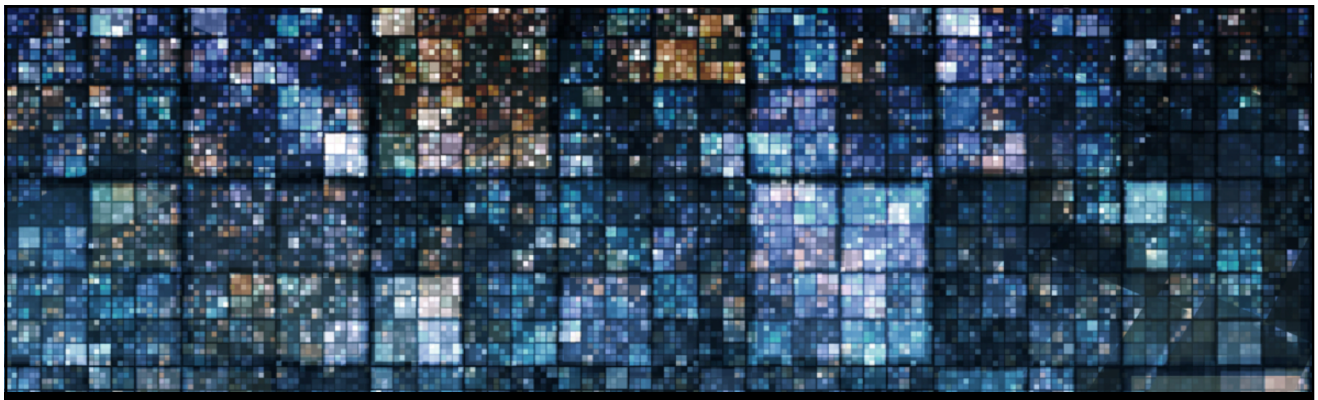
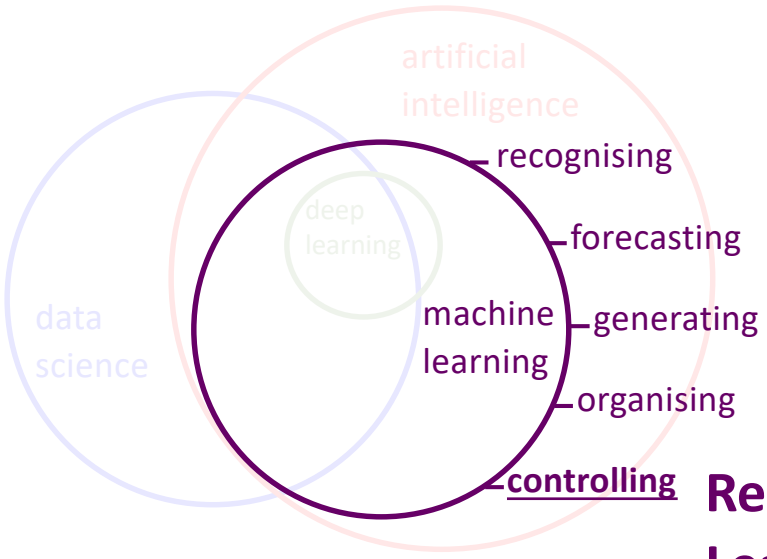


COMP47590
Advanced Machine Learning
RL - Deep Q Learning

Dr. Brian Mac Namee

Information

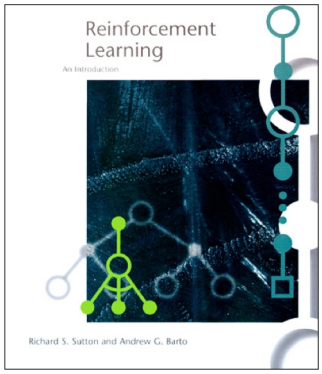
Email:	Brian.MacNamee@ucd.ie
Course Materials:	All material posted on UCD CS moodlehttps://csmoodle.ucd.ie/moodle/course/view.php?id=663 Enrolment key **UCDAdvML2017**



