Deep Reinforcement Learning

Seminar, Summer Term 2019

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

Fraunhofer
IIS

Contents

1. Who we are - Intro
2. What is (Deep) Reinforcement Learning
3. Goals of this seminar
4. Prerequisites
5. Topics
6. When to hold the presentations and other schedules

Who we are

Chris M.

Chris L.

Georgios

Introduction
Fraunhofer Institute for Integrated Circuits IIS

Erlangen/Tennenlohe

Nürnberg/Nordostpark

Introduction
Fraunhofer Institute for Integrated Circuits IIS

- Founding year: 1985
- Employees: about 1050
- Budget about. 180 Mio. Euro (2017)
- Financing: 74% internal financing by projects, 26% funded by state
- 11 Locations: Erlangen (headquarters), Nürnberg, Fürth, Dresden, Bamberg, Waischenfeld, Coburg, Würzburg, Ilmenau, Deggendorf and Passau
- Extensive networking with Universities

Reinforcement Learning

- Playing Games with RL

https://www.youtube.com/watch?v=lc1fl5bdZdA

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

Reinforcement Learning

- Controlling robots with RL

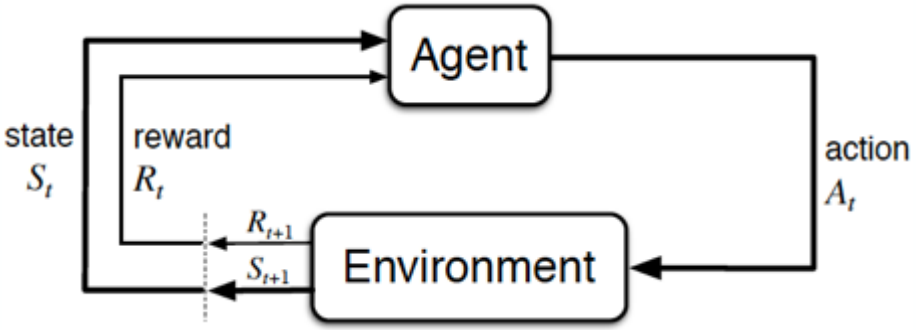
https://www.youtube.com/watch?v=0JL04JJjocc

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

Reinforcement Learning

- The RL Paradigm (reward hypothesis):
  - Do you agree with following statement?

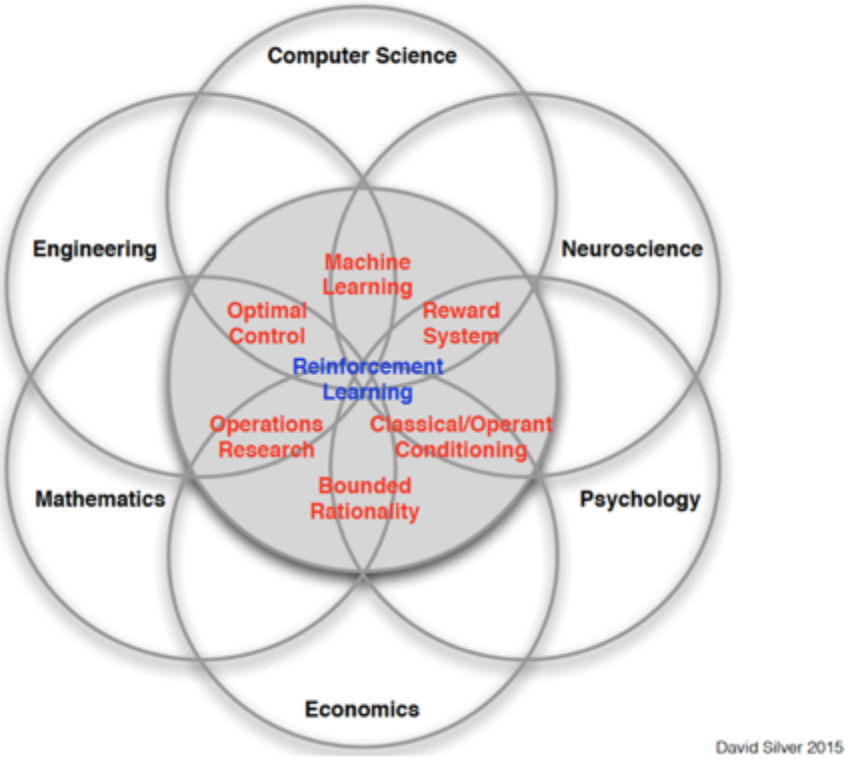
*”All goals can be described by the maximization of expected cumulative **reward**.”*



Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. MIT press.

Reinforcement Learning

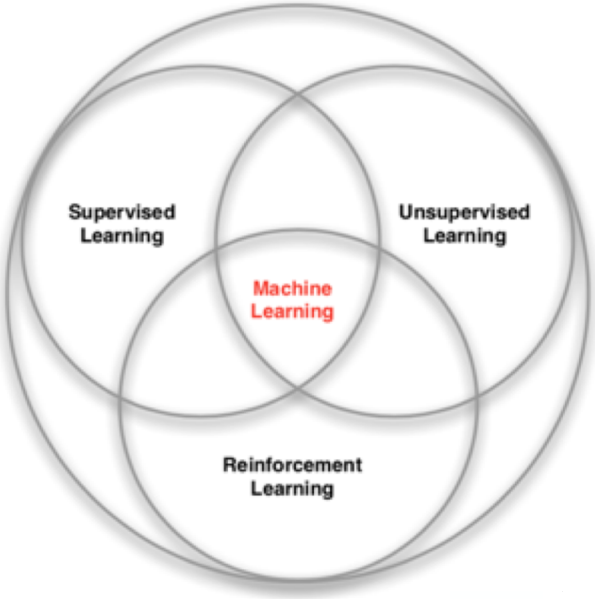
- RL vs the world: the many faces of Reinforcement Learning



Reinforcement Learning

- RL vs other ML branches:
  - No teacher/supervisor, only reward signals.
  - Delayed feedback, not instantaneous (credit assignment problem).
  - Learning by interaction between environment and agent over time.
  - Agent's actions affect the environment: Actions have consequences!!!

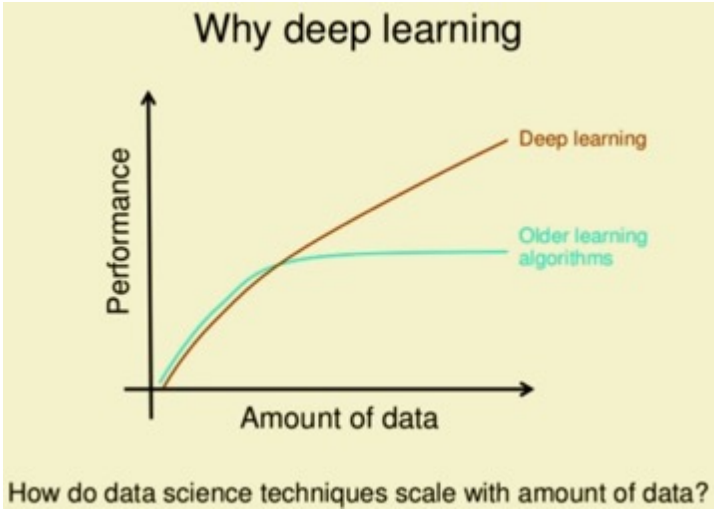
    → non i.i.d. aspect.
  - Active Learning process: the actions that the agent takes affect the subsequent data it receives



