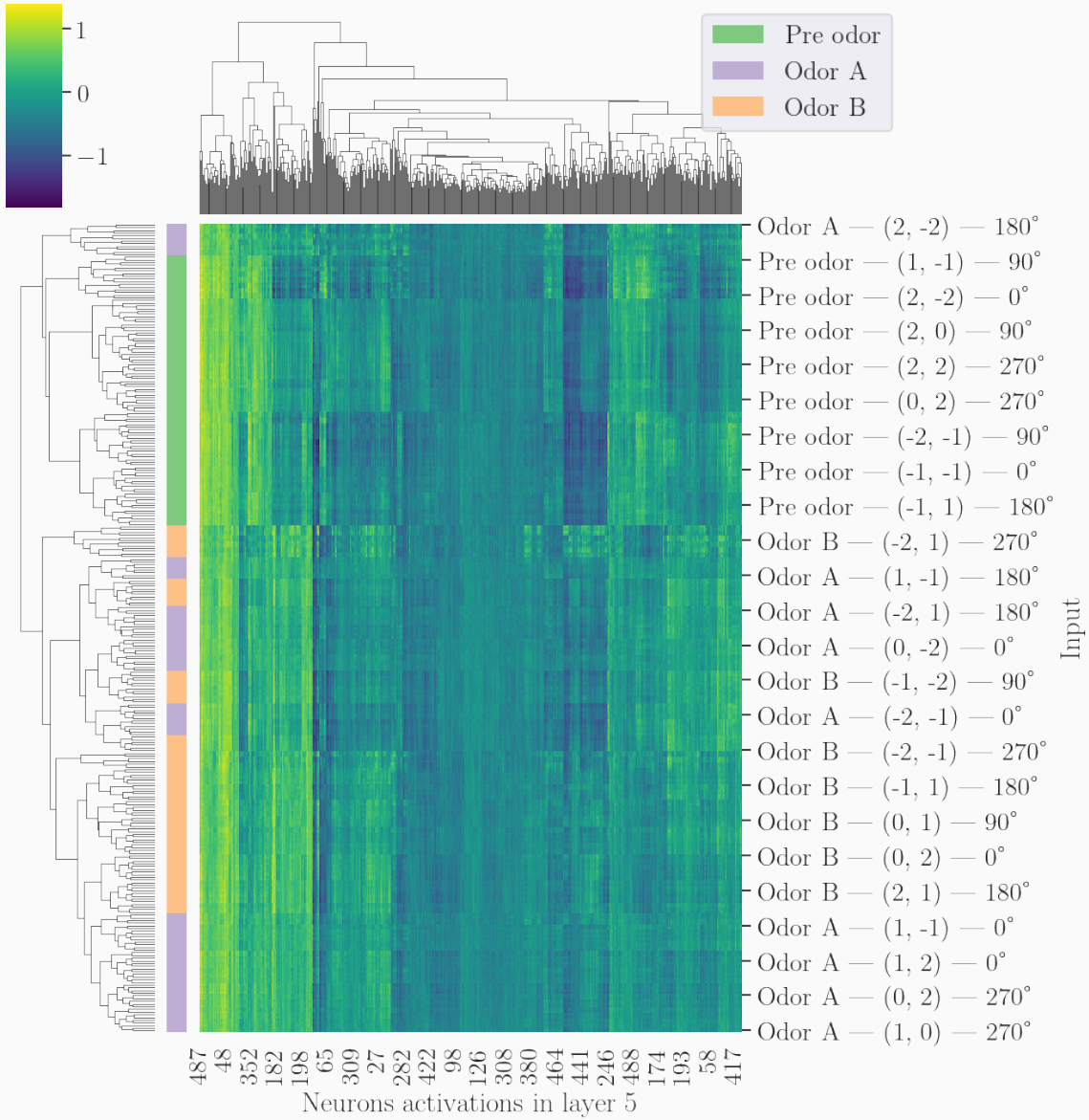
East/West



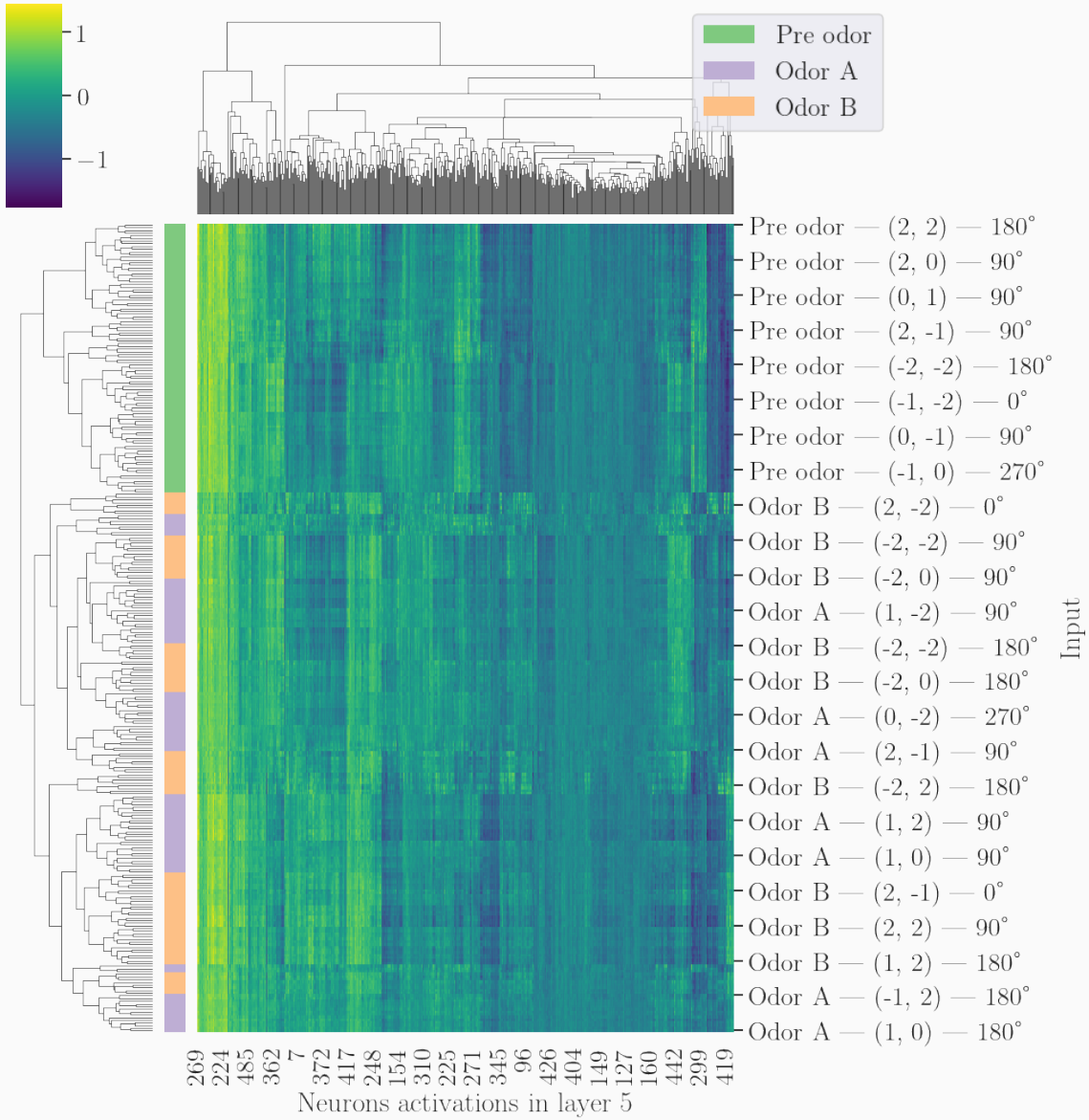
Left/Right



Activations learned – East/West



Activations learned – Left/Right



Use the behavior as proxy – Perturbation experiment

- Perturb the Cartesian/polar part of the input on a trained agent and look at how the agent behaves
- Expectation:
  - Left/right task:
  - East/west task:

Use the behavior as proxy – Perturbation experiment

- Perturb the Cartesian/polar part of the input on a trained agent and look at how the agent behaves
- Expectation:
  - Left/right task:
    - With the Cartesian inputs perturbed → agent's performance unchanged
    - With the polar inputs perturbed → agent's performance degrades
  - East/west task:
    - With the polar inputs perturbed → agent's performance unchanged
    - With the Cartesian inputs perturbed → agent's performance degrades

Use the behavior as proxy – Perturbation experiment

- Perturb the Cartesian/polar part of the input on a trained agent and look at how the agent behaves
- Expectation:
  - Left/right task:
    - With the Cartesian inputs perturbed → agent's performance unchanged
    - With the polar inputs perturbed → agent's performance degrades
  - East/west task:
    - With the polar inputs perturbed → agent's performance unchanged
    - With the Cartesian inputs perturbed → agent's performance degrades