Reinforcement Learning

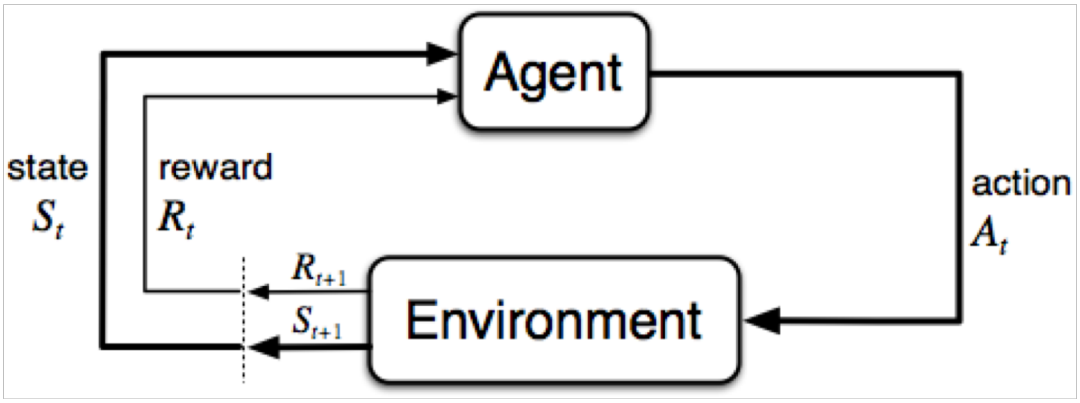
Reference Book

Much of the content in this section will follow "Reinforcement Learning An Introduction", 2nd Edition, Richard S. Sutton and Andrew G. Barto, MIT Press, 2018 (to appear)



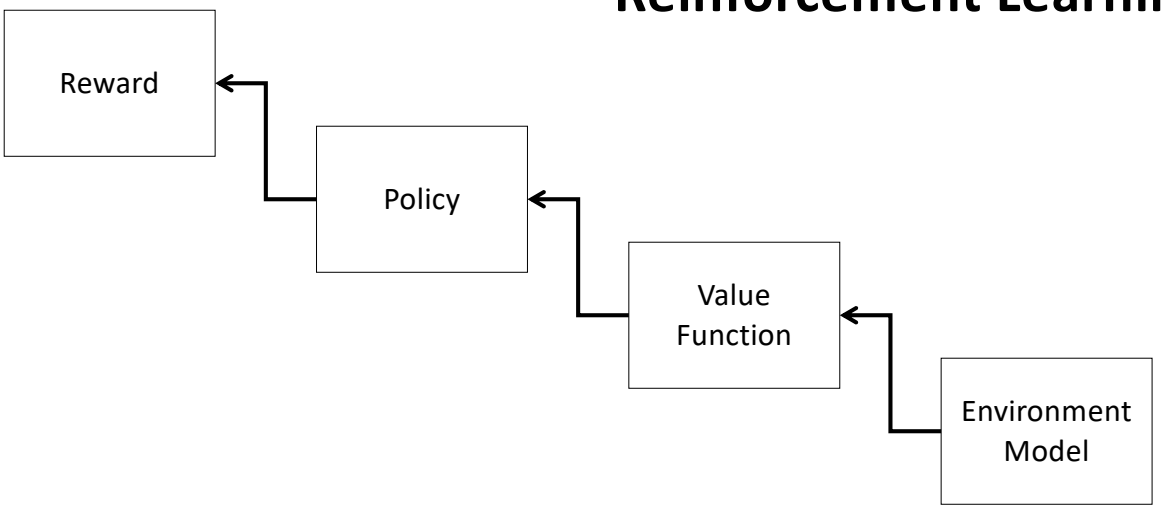
A legitimate online version of this book is available at:

www.incompleteideas.net/book/the-book-2nd.html



Reinforcement Learning: An Introduction , Second edition, in progress
Richard S. Sutton and Andrew G. Barto, MIT Press, 2017
www.incompleteideas.net/book/the-book-2nd.html

