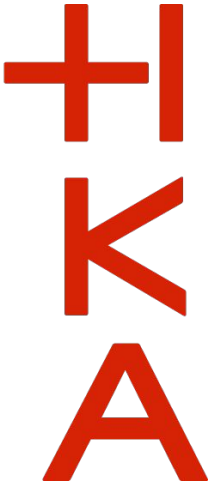
inovex



KI Labor - Wintersemester 2021

Reinforcement **L**earning
Sprintwechsel & Vorstellung Assignment

Darjan Salaj, Sven Müller, Maximilian Blanck,
Sebastian Blank, Frederik Martin, Pascal Fecht

Karlsruhe, 14. Jan. 2022

Schedule

Datum	Thema	Inhalt	Präsenz
01.10.21	Allg.	Organisation, Teamfindung	Nein
08.10.21	CV	Vorstellung CV	Nein
15.10.21	CV	Q&A Sessions	Nein
22.10.21	CV	Sprintwechsel, Vorstellung Assignment	Ja
29.10.21	CV	Q&A Sessions	Nein
05.11.21	CV / NLP	Abgabe CV, Vorstellung NLP	Ja
12.11.21	NLP	Q&A Sessions	Nein
19.11.21	NLP	Sprintwechsel, Vorstellung Assignment	Ja
26.11.21	NLP	Q&A Sessions	Nein
~~*03.12.21*~~	~~*NLP*~~	*Keine Veranstaltung* ~~*Q&A Sessions*~~	~~*Nein*~~
10.12.21	NLP / RL	Abgabe NLP, Vorstellung RL	Nein ~~*Ja*~~
17.12.21	RL	Q&A Sessions	Nein
14.01.22	**RL**	**Sprintwechsel, Vorstellung Assignment**	**Nein** ~~***Ja***~~
21.01.22	RL	Q&A Sessions	Nein
28.01.22	RL	Abgabe RL, Abschluss KI Labor	**Nein** ~~***Ja***~~ **(?)**

inovex

Agenda

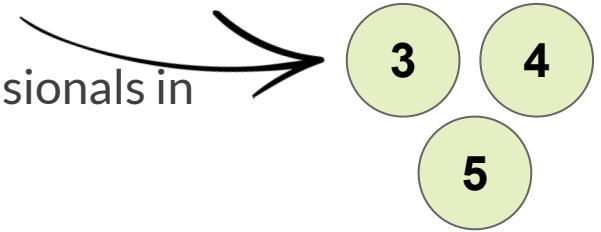
- **Theorie**
  - Deep Q-Network
  - Experience Replay
  - Target Model
  - Vorverarbeitung für Pixel-basierte Atari Games (Framestacking, etc.)

- **Praxis**
  - CartPole Gym mit Deep Q-Learning (Aufgabe 3)
  - Pong (Pixel-basiert) mit Deep Q-Learning (Aufgabe 4)
  - Space Invaders (Assignment)