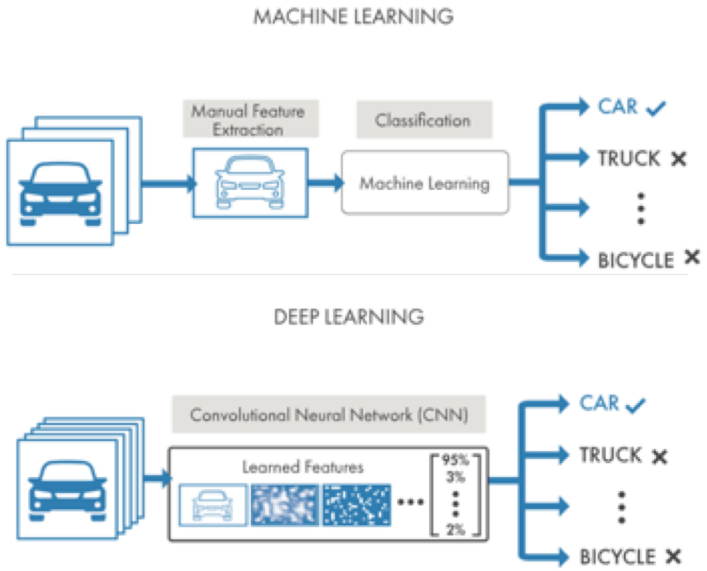
David Silver 2015

Reinforcement Learning

- Why RL now?
  - Taking advantage of advances in:
    - Deep Learning Algorithms (DL)



https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063



https://www.mathworks.com/discovery/deep-learning.html

Reinforcement Learning

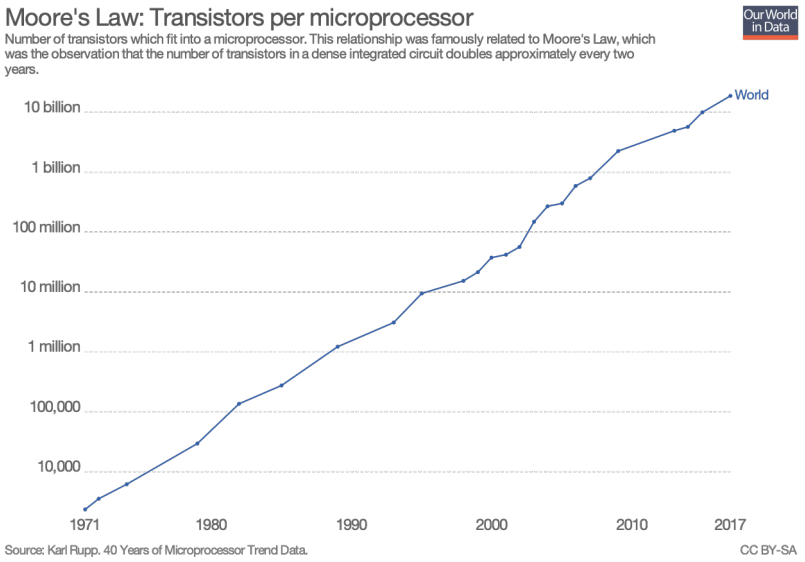
- Why RL now?
  - Taking advantage of advances in:
    - Deep Learning Algorithms (DL)

Reinforcement Learning

- Why RL now?
  - Taking advantage of advances in:
    - Hardware (CPU & Memory)

	OPENAI 1V1 BOT	OPENAI FIVE
CPUs	60,000 CPU cores on Azure	128,000 preemptible CPU cores on GCP
GPUs	256 K80 GPUs on Azure	256 P100 GPUs on GCP
Experience collected	~300 years per day	~180 years per day (~900 years per day counting each hero separately)

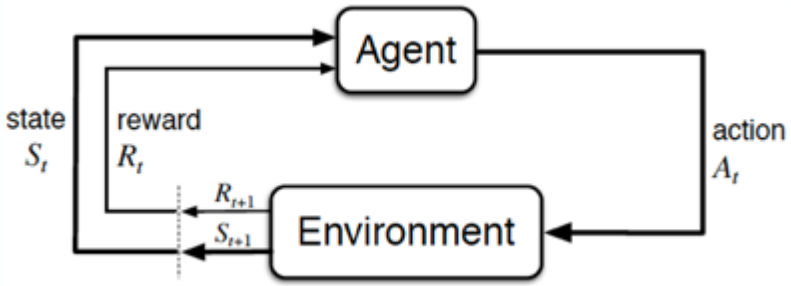
https://blog.openai.com/openai-five/



https://ourworldindata.org/technological-progress

Reinforcement Learning

- Agent learns by interacting with an environment over many time-steps:
- Markov Decision Process (MDP) is a tool to formulate RL problems
  - Description of an MDP $(\mathcal{S},\mathcal{A},\mathcal{P},\mathcal{R},\gamma)$:



Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. MIT press.

- At each step t, the agent:
  - is at state S_t,
  - performs action A_t,
  - receives reward R_t.
- At each step t, the environment:
  - receives action A_t from the agent,
  - provides reward R_t,
  - moves at state S_{t+1},
  - increments time $t \leftarrow t+1$.

Reinforcement Learning

- Expected long-term value of state s:

$$v(s) = \mathbb{E}(\mathrm{G}) = \mathbb{E}(R_0 + \gamma R_1 + \gamma^2 R_2 + \gamma^3 R_3 + \ldots + \gamma^t R_t)$$

- **Goal: maximize the expected return $\mathbb{E}(\mathrm{G})$.**
- We need a controller that helps us select the actions to maximize $\mathbb{E}(G)$.
- A policy π represents this controller:
  - π determines the agent's behavior, i.e., its way of acting
  - π is a mapping from state space $\mathcal{S}$ to action space $\mathcal{A}$

$$\pi : \mathcal{S} \mapsto \mathcal{A}$$

  - Two types of policies:
    - Deterministic policy: $a = \pi(s)$.
    - Stochastic policy: $\pi(a \mid s) = \mathbb{P}[A_t = a \mid S_t = s]$.
- **New goal: find a policy that maximizes the expected return!**

Reinforcement Learning

- How do we find optimal controllers for given (known) MDPs?
  - Bellman Optimality Equation for V

$$s \xrightarrow{\pi^*(s),\, \mathcal{R}(s,\pi(s))} s' \xrightarrow{\pi^*(s'),\, R_1} S_2 \xrightarrow{\pi^*(S_2),\, R_2} S_3 \dots S_{h-1} \xrightarrow{\pi^*(S_{h-1}),\, R_{h-1}} S_h$$

$$V^{\pi^*}(s) = \max_{a \in \mathcal{A}} \left\{ \underbrace{\mathcal{R}(s,a)}_{} + \underbrace{\gamma \sum_{s' \in \mathcal{S}} \mathcal{P}(s'|s,a)\, V^{\pi^*}(s')}_{} \right\}$$

  - Bellman Optimality Equation for Q

$$s \xrightarrow{a,\, \mathcal{R}(s,a)} s' \xrightarrow{\pi^*(s'),\, R_1} S_2 \xrightarrow{\pi^*(S_2),\, R_2} S_3 \dots S_{h-1} \xrightarrow{\pi^*(S_{h-1}),\, R_{h-1}} S_h$$

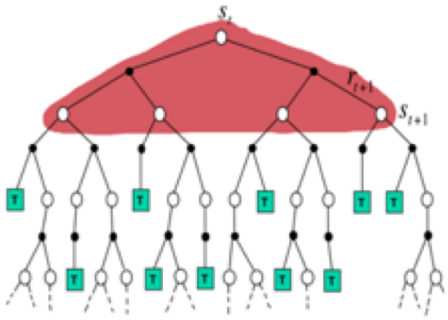
$$Q^{\pi^*}(s,a) = \underbrace{\mathcal{R}(s,a)}_{} + \underbrace{\gamma \sum_{s' \in \mathcal{S}} \mathcal{P}(s'|s,a) \max_{a' \in \mathcal{A}} Q^{\pi^*}(s',a')}_{}$$