Reinforcement Learning



Sarsa: On-Policy TD Control

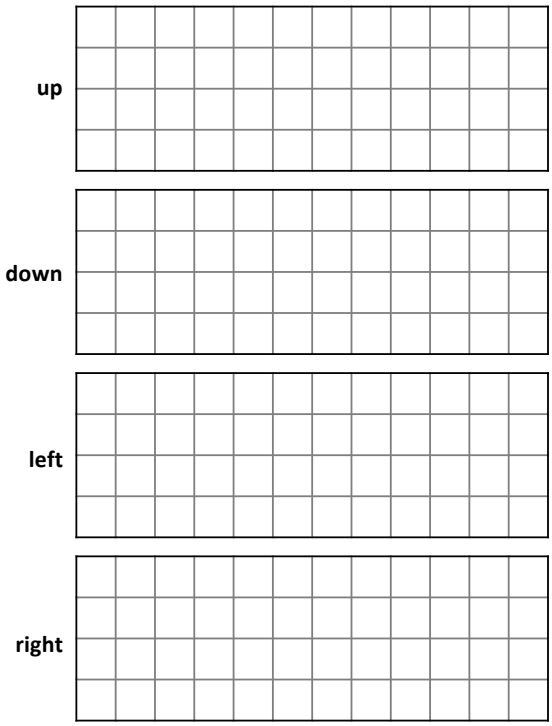
Initialize $Q(s,a), \forall s \in \mathcal{S}, a \in \mathcal{A}(s)$, arbitrarily, and $Q(\textit{terminal-state}, \cdot) = 0$
Repeat (for each episode):
  Initialize S
  Choose A from S using policy derived from Q (e.g., ε-greedy)
  Repeat (for each step of episode):
    Take action A, observe R, S'
    Choose A' from S' using policy derived from Q (e.g., ε-greedy)
    $Q(S,A) \leftarrow Q(S,A) + \alpha[R + \gamma Q(S',A') - Q(S,A)]$
    $S \leftarrow S'$; $A \leftarrow A'$;
  until S is terminal

Q-Learning: Off-Policy TD Control

Initialize $Q(s,a), \forall s \in \mathcal{S}, a \in \mathcal{A}(s)$, arbitrarily, and $Q(\textit{terminal-state}, \cdot) = 0$
Repeat (for each episode):
  Initialize S
  Repeat (for each step of episode):
    Choose A from S using policy derived from Q (e.g., ε-greedy)
    Take action A, observe R, S'
    $Q(S,A) \leftarrow Q(S,A) + \alpha[R + \gamma \max_a Q(S',a) - Q(S,A)]$
    $S \leftarrow S'$;
  until S is terminal

Tabular Methods

The methods we have looked at so far result in the construction of a table of value for the output of the action-value function for every state action pair



Tabular Methods

Learn to fill up the action-value function table by experimenting in the world

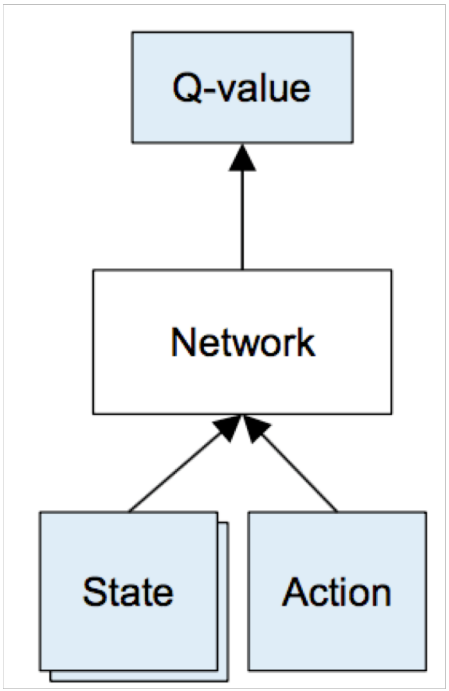
Tabular methods stop working well when the number of states present in a world grows - and they grow out of hand pretty quickly!

Think about how many states might exist in the LunarLander world?