Reinforcement Learning

Meilensteine im Reinforcement Learning

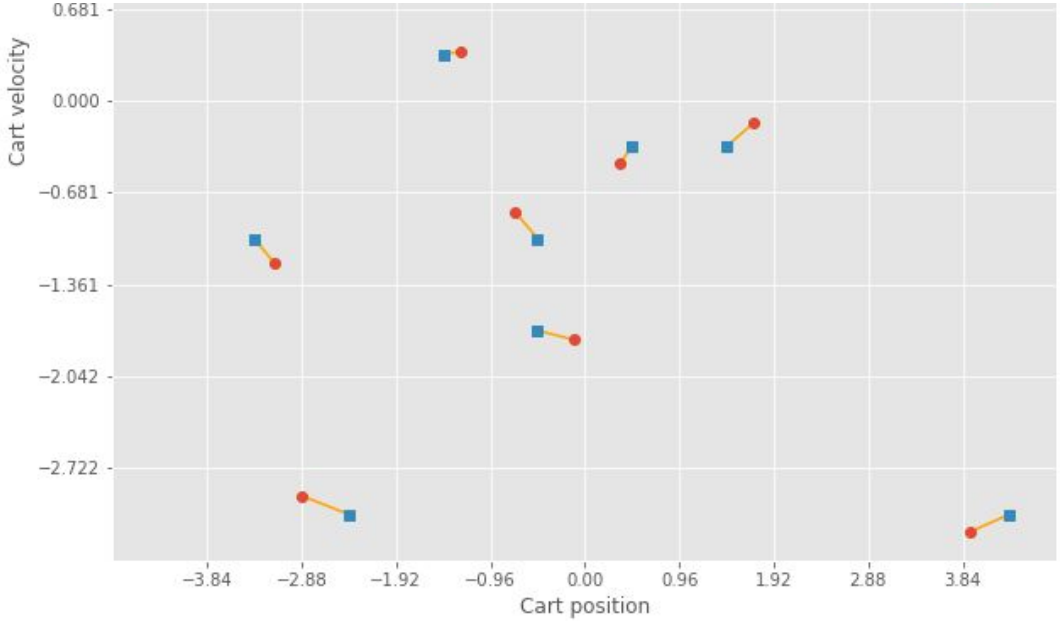
1950’s	Optimal control, dynamic programming
1970-80’s	Temporal difference learning
1989	Q-Learning
1992	TD Gammon
2013	Playing atari with deep reinforcement learning
2016	AlphaGo vs. Lee Sedol
2019	AlphaStar & OpenAI Five beat professionals in Starcraft and Dota 2



inovex

Was bisher geschah ...