Reinforcement Learning

- Estimate the Value Function

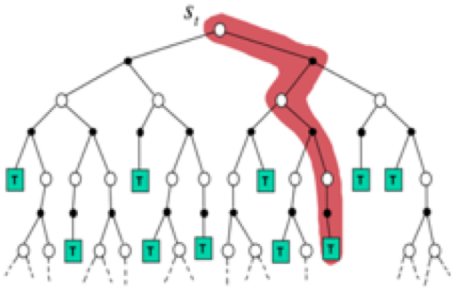
DP Backup

$$V(S_t) \leftarrow \mathbb{E}_\pi [R_{t+1} + \gamma V(S_{t+1})]$$



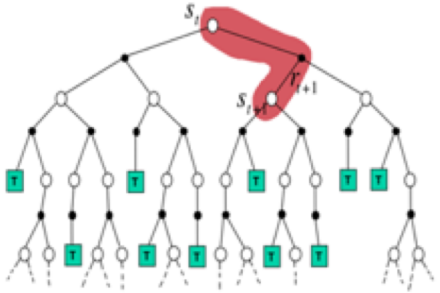
MC Backup

$$V(S_t) \leftarrow V(S_t) + \alpha (G_t - V(S_t))$$



TD Backup

$$V(S_t) \leftarrow V(S_t) + \alpha (R_{t+1} + \gamma V(S_{t+1}) - V(S_t))$$

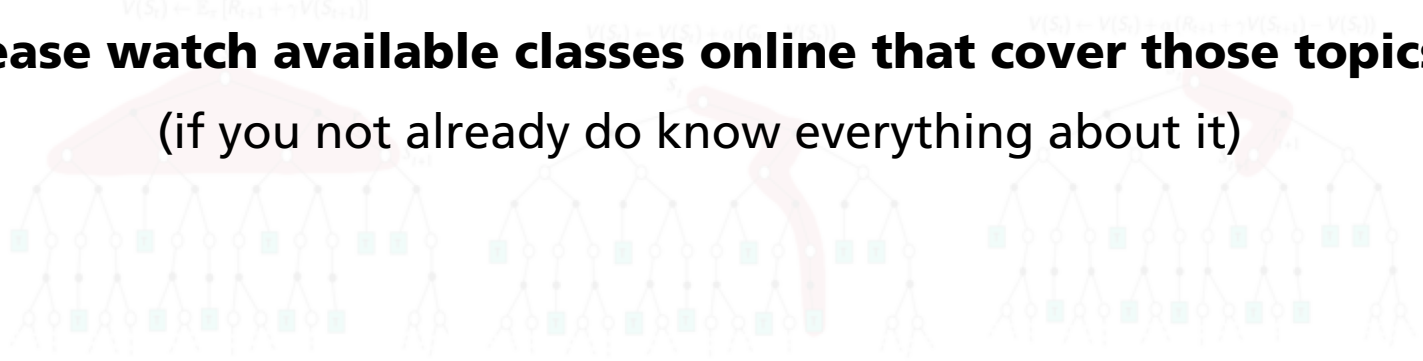


Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. MIT press.

Reinforcement Learning

- Estimate the Value Function

DP Backup | MC Backup | TD Backup

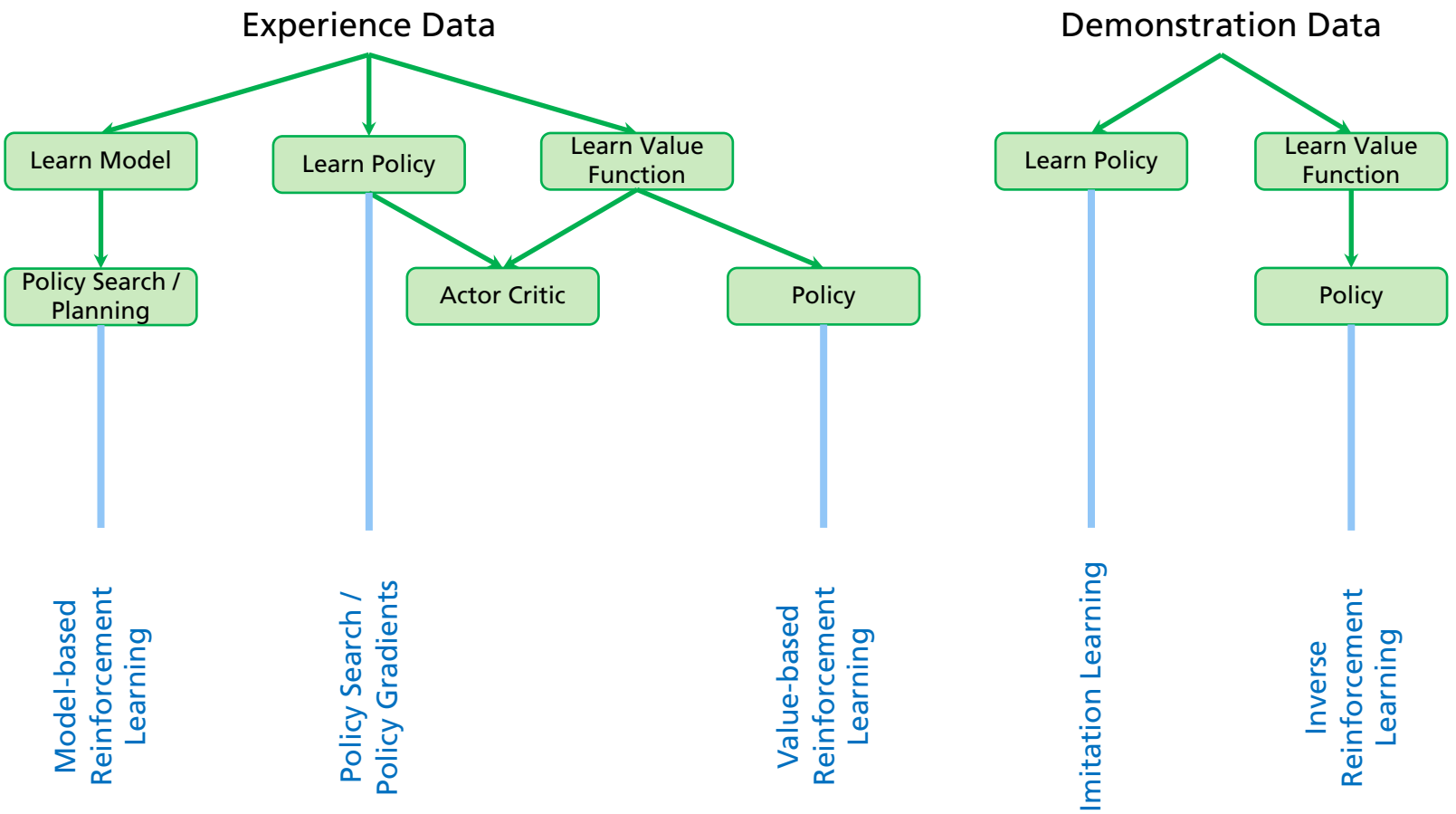


Please watch available classes online that cover those topics!