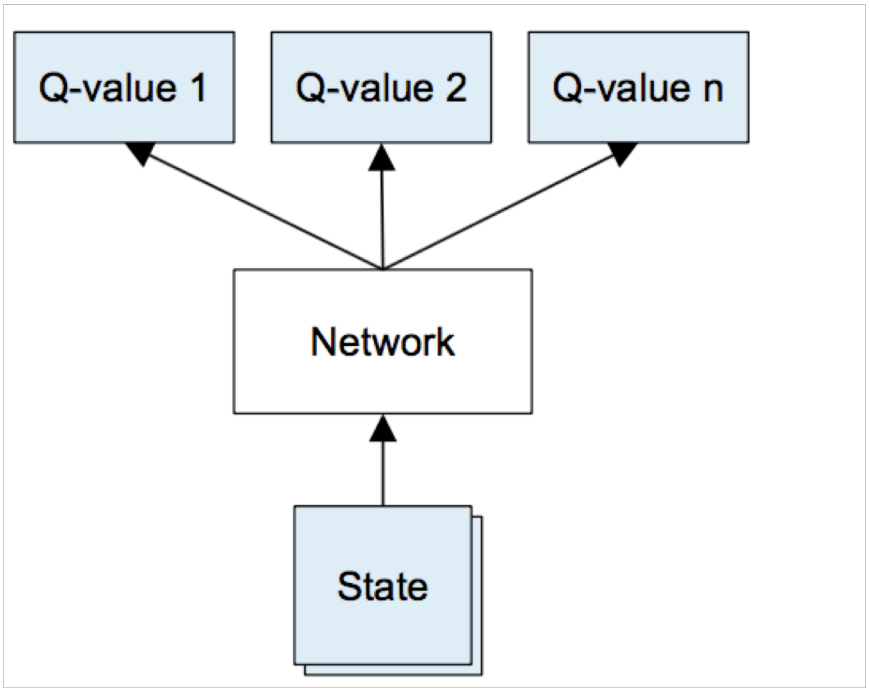
DEEP Q LEARNING

Deep Q Learning

Rather than using a lookup table for determining the values of state-action pairs use a non-linear deep neural network based model (referred to as the Q network)



Guest Post (Part I): Demystifying Deep Reinforcement Learning
https://ai.intel.com/demystifying-deep-reinforcement-learning/



Guest Post (Part I): Demystifying Deep Reinforcement Learning
https://ai.intel.com/demystifying-deep-reinforcement-learning/

Deep Q Learning: Loss & Gradient

This network is making the following assumption

$$Q(s, a, w) \approx Q^{\pi}(s, a)$$

where *w* is the set of weights from the network

To train a network we need two things:

- A loss function
- A gradient

Deep Q Learning: Loss & Gradient

We can use a a simple sum of squared errors loss function

- we assume the values associated with the best possible action plus the reward received is the target
- we assume the Q network's output for the state action pair is the prediction

$$\mathcal{L}(w) = \left(\underbrace{r + \gamma \max_{a'} Q(s', a', w)}_{\text{target}} - Q(s, a, w) \right)^2$$

Deep Q Learning: Loss & Gradient

The derivative of this loss function is as follows

$$\frac{\partial \mathcal{L}(w)}{\partial w} = \left(r + \gamma \max_{a'} Q(s', a', w) - Q(s, a, w) \right) \frac{\partial Q(s, a, w)}{\partial w}$$

Deep Reinforcement Learning, David Silver
http://videolectures.net/rldm2015_silver_reinforcement_learning/

Deep Q Learning (Naive)

Use back-propagation and mini-batches stochastic gradient descent to update the network

- For each mini-batch
  - For each instance
    - Do a forward pass for the current state to get all possible Q values
    - Do a forward pass on the new state and find the action with the biggest Q value
    - Set Q-value target for action to r + γmax a' Q(s', a')
      - For all other actions, set the Q-value target to the same as originally returned from step 1, making the error 0 for those outputs
    - Update loss function
  - Update network weights using gradient equation

Deep Q Learning

Naive Q-learning oscillates or diverges

- Data is sequential
  - Successive samples are correlated
- Policy changes rapidly with slight changes to Q-values
  - Policy may oscillate
  - Distribution of data can swing from one extreme to another
- Scale of rewards and Q-values is unknown
  - Naive Q-learning gradients can be large unstable when back-propagated