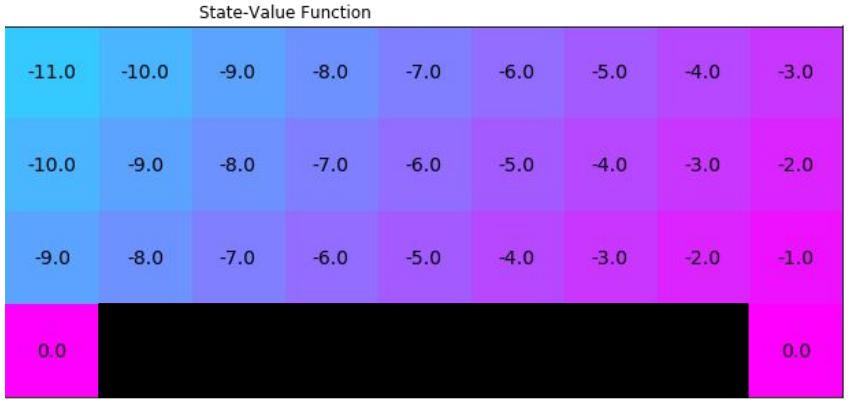
Diskretisierung

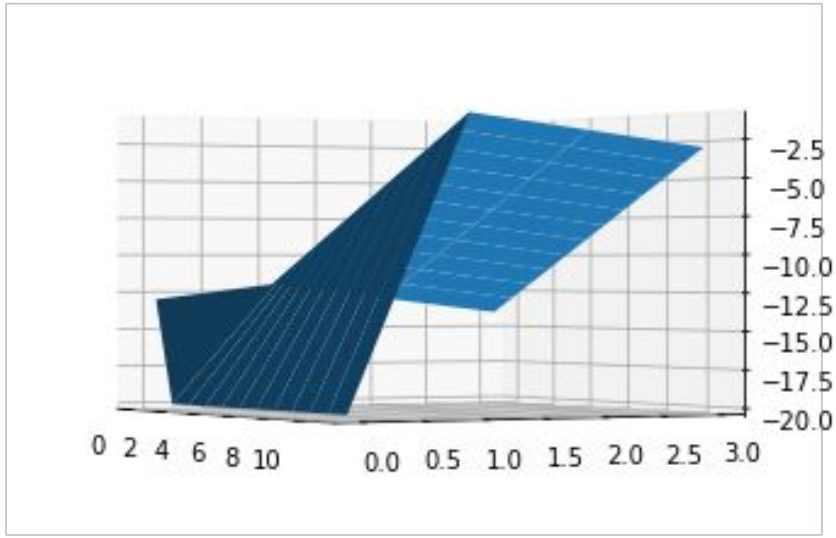


inovex

Funktionsapproximation

Beispiel Cliffwalking





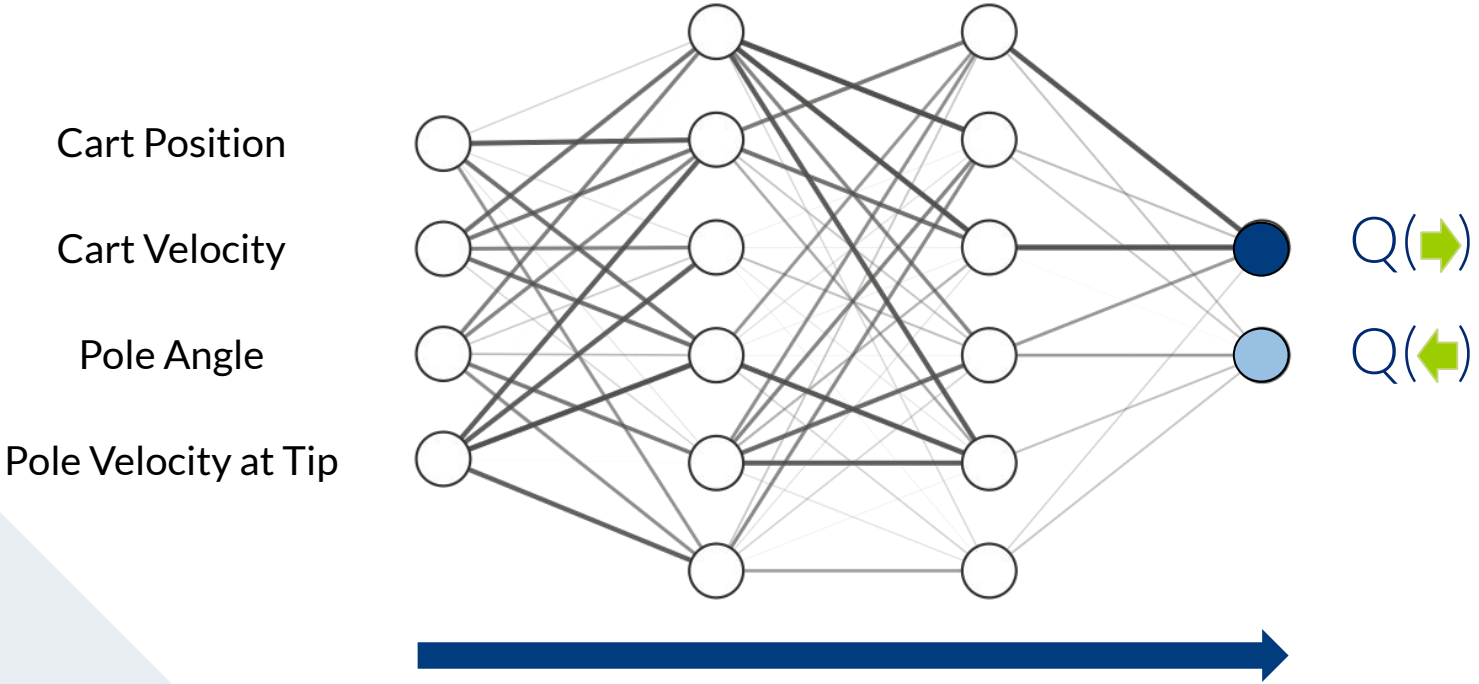
$$\hat{v}(s,\theta) \approx v_\pi(s), \theta \in \mathbb{R}^d$$