Use the behavior as proxy – Perturbation experiment

- Perturb the Cartesian/polar part of the input on a trained agent and look at how the agent behaves
- Expectation:
  - Left/right task:
    - With the Cartesian inputs perturbed → agent's performance unchanged
    - With the polar inputs perturbed → agent's performance degrades
  - East/west task:
    - With the polar inputs perturbed → agent's performance unchanged
    - With the Cartesian inputs perturbed → agent's performance degrades

Use the behavior as proxy – Perturbation experiment

- Perturb the Cartesian/polar part of the input on a trained agent and look at how the agent behaves
- Expectation:
  - Left/right task:
    - With the Cartesian inputs perturbed → agent's performance unchanged
    - With the polar inputs perturbed → agent's performance degrades
  - East/west task:
    - With the polar inputs perturbed → agent's performance unchanged
    - With the Cartesian inputs perturbed → agent's performance degrades

Use the behavior as proxy – Perturbation experiment

- Perturb the Cartesian/polar part of the input on a trained agent and look at how the agent behaves
- Expectation:
  - Left/right task:
    - With the Cartesian inputs perturbed → agent's performance unchanged
    - With the polar inputs perturbed → agent's performance degrades
  - East/west task:
    - With the polar inputs perturbed → agent's performance unchanged
    - With the Cartesian inputs perturbed → agent's performance degrades

Use the behavior as proxy – Perturbation experiment

- Perturb the Cartesian/polar part of the input on a trained agent and look at how the agent behaves
- Expectation:
  - Left/right task:
    - With the Cartesian inputs perturbed → agent's performance unchanged
    - With the polar inputs perturbed → agent's performance degrades
  - East/west task:
    - With the polar inputs perturbed → agent's performance unchanged
    - With the Cartesian inputs perturbed → agent's performance degrades

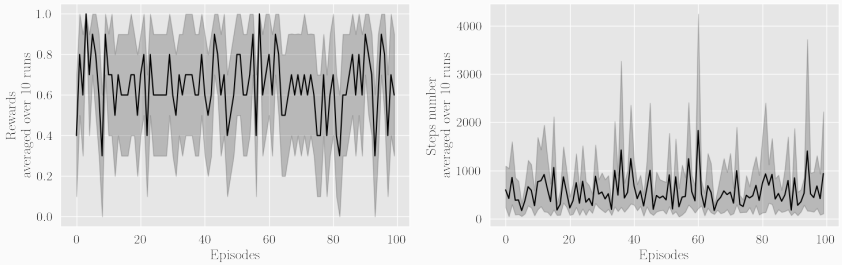
Use the behavior as proxy – Perturbation experiment

- Perturb the Cartesian/polar part of the input on a trained agent and look at how the agent behaves
- Expectation:
  - Left/right task:
    - With the Cartesian inputs perturbed $\rightarrow$ agent's performance unchanged
    - With the polar inputs perturbed $\rightarrow$ agent's performance degrades
  - East/west task:
    - With the polar inputs perturbed $\rightarrow$ agent's performance unchanged
    - With the Cartesian inputs perturbed $\rightarrow$ agent's performance degrades

