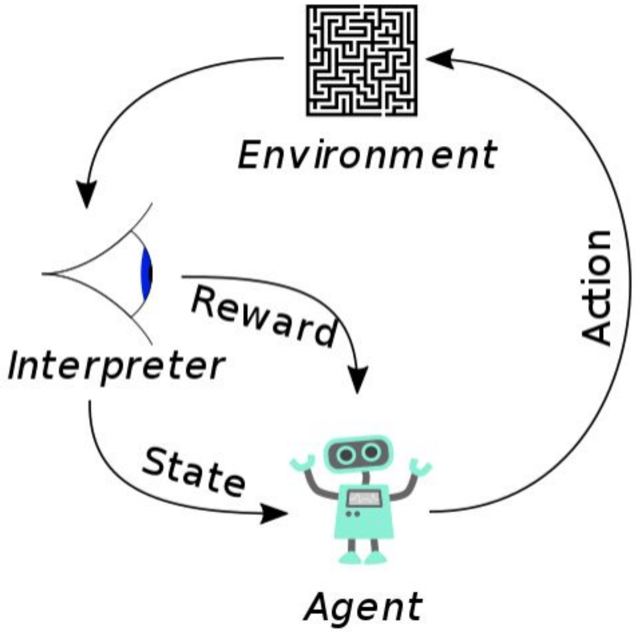
Human-level control through deep reinforcement learning

Siyabonga Matchaba

Northwestern

Terms

- State: Refers to the current environment, in the case of this paper it refers to the given pixel values at a specific time step t
- Actions: The set of moves an agent can make in a given environment
  - Ex. moving left or right in space invaders
- Policy: A way of acting/behaving at a given time
  - This is learned over the course of training
- Rewards: The goal that the agent is trying to achieve by taking a certain set of actions
  - Ex. High score, Obtaining desired object, Completing Level
- Agent: The thing that is capable of acting within an environment



Northwestern

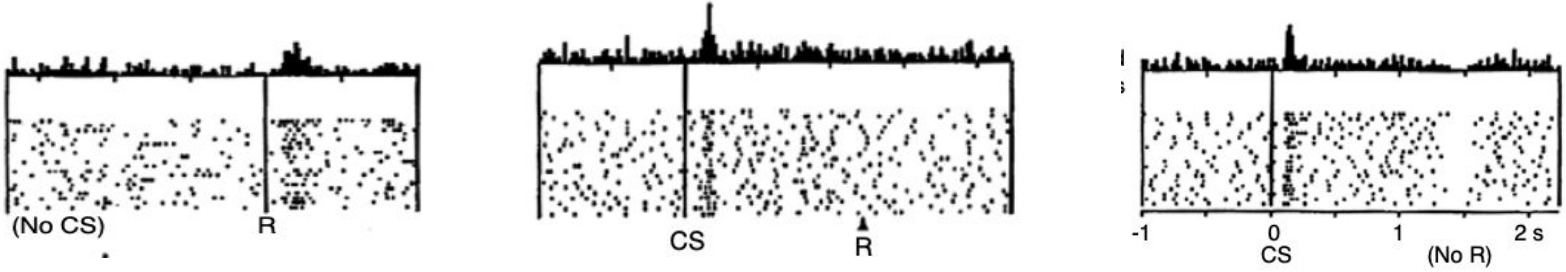
Background - Inspiration

- Primary inspiration derived from biological phenomena
    - Animals and humans ability to use reinforcement learning principles and sensor processing systems to learn new situations
- Idea: an agent(person/animal) learns by interacting with its environment and earns rewards by performing correctly and earns penalties by performing incorrectly
- Experiment uses neural data to highlight parallels between phase shifts in dopamine neurons and temporal difference reinforcement learning