Approximation — Value-Function — Gewichtsvektor

inovex

Funktionsapproximation

Beispiel CartPole: **Forward-Pass**



inovex

Funktionsapproximation

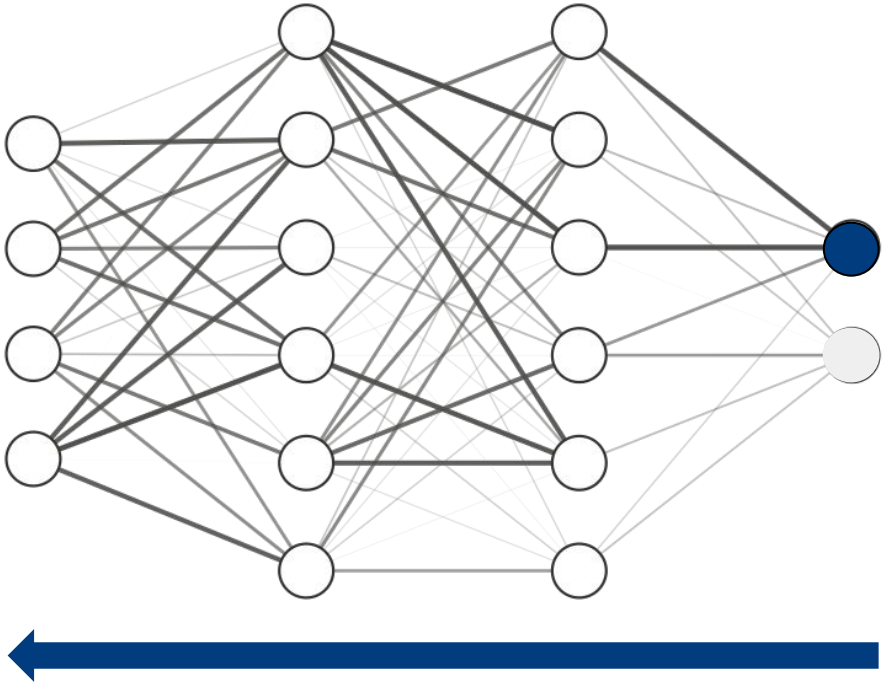
Beispiel CartPole: **Backward-Pass**

Cart Position

Cart Velocity

Pole Angle

Pole Velocity at Tip



Action: right selected

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

inovex

Funktionsapproximation

Deep Q-Network

- Approximation der Q-Funktion mit NN
  - Optimierung mit SGD
  - Minimierung des Abstands zwischen Schätzer und Target

$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s')\sim U(D)}[(y_i - \hat{q}(s, a, \theta_i))^2]$$

Loss L_i

Weights θ_i **in Iteration i**

Experience Replay $(s,a,r,s')\sim U(D)$

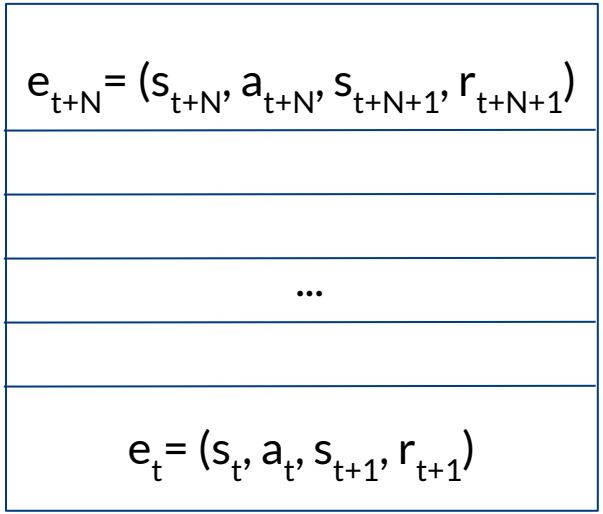
Target Q-Value y_i

approximierter Q-Value $\hat{q}(s, a, \theta_i)$

Mnih et al. (2015) - Human-level Control through Deep Reinforcement Learning

Experience Replay

- Problem
  - Starke Korrelation der States erschwert das Lernen
- Lösung
  - Letzte *N* Experiences werden in **Replay Memory** gespeichert
  - Random Uniform Sampling



Replay Memory

Target Network

- Kopie der eigentlichen Architektur mit fixen Gewichten
- Update alle c Iterationen

$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s')\sim U(D)}[(y_i - \hat{q}(s, a, \theta_i))^2]$$