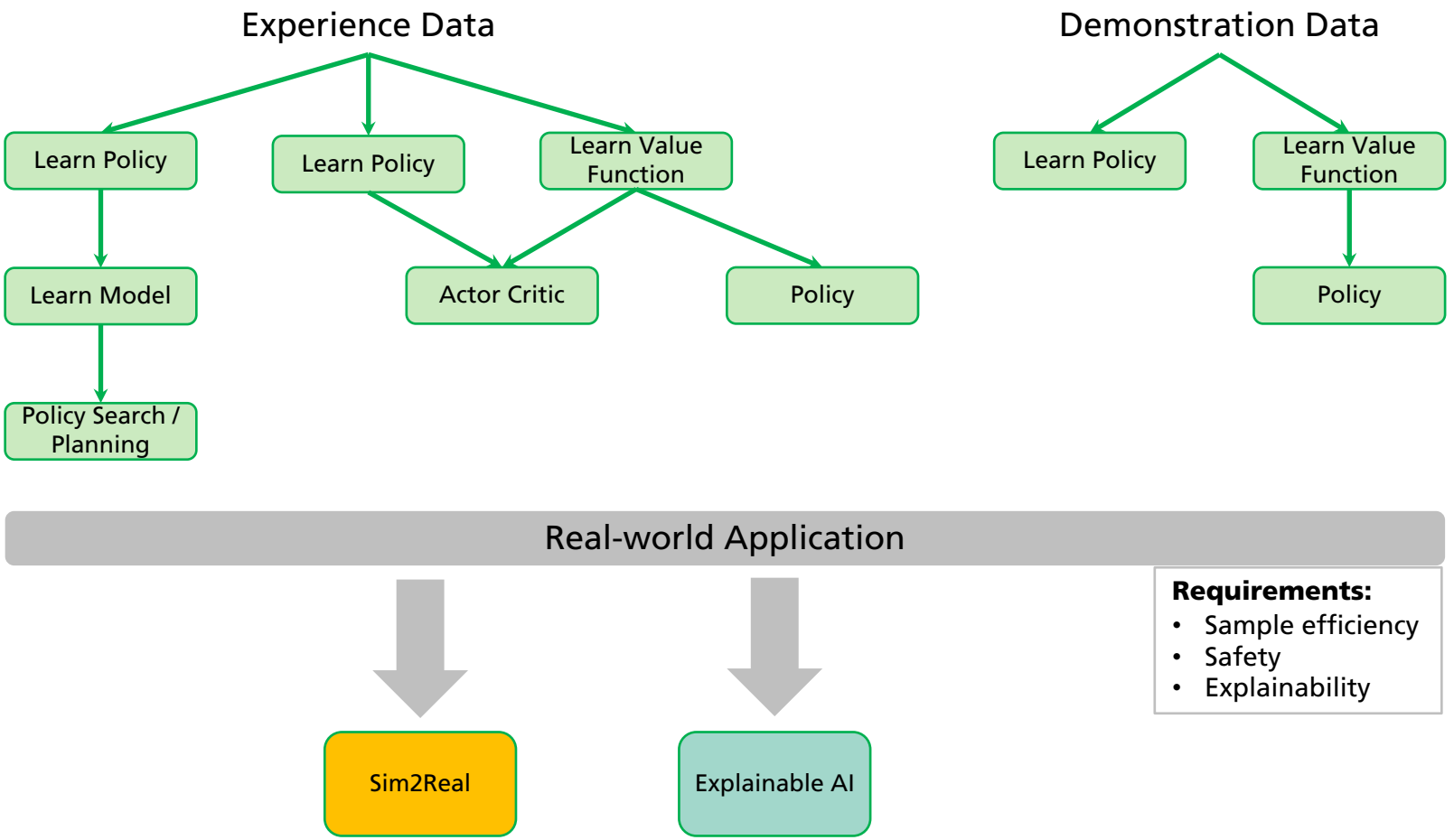
(if you not already do know everything about it)

Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. MIT press.

Deep Reinforcement Learning in the Real World



Deep Reinforcement Learning in the Real World

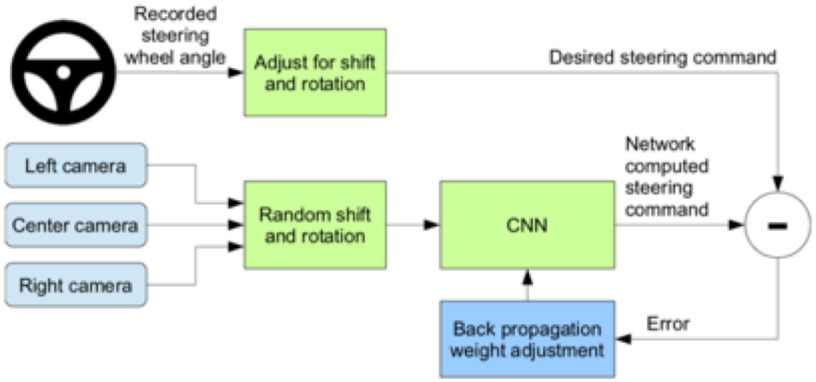


IMITATION LEARNING

Imitation Learning

Description:

- Learning from demonstrations provided by an expert
- Safe and sample efficient learning
- Two approaches:
  - Direct (Behavior Cloning)
  - Indirect (Inverse Reinforcement Learning)



Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., ... & Zhang, X. (2016). End to end learning for self-driving cars

https://www.youtube.com/watch?v=-96BEoXJMs0

Imitation Learning

Keywords (see references section):

- ALVINN
- NVIDIA end-to-end self-driving
- DAGGER
- Conditional Imitation Learning

Optional:

- Inverse Reinforcement Learning

Tips:

- http://rail.eecs.berkeley.edu/deeprlcourse/
- http://www.andrew.cmu.edu/course/10-703/
- https://sites.google.com/view/icml2018-imitation-learning/

Imitation Learning

References:

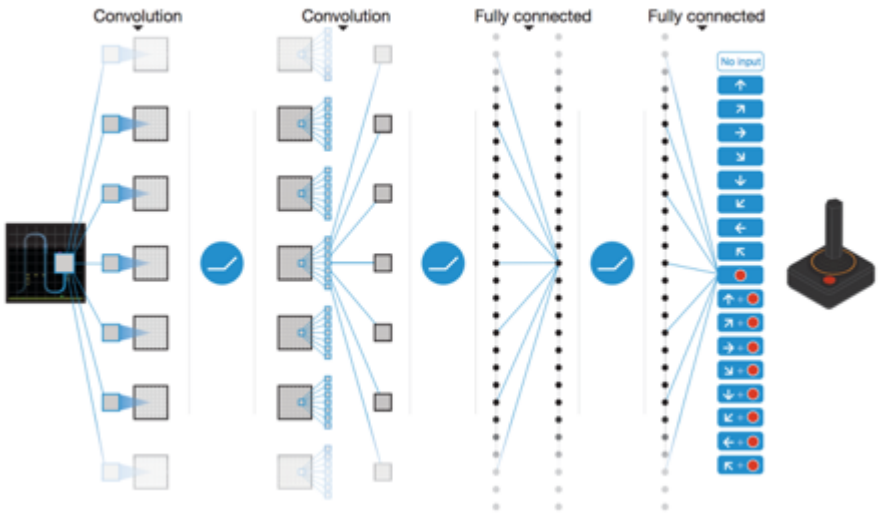
- Pomerleau, D. A. (1989). Alvinn: An autonomous land vehicle in a neural network. In *Advances in neural information processing systems* (pp. 305-313): https://papers.nips.cc/paper/95-alvinn-an-autonomous-land-vehicle-in-a-neural-network
- Giusti, A., Guzzi, J., Ciresan, D. C., He, F. L., Rodríguez, J. P., Fontana, F., ... & Scaramuzza, D. (2016). A Machine Learning Approach to Visual Perception of Forest Trails for Mobile Robots. *IEEE Robotics and Automation Letters*, *1*(2), 661-667: http://rpg.ifi.uzh.ch/docs/RAL16_Giusti.pdf
- **Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., ... & Zhang, X. (2016). End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316:* https://arxiv.org/abs/1604.07316**
- **Ross, S., Gordon, G., & Bagnell, D. (2011, June). A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 627-635): https://arxiv.org/abs/1011.0686**
- Ross, S., Melik-Barkhudarov, N., Shankar, K. S., Wendel, A., Dey, D., Bagnell, J. A., & Hebert, M. (2013, May). Learning monocular reactive uav control in cluttered natural environments. In *2013 IEEE international conference on robotics and automation* (pp. 1765-1772). IEEE: https://arxiv.org/abs/1211.1690
- Codevilla, F., Miiller, M., López, A., Koltun, V., & Dosovitskiy, A. (2018, May). End-to-end driving via conditional imitation learning. In 2018 IEEE International Conference on Robotics and Automation (ICRA) (pp. 1-9). IEEE: http://vladlen.info/papers/conditional-imitation.pdf
- Osa, T., Pajarinen, J., Neumann, G., Bagnell, J. A., Abbeel, P., & Peters, J. (2018). An algorithmic perspective on imitation learning. Foundations and Trends® in Robotics, 7(1-2), 1-179. ArXiv: https://arxiv.org/abs/1811.06711
- http://heli.stanford.edu

ADVANCED Q-LEARNING

Advanced Q-Learning

Description:

- Value-based Reinforcement Learning:
  - Estimate the optimal action value function $Q^*(s, a)$
  - Select best action in any state s following a greedy policy



Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Petersen, S. (2015). Human-level control through deep reinforcement learning. Nature, 518(7540), 529

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

Advanced Q-Learning

Keywords (see references section):

- Q-learning, DQN, Double DQN
- Prioritzed Experience Replay
- Dueling Networks
- Deep Recurrent Q-Learning