Deep Q Learning

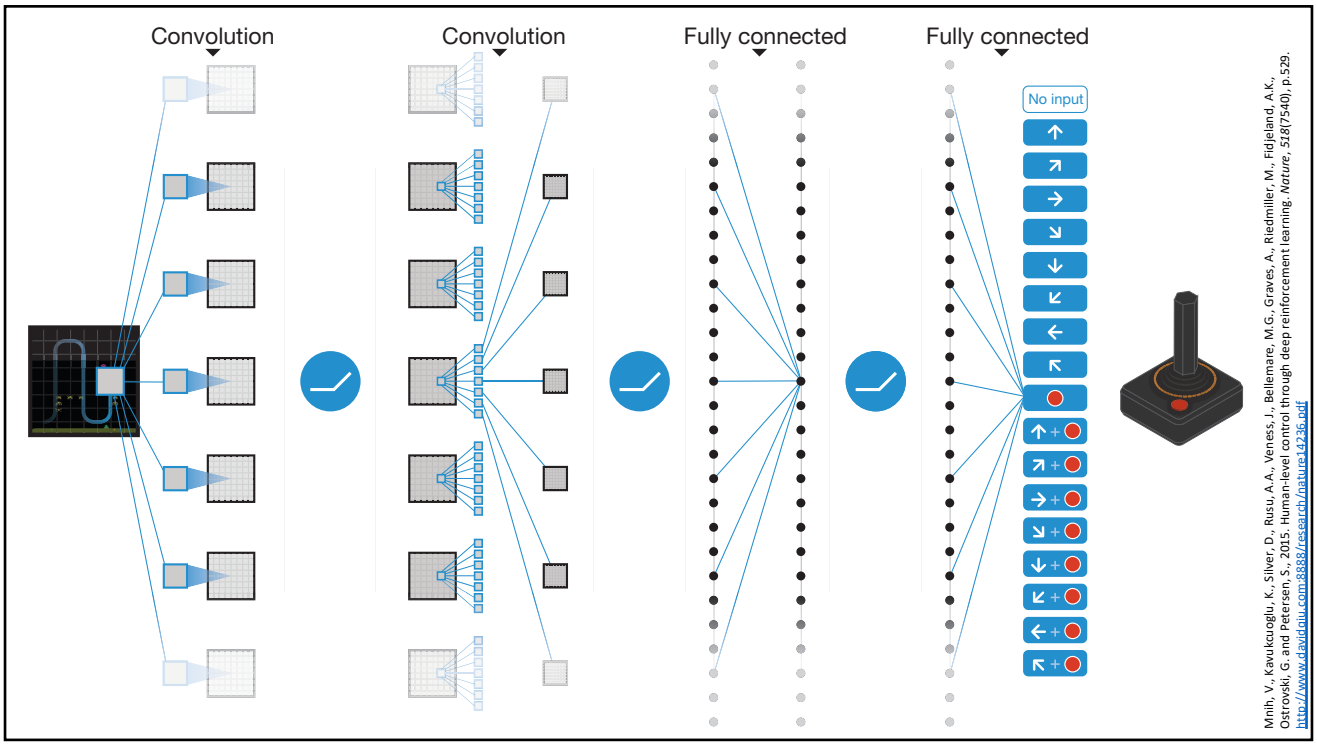
DQN provides a stable solution to deep RL

- Use experience replay
  - Break correlations in data
  - Learn from all past policies
  - Using off-policy Q-learning
- Freeze target Q-network
  - Avoid oscillations
  - Break correlations between Q-network and target
- Clip rewards or normalize network adaptively to sensible range
  - Robust gradients

FAMOUS CASE STUDY ATARI GAME PLAYING

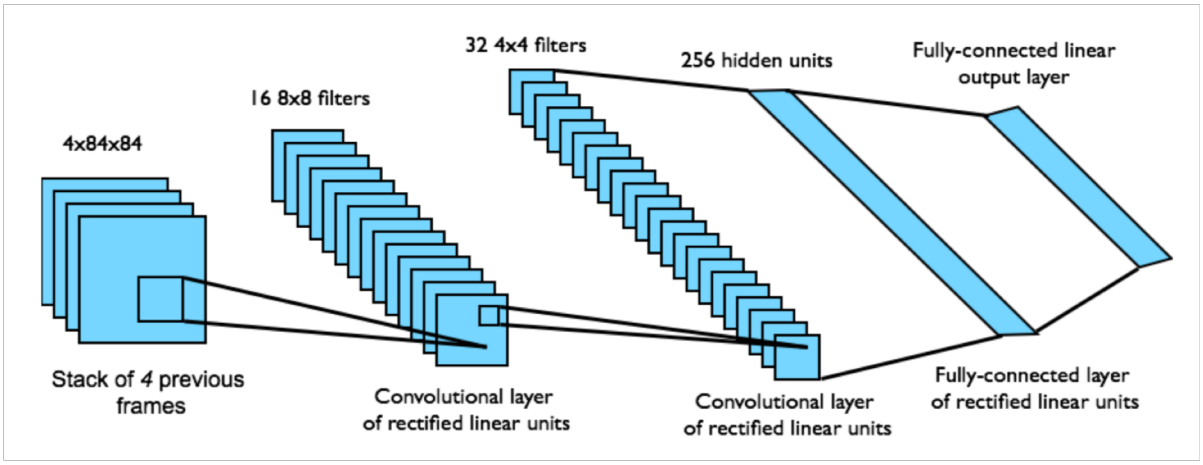
Atari Game Playing

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



Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G. and Petersen, S., 2015. Human-level control through deep reinforcement learning. *Nature*, *518*(7540), p.529. http://www.davidqiu.com:8888/research/nature14236.pdf