Q-Network

$$r + \gamma \max_{a'} Q(s', a', \theta_i^-)$$

Target Network

Mnih et al. (2015) - Human-level Control through Deep Reinforcement Learning

Aufgaben

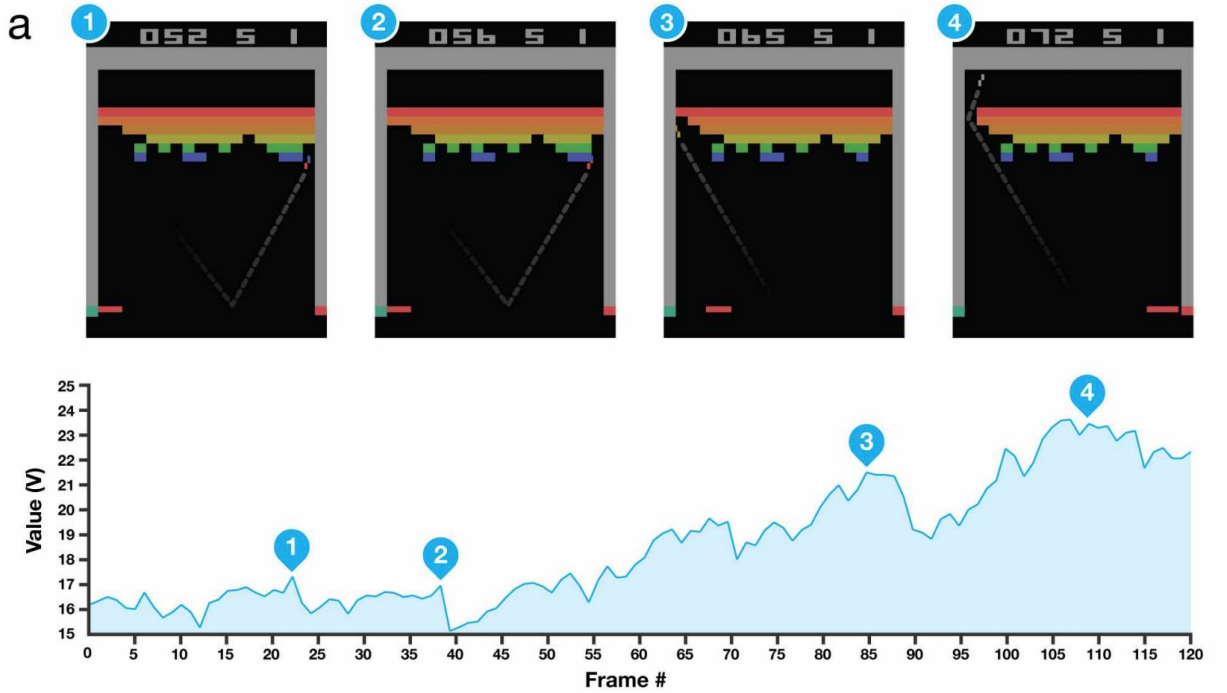
- Aufgabe 3: CartPole Gym mit Deep Q-Learning (freiwillig)
- Aufgabe 4: Pong mit Deep Q-Learning (freiwillig)
- Aufgabe 5: Space Invaders (Assignment = Bewertungsgrundlage)

Aufgabe 3: CartPole Gym mit Deep Q-Learning

- **Freiwillige Bearbeitung** als Vorbereitung auf Assignment
- Jupyter Lab Notebook