Cartesian inputs unchanged – polar inputs perturbed

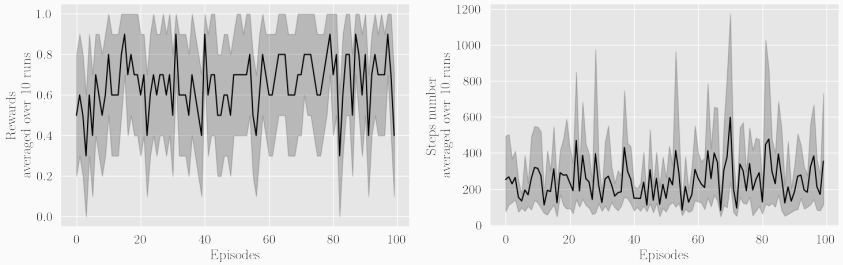
East/West

Silencing inputs

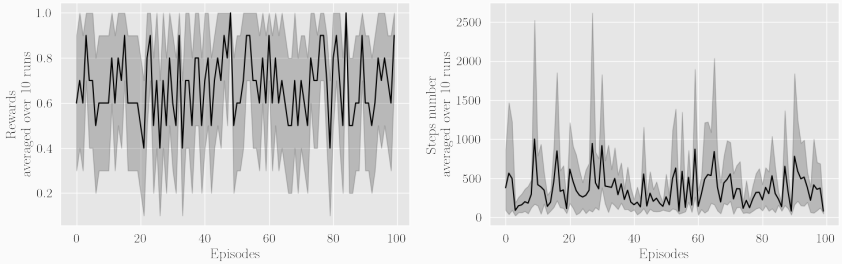


Left/Right

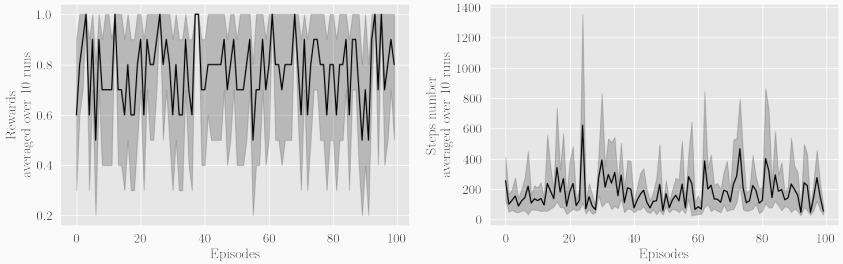
Silencing inputs



Randomizing inputs



Randomizing inputs



Polar inputs unchanged – Cartesian inputs perturbed

- Simulation does not end
  $\rightarrow$ couldn't figure out why yet...

Outline

Partial conclusions so far

- ∅ pattern on the weights
- The pre-odor activations cluster together, but no other clear pattern seems to emerge
- So far with this task setup, it seems both types of coordinates information are required to solve the task

Partial conclusions so far

- ∅ pattern on the weights
- The pre-odor activations cluster together, but no other clear pattern seems to emerge
- So far with this task setup, it seems both types of coordinates information are required to solve the task

Partial conclusions so far

- ∅ pattern on the weights
- The pre-odor activations cluster together, but no other clear pattern seems to emerge
- So far with this task setup, it seems both types of coordinates information are required to solve the task

Next steps

- Perturbation experiment:
  - Fix issue on Cartesian inputs
  - Setup more metrics for the study: performance histogram, % correct, etc.
- Need for some causal inference framework?
- Use of techniques from explainable AI?
- Study of the derivative of the output w.r.t. the inputs?
- Timeline: wrap the project by end of August

Next steps

- Perturbation experiment:
  - Fix issue on Cartesian inputs
  - Setup more metrics for the study: performance histogram, % correct, etc.
- Need for some causal inference framework?
- Use of techniques from explainable AI?
- Study of the derivative of the output w.r.t. the inputs?
- Timeline: wrap the project by end of August

Next steps

- Perturbation experiment:
  - Fix issue on Cartesian inputs
  - Setup more metrics for the study: performance histogram, % correct, etc.
- Need for some causal inference framework?
- Use of techniques from explainable AI?
- Study of the derivative of the output w.r.t. the inputs?
- Timeline: wrap the project by end of August

Next steps

- Perturbation experiment:
  - Fix issue on Cartesian inputs
  - Setup more metrics for the study: performance histogram, % correct, etc.
- Need for some causal inference framework?
- Use of techniques from explainable AI?
- Study of the derivative of the output w.r.t. the inputs?
- Timeline: wrap the project by end of August

Next steps

- Perturbation experiment:
  - Fix issue on Cartesian inputs
  - Setup more metrics for the study: performance histogram, % correct, etc.
- Need for some causal inference framework?
- Use of techniques from explainable AI?
- Study of the derivative of the output w.r.t. the inputs?
- Timeline: wrap the project by end of August

Next steps

- Perturbation experiment:
  - Fix issue on Cartesian inputs
  - Setup more metrics for the study: performance histogram, % correct, etc.
- Need for some causal inference framework?
- Use of techniques from explainable AI?
- Study of the derivative of the output w.r.t. the inputs?
- Timeline: wrap the project by end of August

Next steps

- Perturbation experiment:
  - Fix issue on Cartesian inputs
  - Setup more metrics for the study: performance histogram, % correct, etc.
- Need for some causal inference framework?
- Use of techniques from explainable AI?
- Study of the derivative of the output w.r.t. the inputs?
- Timeline: wrap the project by end of August

Questions ?