Optional:

- Present the findings of the paper: „Deep Reinforcement Learning and the Deadly Triad“

Tips:

- http://rail.eecs.berkeley.edu/deeprlcourse/
- http://www.andrew.cmu.edu/course/10-703/

Advanced Q-Learning

References:

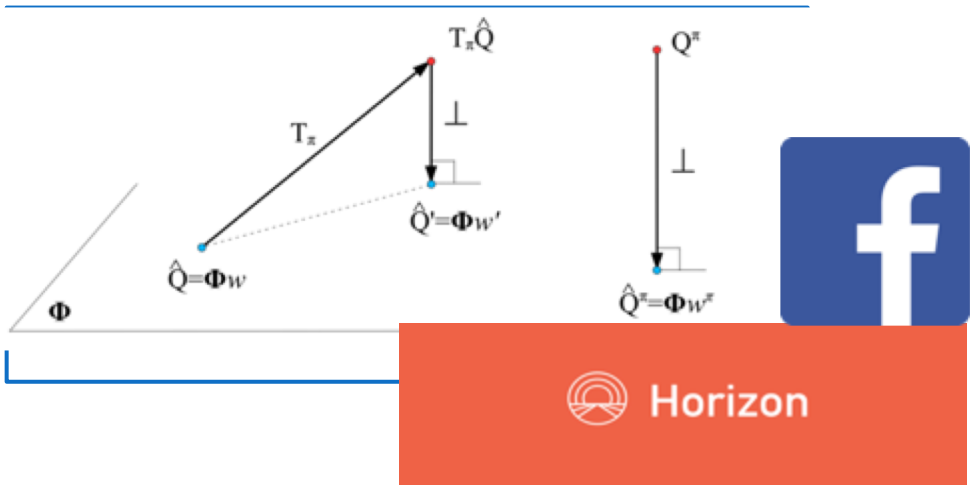
- Watkins, C. J. C. H. (1989). *Learning from delayed rewards*(Doctoral dissertation, King's College, Cambridge): https://link.springer.com/content/pdf/10.1007/BF00992698.pdf
- Lin, L. J. (1993). *Reinforcement learning for robots using neural networks* (No. CMU-CS-93-103). Carnegie-Mellon Univ Pittsburgh PA School of Computer Science: http://www.dtic.mil/dtic/tr/fulltext/u2/a261434.pdf
- **Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Petersen, S. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529: http://web.stanford.edu/class/psych209/Readings/MnihEtAlHassibis15NatureControlDeepRL.pdf**
- Van Hasselt, H., Guez, A., & Silver, D. (2016, February). Deep Reinforcement Learning with Double Q-Learning. In *AAAI* (Vol. 2, p. 5): http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/download/12389/11847
- **Wang, Z., Schaul, T., Hessel, M., Van Hasselt, H., Lanctot, M., & De Freitas, N. (2015). Dueling network architectures for deep reinforcement learning. *arXiv preprint arXiv:1511.06581*: https://arxiv.org/abs/1511.06581**
- Hausknecht, M., & Stone, P. (2015). Deep recurrent q-learning for partially observable mdps. *CoRR, abs/1507.06527*, *7*(1): http://www.aaai.org/ocs/index.php/FSS/FSS15/paper/download/11673/11503
- Van Hasselt, H., Doron, Y., Strub, F., Hessel, M., Sonnerat, N., & Modayil, J. (2018). Deep Reinforcement Learning and the Deadly Triad. arXiv preprint arXiv:1812.02648.: https://arxiv.org/abs/1812.02648

BATCH RL

Batch RL

Description:

- Batch RL decouples data collection from policy learning
- We learn a new policy offline using available data, without continuous interaction with the simulator or the real system



Gauci, J., Conti, E., Liang, Y., Virochsiri, K., He, Y., Kaden, Z., ... & Ye, X. (2018). Horizon: Facebook's Open Source Applied Reinforcement Learning Platform

https://www.youtube.com/watch?v=i8Cnas7QrMc&t=1s

Batch RL

Keywords (see references section):

- Least Squares Policy Iteration
- (Neural) Fitted Q Iteration
- Facebook‘s Horizon Framework

Optional:

- Counterfactual Policy Evaluation

Tips:

- http://www.intelligence.tuc.gr/~lagoudakis/DOCS/thesis.pdf
- Batch Reinforcement Learning presentation by Alan Fern, Oregon State University

Batch RL

References:

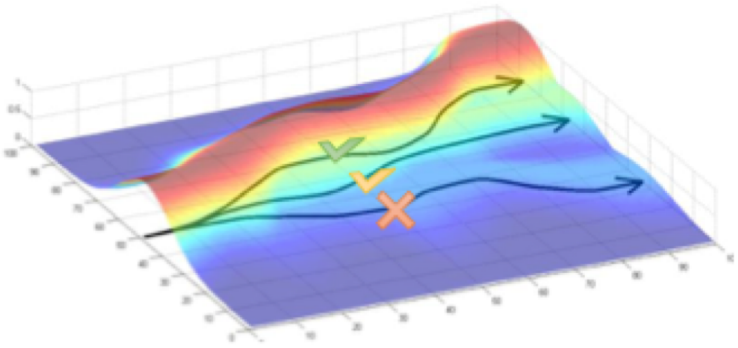
- Lange, S., Gabel, T., & Riedmiller, M. (2012). Batch reinforcement learning. In *Reinforcement learning* (pp. 45-73). Springer, Berlin, Heidelberg: http://ml.informatik.uni-freiburg.de/former/_media/publications/langegabelriedmiller2011chapter.pdf
- **Lagoudakis, M. G., & Parr, R. (2003). Least-squares policy iteration. *Journal of machine learning research*, *4*(Dec), 1107-1149: http://www.jmlr.org/papers/volume4/lagoudakis03a/lagoudakis03a.pdf + http://www.intelligence.tuc.gr/~lagoudakis/DOCS/thesis.pdf**
- **Ernst, D., Geurts, P., & Wehenkel, L. (2005). Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, *6*(Apr), 503-556: http://www.jmlr.org/papers/volume6/ernst05a/ernst05a.pdf**
- Riedmiller, M. (2005, October). Neural fitted Q iteration–first experiences with a data efficient neural reinforcement learning method. In *European Conference on Machine Learning* (pp. 317-328). Springer, Berlin, Heidelberg: http://ml.informatik.uni-freiburg.de/former/_media/publications/rieecml05.pdf
- **Gauci, J., Conti, E., Liang, Y., Virochsiri, K., He, Y., Kaden, Z., ... & Ye, X. (2018). Horizon: Facebook's Open Source Applied Reinforcement Learning Platform. arXiv preprint arXiv:1811.00260. ArXiv: https://arxiv.org/pdf/1811.00260.pdf**
- Levine, N., Zahavy, T., Mankowitz, D. J., Tamar, A., & Mannor, S. (2017). Shallow updates for deep reinforcement learning. In Advances in Neural Information Processing Systems (pp. 3135-3145). ArXiv: https://papers.nips.cc/paper/6906-shallow-updates-for-deep-reinforcement-learning.pdf

POLICY SEARCH

Policy Search

Description:

- For problems with continuous action spaces it is easier to learn/optimize the policy directly, instead of estimating the value function



Deep RL Class 2018, Sergey Levine

https://www.youtube.com/watch?v=5oBAYbOF2Qo

Policy Search

Keywords (see references section):

- Finite Differences, Cross Entropy, Augmented Random Seach
- Policy Gradient Theorem, REINFORCE, G(PO)MDP, PoWER
- Variance reduction and baseline selection

Optional:

- Natural Policy Gradient

Tips:

- http://rail.eecs.berkeley.edu/deeprlcourse/
- http://www.andrew.cmu.edu/course/10-703/
- https://spinningup.openai.com/en/latest/
- **https://icml.cc/2015/tutorials/PolicySearch.pdf**

Policy Search

References:

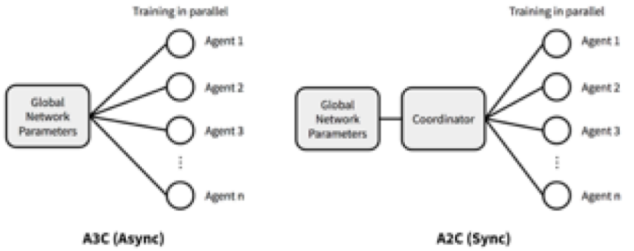
- **Deisenroth, M. P., Neumann, G., & Peters, J. (2013). A survey on policy search for robotics. *Foundations and Trends® in Robotics*, *2*(1–2), 1-142: https://spiral.imperial.ac.uk/bitstream/10044/1/12051/7/fnt_corrected_2014-8-22.pdf**
- Sigaud, O., & Stulp, F. (2019). Policy search in continuous action domains: an overview. *Neural Networks*. ArXiv: https://arxiv.org/pdf/1803.04706.pdf
- **Szita, I., & Lörincz, A. (2006). Learning Tetris using the noisy cross-entropy method. *Neural computation*, *18*(12), 2936-2941. link: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.80.6681&rep=rep1&type=pdf**
- Sutton, R. S., McAllester, D. A., Singh, S. P., & Mansour, Y. (2000). Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems* (pp. 1057-1063). Link: http://papers.nips.cc/paper/1713-policy-gradient-methods-for-reinforcement-learning-with-function-approximation.pdf
- Kakade, S. M. (2002). A natural policy gradient. In *Advances in neural information processing systems* (pp. 1531-1538). Link: http://papers.nips.cc/paper/2073-a-natural-policy-gradient.pdf
- Duan, Y., Chen, X., Houthooft, R., Schulman, J., & Abbeel, P. (2016, June). Benchmarking deep reinforcement learning for continuous control. In *International Conference on Machine Learning* (pp. 1329-1338): http://proceedings.mlr.press/v48/duan16.pdf
- Riedmiller, M., Peters, J., & Schaal, S. (2007, April). Evaluation of policy gradient methods and variants on the cart-pole benchmark. In *2007 IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning* (pp. 254-261). IEEE. link: http://is.tuebingen.mpg.de/fileadmin/user_upload/files/publications/ADPRL2007-Peters2_[0].pdf
- Kober, J., & Peters, J. R. (2009). Policy search for motor primitives in robotics. In *Advances in neural information processing systems* (pp. 849-856). Link: https://papers.nips.cc/paper/3545-policy-search-for-motor-primitives-in-robotics.pdf

ACTOR CRITIC

Actor Critic

Description:

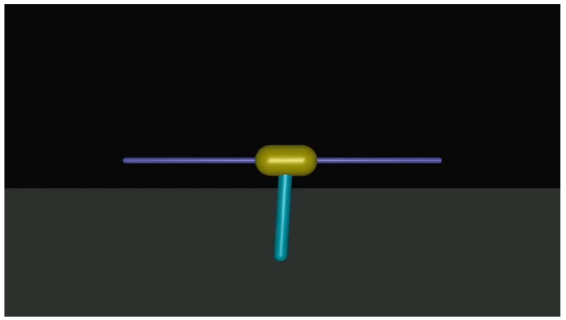
- Monte-Carlo policy gradients have high variance
- A critic (baseline) is used for variance reduction



https://lilianweng.github.io/lil-log/2018/04/08/policy-gradient-algorithms.html

https://www.youtube.com/watch?v=KJ15iGGJFvQ

https://www.youtube.com/watch?v=8RILnqPxo1s



https://www.youtube.com/watch?v=Ajjc08-iPx8

Actor Critic

Keywords (see references section):

- Actor-Critic Framework
- TRPO, PPO, PPO with clipped rewards, A3C/A2C
- DDPG, TD3

Optional:

- ACTKR

Tips:

- http://rail.eecs.berkeley.edu/deeprlcourse/
- http://www.andrew.cmu.edu/course/10-703/
- https://spinningup.openai.com/en/latest/
- https://lilianweng.github.io/lil-log/2018/04/08/policy-gradient-algorithms.html

Actor Critic

References:

- **Deisenroth, M. P., Neumann, G., & Peters, J. (2013). A survey on policy search for robotics. *Foundations and Trends® in Robotics*, *2*(1–2), 1-142: https://spiral.imperial.ac.uk/bitstream/10044/1/12051/7/fnt_corrected_2014-8-22.pdf**
- Sutton, R. S., McAllester, D. A., Singh, S. P., & Mansour, Y. (2000). Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems* (pp. 1057-1063). Link: http://papers.nips.cc/paper/1713-policy-gradient-methods-for-reinforcement-learning-with-function-approximation.pdf
- Schulman, J., Levine, S., Abbeel, P., Jordan, M. I., & Moritz, P. (2015, July). Trust Region Policy Optimization. In *Icml* (Vol. 37, pp. 1889-1897). ArXiv: https://arxiv.org/abs/1502.05477
- **Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*. ArXiv: https://arxiv.org/abs/1707.06347**
- Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., ... & Wierstra, D. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*. ArXiv: https://arxiv.org/abs/1509.02971
- Fujimoto, S., van Hoof, H., & Meger, D. (2018). Addressing function approximation error in actor-critic methods. *arXiv preprint arXiv:1802.09477*. ArXiv: https://arxiv.org/abs/1802.09477
- Wu, Y., Mansimov, E., Grosse, R. B., Liao, S., & Ba, J. (2017). Scalable trust-region method for deep reinforcement learning using kronecker-factored approximation. In *Advances in neural information processing systems* (pp. 5279-5288). ArXiv:
- Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., ... & Kavukcuoglu, K. (2016, June). Asynchronous methods for deep reinforcement learning. In *International conference on machine learning* (pp. 1928-1937). link: http://www.jmlr.org/proceedings/papers/v48/mniha16.pdf

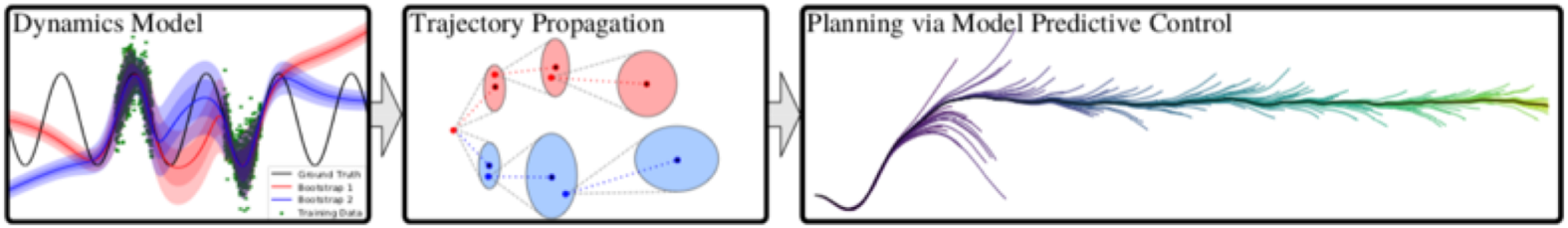
MODEL-BASED RL

Model-based RL

Description:

- Learn a model of the MDP dynamics
- Use the model for planning

https://www.youtube.com/watch?v=XiigTGKZfks&t=46s



Chua, K., Calandra, R., McAllister, R., & Levine, S. (2018). Deep Reinforcement Learning in a Handful of Trials using Probabilistic Dynamics Models

Model-based RL

Keywords (see references section):

- Monte Carlo Tree Search
- Model Predictive Control
- PILCO, Deep PILCO, Neural Network Dynamics Models

Optional:

- Present the findings of the paper: „Differentiable MPC for End-to-end Planning and Control“

Tips:

- http://rail.eecs.berkeley.edu/deeprlcourse/
- http://www.andrew.cmu.edu/course/10-703/
- https://github.com/nrontsis/PILCO

Model-based RL

References:

- **Deisenroth, M., & Rasmussen, C. E. (2011). PILCO: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on machine learning (ICML-11)* (pp. 465-472): https://www.ias.informatik.tu-darmstadt.de/uploads/Publications/Deisenroth_ICML_2011.pdf**
- **Deisenroth, M. P., Fox, D., & Rasmussen, C. E. (2015). Gaussian processes for data-efficient learning in robotics and control. *IEEE transactions on pattern analysis and machine intelligence*, *37*(2), 408-423: http://robotics.caltech.edu/wiki/images/d/d2/GPsDataEfficientLearning.pdf**
- Deisenroth, M. P. (2010). *Efficient reinforcement learning using Gaussian processes* (Vol. 9). KIT Scientific Publishing: https://pdfs.semanticscholar.org/c9f2/1b84149991f4d547b3f0f625f710750ad8d9.pdf
- **Gal, Y., McAllister, R., & Rasmussen, C. E. (2016, April). Improving PILCO with Bayesian neural network dynamics models. In *Data-Efficient Machine Learning workshop, ICML: http://mlg.eng.cam.ac.uk/yarin/website/PDFs/DeepPILCO.pdf***
- Punjani, A., & Abbeel, P. (2015, May). Deep learning helicopter dynamics models. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on* (pp. 3223-3230). IEEE: https://people.eecs.berkeley.edu/~pabbeel/papers/2015-ICRA-deep-learning-heli.pdf
- Chua, K., Calandra, R., McAllister, R., & Levine, S. (2018). Deep Reinforcement Learning in a Handful of Trials using Probabilistic Dynamics Models. *arXiv preprint arXiv:1805.12114*: https://arxiv.org/abs/1805.12114
- Amos, B., Jimenez, I., Sacks, J., Boots, B., & Kolter, J. Z. (2018). Differentiable MPC for End-to-end Planning and Control. In *Advances in Neural Information Processing Systems* (pp. 8299-8310).: https://www.cc.gatech.edu/~bboots3/files/DMPC.pdf
- Guo, X., Singh, S., Lee, H., Lewis, R. L., & Wang, X. (2014). Deep learning for real-time Atari game play using offline Monte-Carlo tree search planning. In *Advances in neural information processing systems* (pp. 3338-3346).

RL AND CONTINUOUS CONTROL

RL and Continuous Control

Description:

- There is a clear connection between reinforcement learning with continuous action spaces and modern (optimal) control theory
- Recently, a large set of continuous control robotic simulations have been developed for benchmarking deep RL algorithms
- Empirical results show that combining classical control theory designs and DRL can lead to faster learning

https://www.youtube.com/watch?v=OBcjhp4KSgQ&t=26s

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

RL and Continuous Control

Keywords (see references section):

- OpenAI Gym, MuJoCo, DeepMind Control Suite, pybullet
- Present the findings of the papers: „Structured control nets for deep reinforcement learning“ and „Towards Generalization and Simplicity in Continuous Control“

Optional:

- Present the findings of the papers: „ A tour of reinforcement learning: The view from continuous control“ and „ Emergent Complexity via Multi-agent Competition“

Tips:

- https://www.youtube.com/watch?v=nF2-39a29Pw

RL and Continuous Control

References:

- Recht, B. (2018). A Tour of Reinforcement Learning: The View from Continuous Control. *arXiv preprint arXiv:1806.09460:* https://arxiv.org/abs/1806.09460
- Duan, Y., Chen, X., Houthooft, R., Schulman, J., & Abbeel, P. (2016, June). Benchmarking deep reinforcement learning for continuous control. In *International Conference on Machine Learning* (pp. 1329-1338): http://proceedings.mlr.press/v48/duan16.pdf
- **Rajeswaran, A., Lowrey, K., Todorov, E. V., & Kakade, S. M. (2017). Towards generalization and simplicity in continuous control. In *Advances in Neural Information Processing Systems* (pp. 6550-6561): http://papers.nips.cc/paper/7233-towards-generalization-and-simplicity-in-continuous-control.pdf**
- **Todorov, E., Erez, T., & Tassa, Y. (2012, October). Mujoco: A physics engine for model-based control. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on* (pp. 5026-5033). IEEE: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.296.6848&rep=rep1&type=pdf**
- Tassa, Y., Erez, T., & Todorov, E. (2012, October). Synthesis and stabilization of complex behaviors through online trajectory optimization. In 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (pp. 4906-4913). IEEE.
- **Srouji, M., Zhang, J., & Salakhutdinov, R. (2018). Structured control nets for deep reinforcement learning. arXiv preprint arXiv:1802.08311. ArXiv: https://arxiv.org/abs/1802.08311**
- Bansal, T., Pachocki, J., Sidor, S., Sutskever, I., & Mordatch, I. (2017). Emergent complexity via multi-agent competition. arXiv preprint arXiv:1710.03748. ArXiv: https://arxiv.org/abs/1710.03748

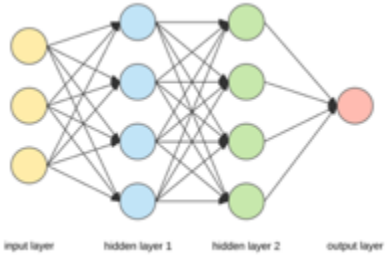
EXPLAINABLE RL (AND AI)

Explainable RL (and AI)

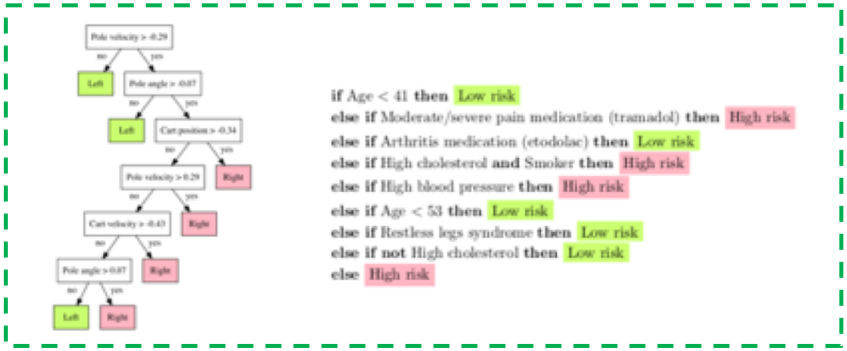
Description:

- AI systems are not perfect: accidents can always happen
- In case of unwanted behavior, the AI system needs to be interpretable, i.e. we should be able to understand what went wrong

https://www.theverge.com/2018/6/22/17492320/safety-driver-self-driving-uber-crash-hulu-police-report



https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6



Bastani, O., Pu, Y., & Solar-Lezama, A. (2018). Verifiable Reinforcement Learning via Policy Extraction

Explainable RL (and AI)

Keywords (see references section):

- LIME, VIPER
- Salient maps for Reinforcement learning (hint: start with the paper and code from: „Visualizing and Understanding Atari Agents“)

Optional:

- Present the findings of the paper: „ Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models“

Tips:

- https://www.hhi.fraunhofer.de/en/departments/vca/research-groups/machine-learning/research-topics/interpretable-machine-learning.html
- https://greydanus.github.io/2017/11/01/visualize-atari/

Explainable RL (and AI)

References:

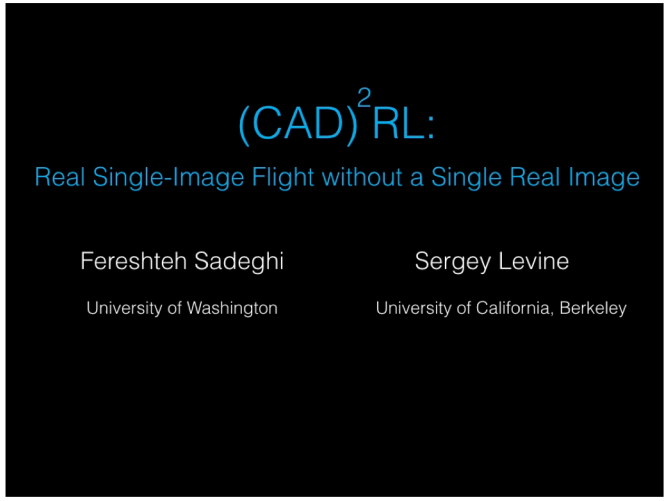
- **Gunning, D. (2017). Explainable artificial intelligence (xai). *Defense Advanced Research Projects Agency (DARPA), nd Web:* https://www.darpa.mil/attachments/XAIProgramUpdate.pdf**
- **Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135-1144).**
  ACM: https://arxiv.org/pdf/1602.04938.pdf?__hstc=200028081.1bb630f9cde2cb5f07430159d50a3c91.1523923200081.1523923200082.1523923200083.1&__hssc=200028081.1.1523923200084&__hsfp=1773666937
- **http://www.heatmapping.org**
- Bastani, O., Kim, C., & Bastani, H. (2017). Interpreting blackbox models via model extraction. *arXiv preprint arXiv:1705.08504:* https://arxiv.org/abs/1705.08504
- **Bastani, O., Pu, Y., & Solar-Lezama, A. (2018). Verifiable Reinforcement Learning via Policy Extraction. *arXiv preprint arXiv:1805.08328*: https://arxiv.org/pdf/1805.08328.pdf + https://obastani.github.io/docs/viper-presentation.pdf**
- Samek, W., Wiegand, T., & Müller, K. R. (2017). Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*. ArXiv: https://arxiv.org/pdf/1708.08296.pdf
- Greydanus, S., Koul, A., Dodge, J., & Fern, A. (2017). Visualizing and understanding atari agents. *arXiv preprint arXiv:1711.00138*.

SIM2REAL

Simulation to Reality Transfer

Description:

- Deep RL algorithms usually require a large number of samples
- The obvious solution is to use simulators to learn a policy and then fine-tune in the real system



https://www.youtube.com/watch?v=nXBWmzFrj5s&t=218s

Simulation to Reality Transfer

Keywords (see references section):

- Domain/Dynamics Randomization
- EPOPT, Progressive Nets, CAD2RL
- Present the findings of the paper: „Driving Policy Transfer via Modularity and Abstraction“

Optional:

- Present the findings of the paper: „Using simulation and domain adaptation to improve efficiency of deep robotic grasping“

Tips:

- http://rail.eecs.berkeley.edu/deeprlcourse/
- http://www.andrew.cmu.edu/course/10-703/

Simulation to Reality Transfer

References:

- Peng, X. B., Andrychowicz, M., Zaremba, W., & Abbeel, P. (2018, May). Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 1-8). IEEE.: https://arxiv.org/pdf/1710.06537.pdf
- **Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., & Abbeel, P. (2017, September). Domain randomization for transferring deep neural networks from simulation to the real world. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on* (pp. 23-30). IEEE: https://arxiv.org/pdf/1703.06907.pdf**
- **Rusu, A. A., Vecerik, M., Rothörl, T., Heess, N., Pascanu, R., & Hadsell, R. (2016). Sim-to-real robot learning from pixels with progressive nets. *arXiv preprint arXiv:1610.04286:* https://arxiv.org/abs/1610.04286**
- **Sadeghi, F., & Levine, S. (2016). CAD2RL: Real single-image flight without a single real image. *arXiv preprint arXiv:1611.04201:* https://arxiv.org/pdf/1611.04201.pdf**
- Rajeswaran, A., Ghotra, S., Ravindran, B., & Levine, S. (2016). Epopt: Learning robust neural network policies using model ensembles. *arXiv preprint arXiv:1610.01283:* https://arxiv.org/pdf/1610.01283.pdf
- Müller, M., Dosovitskiy, A., Ghanem, B., & Koltun, V. (2018). Driving policy transfer via modularity and abstraction. *arXiv preprint arXiv:1804.09364*. ArXiv: https://arxiv.org/abs/1804.09364
- Bousmalis, K., Irpan, A., Wohlhart, P., Bai, Y., Kelcey, M., Kalakrishnan, M., ... & Levine, S. (2018, May). Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In *2018 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 4243-4250). IEEE: https://arxiv.org/pdf/1709.07857.pdf

What do you need to do?
…and what do you get?

Tasks (5 ECTS):

1. A **30 minute presentation** (+10 mins for questions, language: English)
   - **Being present** for the talks of others participants
   - **Active engagement** through listening, discussion and questioning
2. Writing a **thesis** (8 pages NeurIPS or ICLR style)

Deadline:

- **Slides:** a week prior to your presentation
- **Thesis:** prior to start of classes in winter term, i.e., Mid-Oct, 2019

Certificate:

- 5 ECTS graded (presentation + thesis)
- 2.5 ECTS benotet (presentation + demo)

What do you need to do?
What else makes sense?

Show a Demo / put a live demo in your presentation:

- Either your own or just copy one from the internet (do not forget proper referencing for the latter case)
- While there are Jupyter-Notebooks for almost any machine learning method there is a lack for RL methods
  - However, there is many plan-python available on github

Why should you attend?

Goals of the seminar

Goals of the seminar

Why?

- **Understand** a topic:
  - Motivation (why do we need it?)
  - History (why & how did we get there?)
  - Algorithm (how does it work? & advantages/disadvantages)
  - State of the Art (what is currently happening out there?)
- **Explain** a topic:
  - From easy to hard (start with the inital and easy to understand problem motivation)
  - From common-sense to unknown territorry (build briges among the topics)
- **Present** a topic:
  - Present without textbook to an audience
  - Having fun to present, explain, and teach ☺

Tips & tricks
Preparation

Going familiar with the topic:

- **Contact your supervisor early:**
  - Getting your questions answered
  - Get tips for keywords, papers, examples, literature early
- Start more than 3 to 4 weeks before your presentation
- Search for real-world applications

Presentation:

- Let **others** look at them (language, mistakes, etc.)
- **Send the slides** to us latest a week before your presentation
- **Perform a dry-run!** (Talking fluidly, full sentences, „ehm“,...)
- Be on time to (your own) presentation(s) (and take care of laptop, board, etc. by yourself)
- Check font size and colors on the presenter

Tips & tricks
Slides

- No #wall-of-text – be precise and put only what is necessary
- Use your own words
- No plagiarism!
  - **Any** used **references** as footnote or refertence (be careful also with pictures!)
- Applications:
  - We have a template for MS Office
  - We also have some “nighly-build“ latex style

Tips & tricks
Slides

- **Not more than a single „major-idea“ per slide**
  (what is the take-away from this slide?)
- Not full sentences
- Font size: minimal 20, maximal 28
- Picture (colors are so beautiful!)
- Animations must make sense!
- Expect to need 1-2 minutes per slide (only you know your slides a-priori)
- Take care to be synchronous with what you have on your slides
- Slide numbers on each slide; name & topic at least on starting slide

Tips & tricks
Presentation

- Listeners can only listen carefully for about 20-30 minutes – that might be a problem for you
- Tips:
  - Change topics
  - Pictures, graphs, measurements, videos, etc.
  - (funny/meaningful) citations?
  - Change medium, i.e., use whiteboard
  - **Show something you did**
    - For instance a Live Demo
- Give a summary at the end (main points)

More ideas:

- http://www.vs.inf.ethz.ch/publ/slides/seminarvortraege.pdf

Appointments

Topics

	Topic	Date	Author	Download Material
0.	Introduction	April	Christopher Mutschler	
1.	Imitation Learning		Elgiz Bagcilar	
2.	Policy Search		Karthik Shetty	
3.	Actor-Critic		Sujit Sahoo	
4.	Advanced Q-Learning		Christian Klose	
5.	Model-based RL		Srikrishna Jaganathan	
6.	Explainable RL (and AI)		Daniel Luge	
7.	Batch Reinforcement Learning			
8.	Reinforcement Learning and Continuous Control		Sacha Medaer	
9.	Simulation to Reality Transfer			

Appointment
How do we get there

- Doodle to find it
- We will need 2 appointments
  - *1. Doodle-Poll to check for availability (**until April, 30th 2019**)*

    Please be generous. The results will be a list of available dates for our presentations. You will not be assigned a slot yet!
  - *2. Doodle-Poll to assign your slot (**until May, 12th 2019**)*

    We take a number of available dates from the 1st poll and you can select the dates for your own presentation according to your exam schedule.
- **Advice:** the more generous you are in the 1st poll the higher is the probability to get along with 2 dates!
- Same holds for the 2nd poll…