Deep Q-Network

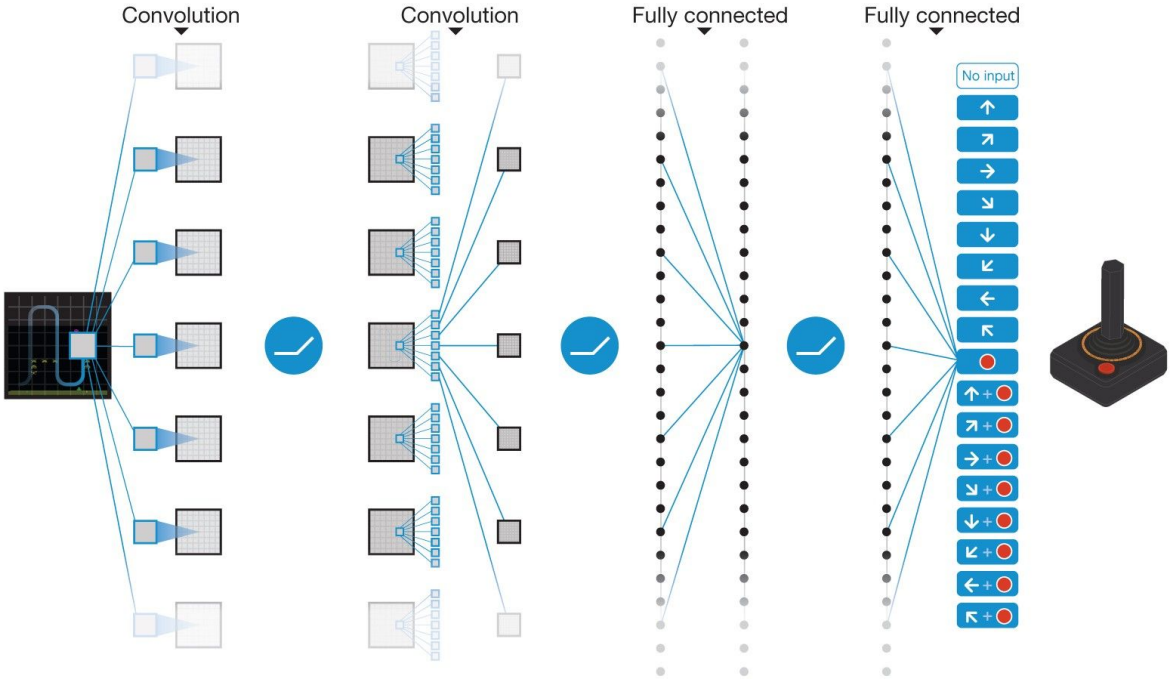
Pixel-basierte Atari Games



Mnih et al. (2015) - Human-level Control through Deep Reinforcement Learning

Deep Q-Network

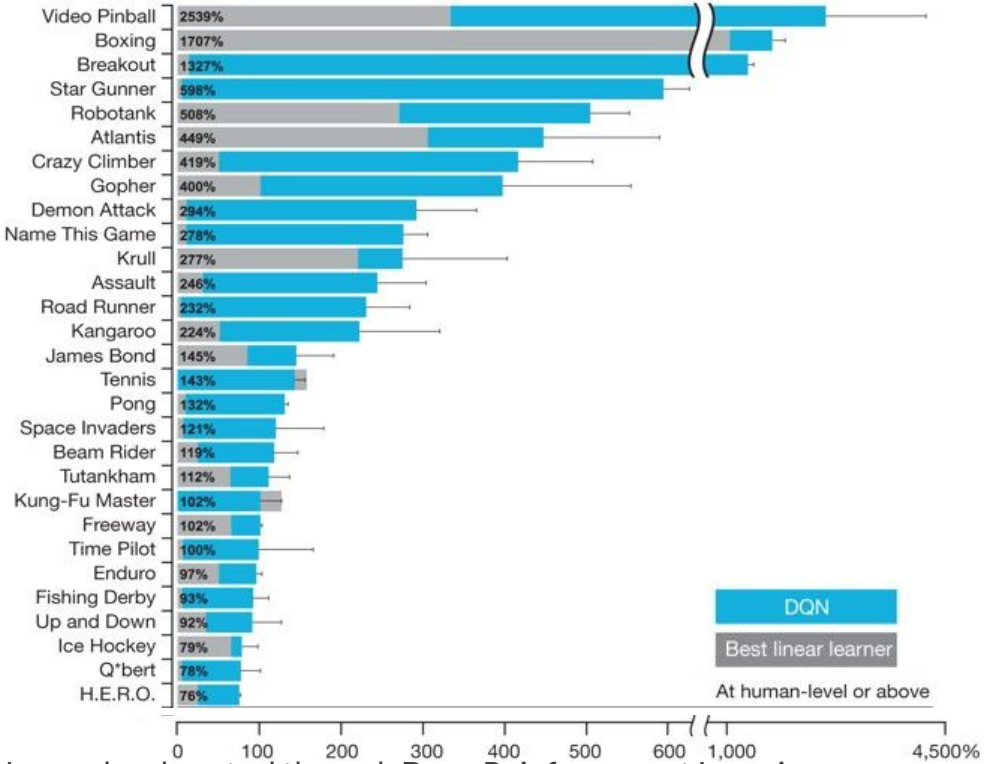
Pixel-basierte Atari Games



Mnih et al. (2015) - Human-level control through Deep Reinforcement Learning

Deep Q-Network

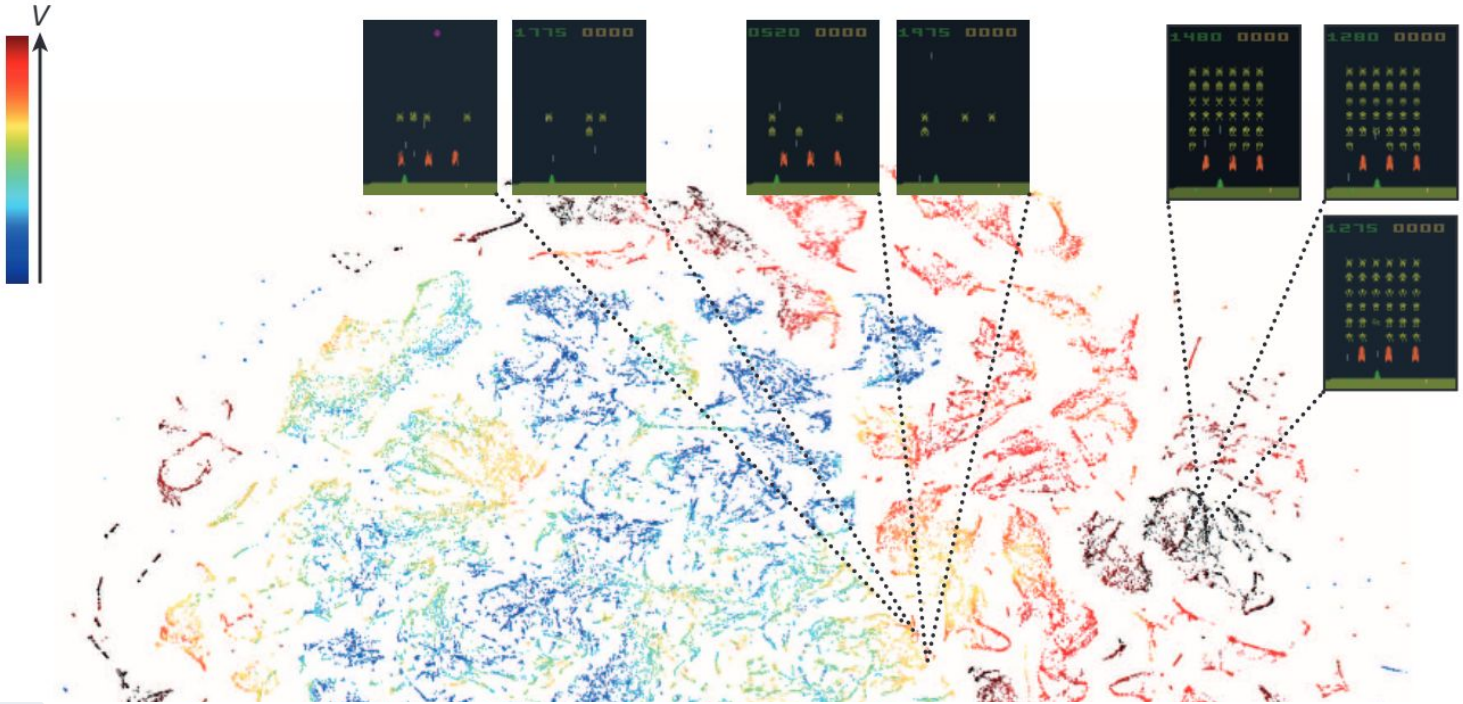
Pixel-basierte Atari Games



Mnih et al. (2015) - Human-level control through Deep Reinforcement Learning

inovex

Deep Q-Network



Mnih et al. (2015) - Human-level control through Deep Reinforcement Learning

Deep Q-Network

Preprocessing Pixel-basierter Inputs

- **Warp Frame**
  - Konvertierung der Frames in Graufstufen
  - Downsampling auf 84x84 Pixel

- **Framestack**
  - Kombination von vier aufeinanderfolgenden Frames um Bewegungen nachvollziehen zu können
  - Auswahl der maximalen Helligkeitswerten
  - Ausführung der gewählten Aktion für Framestack