Northwestern

** A Neural Substrate of Prediction and Reward - Wolfram Schultz, Peter Dayan, P. Read Montague**

Background - Inspiration

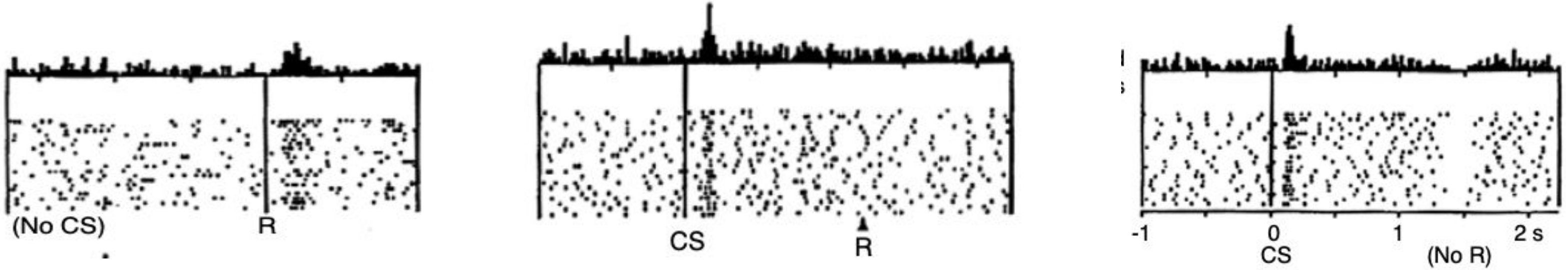


- CS: Condition reward predicting stimulus
- R: Reward
- Illustration of histogram formulated representations of impulses from the same neuron
- Horizontal Distance corresponds to real time intervals, and each line of dots indicates a trial
- High concentration of dopamine neuron activity close together represents an activation

Northwestern

** A Neural Substrate of Prediction and Reward - Wolfram Schultz, Peter Dayan, P. Read Montague**

Background - Inspiration



- Experiment: Monkey pulls lever when light shines and receives a reward
- Pre Learning: dopamine activated by unpredicted reward stimuli
- After Learning, conditioned stimulus predicts a reward
- Contrast: Dopamine neuron is activated by reward predicting stimuli instead of the predicted reward
- In case where reward is predicted by conditioned stimuli but actual rewards fails to occur because of certain behaviour, dopamine activity becomes ‘depressed’
- Monkey able to learn the value of the condition stimuli (the light)
- Rewards used to reinforce behaviour
- Monkey learns Behaviour ‘Policy’ → Pull lever

Northwestern

** A Neural Substrate of Prediction and Reward - Wolfram Schultz, Peter Dayan, P. Read Montague**

Problem/Solution

How do we use biological research and recent development in deep nets to enable representations of an environment from sensors to generalize past experience such that agents can act in new situations

- This paper draws from progress in image recognition to develop convolution based neural nets to learn environment sensory data
- Each state is made up of the pixel representations at each time step of the environment
- Observations, actions and rewards are all obtained from agents acting from state to state
- Goal of agent: Select actions in a way that optimizes the expected future reward

Problem/Solution

How do we use biological research and recent development in deep nets to enable representations of an environment from sensors to generalize past experience such that agents can act in new situations

** Optimal action-value function*

$$Q^*(s,a) = \max_{\pi} \mathbb{E}\left[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots \middle| s_t = s,\ a_t = a,\ \pi\right],$$

- Q value is the expected discounted reward for executing an action at state s by following a specific policy
- Maximum sum of rewards discounted at each time step
- Learned action determined by behaviour policy
- The value of a state is determined by agents expected reward given a set of action

Northwestern

Issues

- Correlations present in sequences of observations may restrict ability to learn generalized policy for highest expected reward
- Small updates to Q may lead to significant changes in the policy and therefore change the data distribution
- Correlations between action-values and the target values may limit agents ability to generalize its ability to learn for different environments

Northwestern

Improvements

Experience Replay → Agent is provided a memory of its past explored paths and rewards

- Random values chosen within experience replay to remove existing correlations in observation sequences
- Each experience is added to dataset D(t), to be randomly picked from in the future time steps

* E(t) → recording of state action reward and the next state at each time step

$$e_t = (s_t, a_t, r_t, s_{t+1})$$

Periodic update of Action Values → Action values are only adjusted periodically to reduce correlations with the targeted state, action

Northwestern

Deep Q learning Algorithm with experience replay

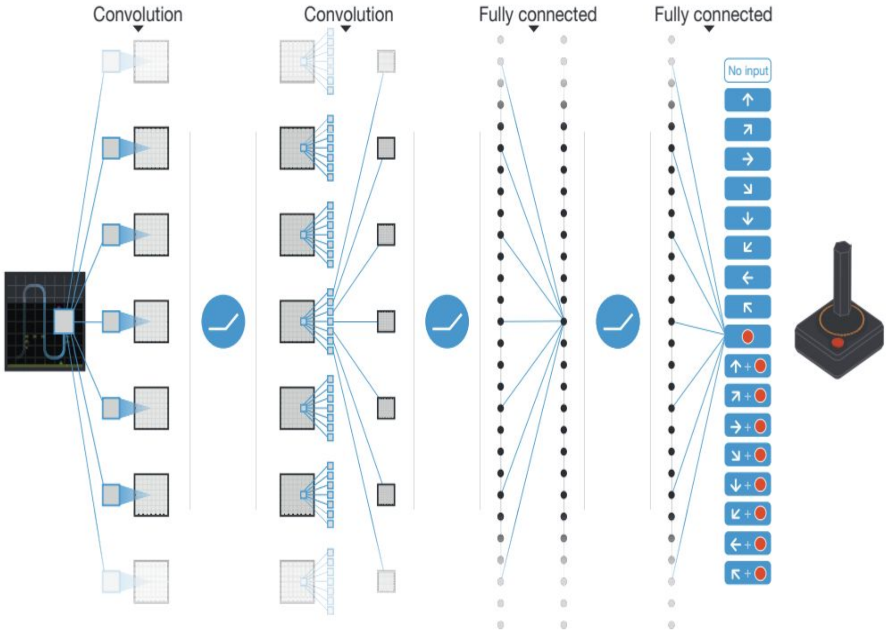
$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s') \sim \mathrm{U}(D)}\left[\left(r + \gamma \max_{a'} Q(s',a';\theta_i^-) - Q(s,a;\theta_i)\right)^2\right]$$