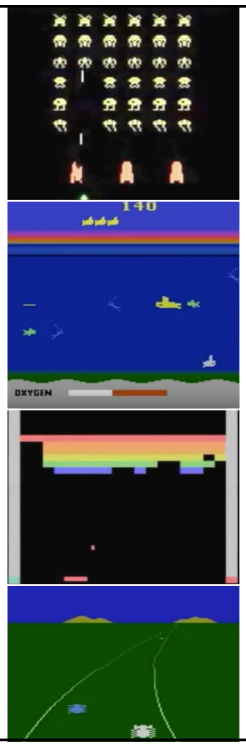
Atari Game Playing



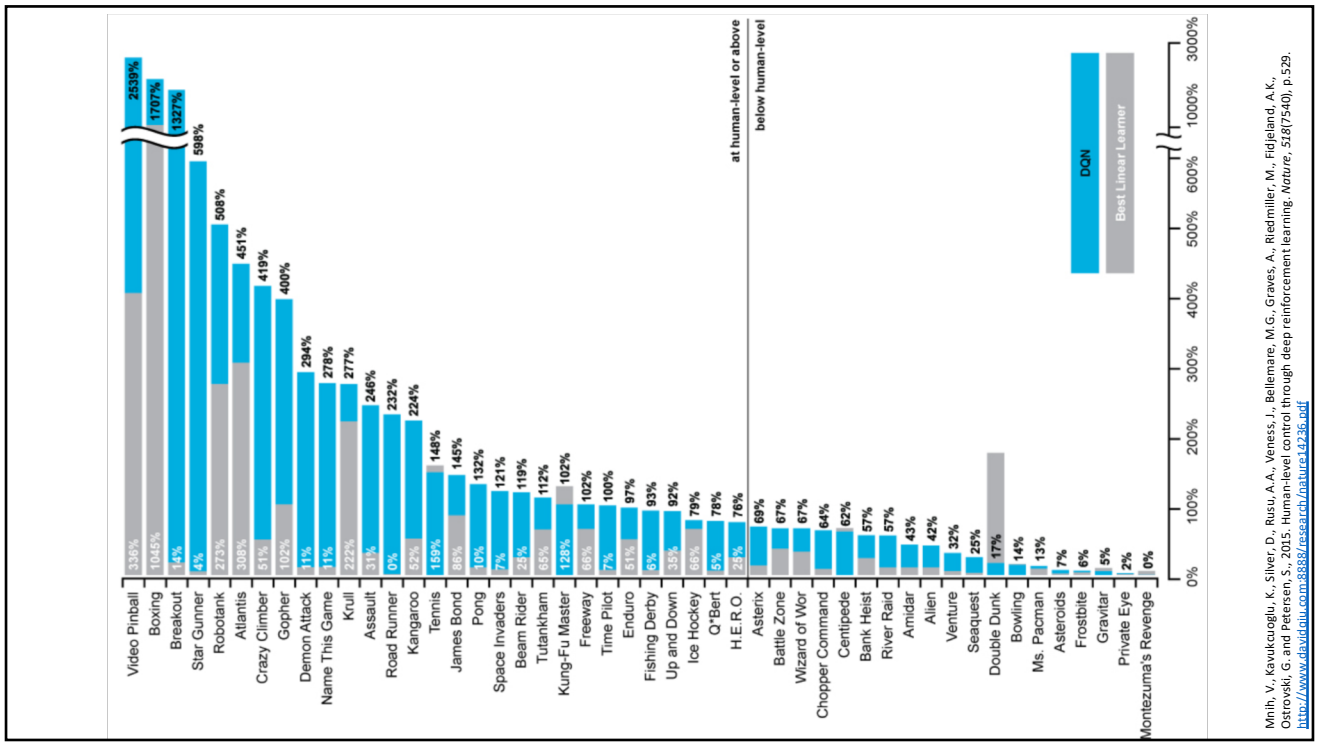
Deep Reinforcement Learning, David Silver
http://videolectures.net/rldm2015_silver_reinforcement_learning/

Atari Game Playing

- Learned to play 49 games for the Atari 2600 game console, without labels or human input, from self-play and the score alone
- Learned to play better than all previous algorithms and at human level for more than half the games
- Same learning algorithm applied to all 49 games w/o human tuning!



Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G. and Petersen, S., 2015. Human-level control through deep reinforcement learning. *Nature*, *518*(7540), p.529.
http://www.davidqiu.com:8888/research/nature14236.pdf



Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G. and Petersen, S., 2015. Human-level control through deep reinforcement learning. *Nature*, *518*(7540), p.529. http://www.davidqiu.com:8888/research/nature14236.pdf

SUMMARY

Summary

Deep Q learning takes the Q learning reinforcement learning ideas and merges them with the representation learning of deep neural networks

Questions