Deep Q-Network

Besonderheiten Atari

- **No-Ops after Reset**
  - Skippen der ersten 30 Steps
- **Fire Reset**
  - Automatisches Drücken der FIRE-Taste als erste Aktion
- **Episodic Life**
  - Verlust eines Lebens bedeutet das Ende der Episode
  - Game Over setzt die Environment komplett zurück
- **Reward Clipping**
  - pos. Reward: +1 neg. Reward: -1 sonst: 0

Deep Q-Learning

Algorithmus

Algorithm 1 Deep Q-learning with Experience Replay

Initialize replay memory $\mathcal{D}$ to capacity N
Initialize action-value function Q with random weights
for episode $= 1, M$ **do**
  Initialise sequence $s_1 = \{x_1\}$ and preprocessed sequenced $\phi_1 = \phi(s_1)$
  for $t = 1, T$ **do**
    With probability ϵ select a random action a_t
    otherwise select $a_t = \max_a Q^*(\phi(s_t), a; \theta)$
    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 $\mathcal{D}$
    Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from $\mathcal{D}$
    Set $y_j = \begin{cases} r_j & \text{for terminal } \phi_{j+1} \\ r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta) & \text{for non-terminal } \phi_{j+1} \end{cases}$
    Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ according to equation 3
  end for
end for

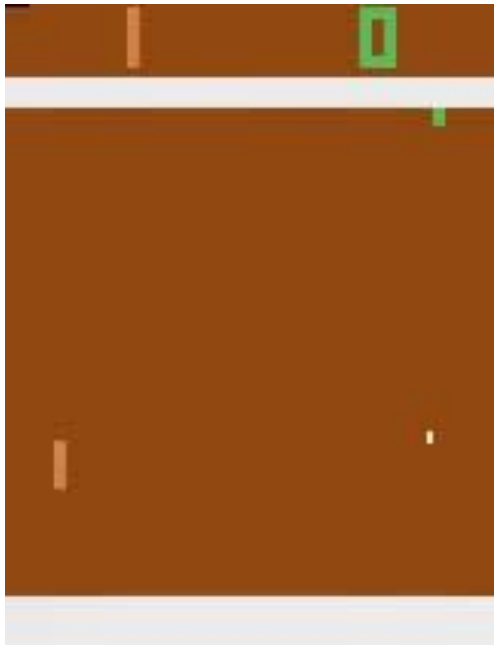
Erstes Paper (2013) ohne Target Network

$$\nabla_{\theta_i} L_i(\theta_i) = \mathbb{E}_{s,a \sim \rho(\cdot); s' \sim \mathcal{E}} \left[\left(r + \gamma \max_{a'} Q(s', a'; \theta_{i-1}) - Q(s, a; \theta_i) \right) \nabla_{\theta_i} Q(s, a; \theta_i) \right]. \quad (3)$$

Mnih et al. (2013) - Playing Atari with Deep Reinforcement Learning

Aufgabe 4: Pong mit Deep Q-Learning

- **Freiwillige Bearbeitung** als Vorbereitung auf Assignment
- Jupyter Lab Notebook



Aufgabe 5: Space Invaders Assignment

- **Assignment dient als Bewertungsgrundlage = Pflicht**
- Jupyter Lab Notebook
- Freie Wahl des Ansatzes

Literatur

- Kostenlose "Standard"-Lektüre für den Einstieg in RL:
  Reinforcement Learning: An Introduction (Sutton and Barto), siehe
  http://incompleteideas.net/book/RLbook2018.pdf
- Ausführlich und gut erklärter Einstieg in RL (Video-Lektionen):
  UCL Course on RL (David Silver, Google DeepMind), siehe
  https://www.davidsilver.uk/teaching/
- *Algorithms in Reinforcement Learning* von Csaba Szepesvári, siehe
  https://sites.ualberta.ca/~szepesva/papers/RLAlgsInMDPs.pdf
- Blog mit Videos zum Einstieg in RL und Q-Learning, DQN und vieles mehr:
  Reinforcement Learning – Introducing Goal Oriented Intelligence, siehe
  https://deeplizard.com/learn/video/nyjbcRQ-uQ8

Feedback

https://forms.gle/9gsVAVBRoBqP4brL6

Vielen Dank

Frederik Martin
fmartin@inovex.de

Sebastian Blank
sblank@inovex.de