Algorithm 1: deep Q-learning with experience replay.
Initialize replay memory D to capacity N
Initialize action-value function Q with random weights θ
Initialize target action-value function $\hat{Q}$ with weights $\theta^- = \theta$
For episode = 1, M **do**
  Initialize sequence $s_1 = \{x_1\}$ and preprocessed sequence $\phi_1 = \phi(s_1)$
  For t = 1,T **do**
    With probability ε select a random action a_t
    otherwise select $a_t = \mathrm{argmax}_a Q(\phi(s_t), a; \theta)$
    Execute action a_t in emulator and observe reward r_t and image x_{t+1}
    Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$
    Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in D
    Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from D
    Set $y_j = \begin{cases} r_j & \text{if episode terminates at step j+1} \\ r_j + \gamma \max_{a'} \hat{Q}(\phi_{j+1}, a'; \theta^-) & \text{otherwise} \end{cases}$
    Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ with respect to the network parameters θ
    Every C steps reset $\hat{Q} = Q$
  End For
End For

- Loss function subtracts the Q value of the given state,action from the expected highest reward Q value term
- Theta represents the parameters of the network at the given iteration

Northwestern

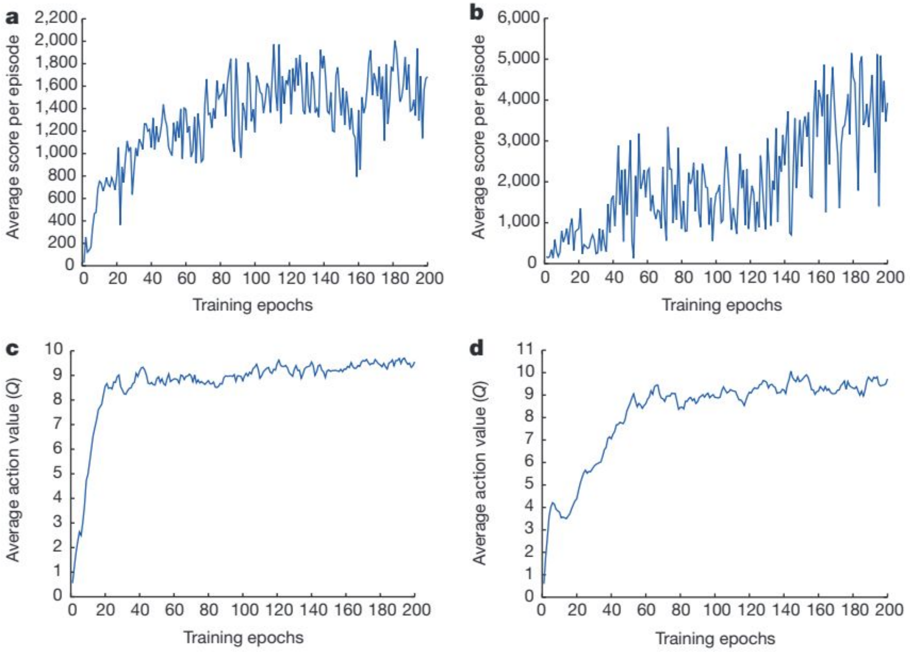
Model Structure

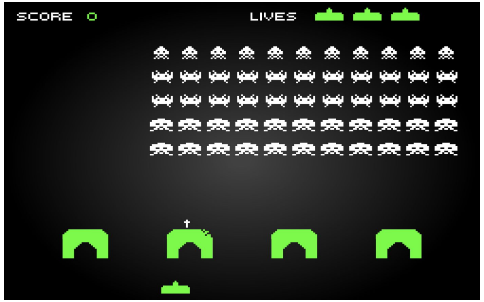


- Preproccessed input: 84x84x4 Image
    - Take max Value for each pixel color value to account for image flickering
    - Extract y channel (luminance/intensity from RGB frame and rescale it to 84x84
    - Stack m(most recent frames) to create input (m-4)
- 3 Convolution Layers
- 2 Fully connected layers
- Output: Valid Action

Northwestern

Evaluations



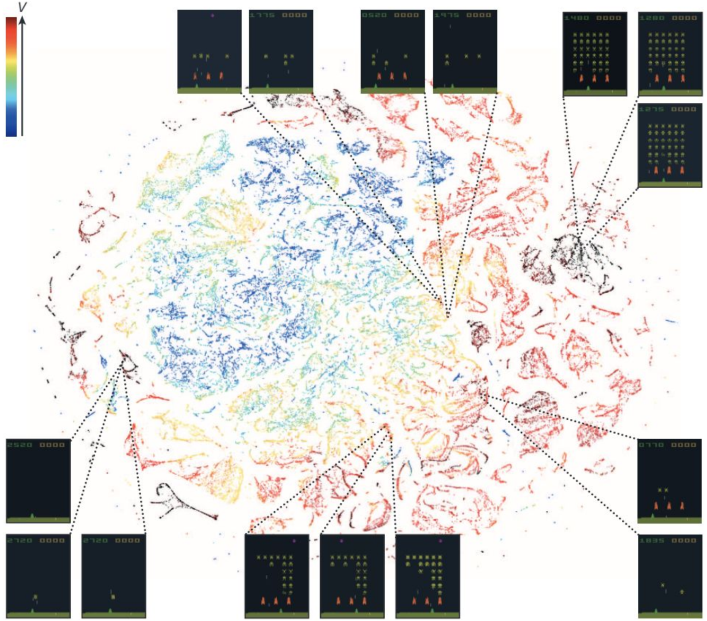


**Tested environments vary greatly, Ex. side scrolling shooters, boxing, 3d car racing games*

Evaluations

** Network is able to learn representations that support* ***adaptive behaviour***

Final Hidden Layer Representations



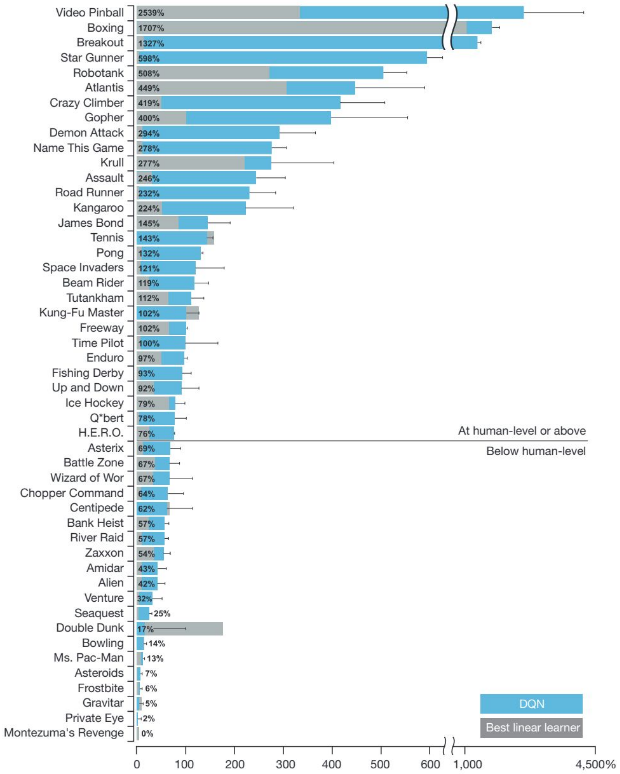
- Testing the effects of replay and target Q

Extended Data Table 3 | The effects of replay and separating the target Q-network

Game	With replay, with target Q	With replay, without target Q	Without replay, with target Q	Without replay, without target Q
Breakout	316.8	240.7	10.2	3.2
Enduro	1006.3	831.4	141.9	29.1
River Raid	7446.6	4102.8	2867.7	1453.0
Seaquest	2894.4	822.6	1003.0	275.8
Space Invaders	1088.9	826.3	373.2	302.0

Northwestern

Evaluations



** Performance : 100 x (DQN score - random play score)/(human score - random play score)*

Northwestern

Challenges

**Games demanding extensive planning strategies are challenging to learn*

Northwestern

Atari Breakout Example

https://www.youtube.com/watch?v=V1eYniJ0Rnk

Northwestern

Interesting finds

https://arxiv.org/pdf/1611.02167.pdf ⟶ *Neural Architecture Search using deep Q-learning*

https://www.cs.utexas.edu/~dana/Reward.pdf → *Paper on adaptive organisms and reinforcement learning*

http://www.gatsby.ucl.ac.uk/~dayan/papers/cjch.pdf → *Paper going into details about Q-learning*

https://arxiv.org/pdf/1511.06581.pdf → *Dueling deep Q-learning*

Northwestern

Time for Code!

Northwestern