

# reinforcement learning

1

# pavlov's dog



2

# robot flipping pancakes

**Robot Motor Skill  
Coordination with EM-based  
Reinforcement Learning**

Petar Kormushev, Sylvain Calinon,  
and Darwin G. Caldwell

Italian Institute of Technology

3



<http://cs.stanford.edu/groups/littledog/>



4

## another example

- a child learning to walk:
  - tries out many different strategies
  - some do not work (**falling**), some seem to work (**staying up longer and longer**)
  - the **ones that do not work are discarded**
  - the **ones that work are tried again and again** until perfected or replaced by better strategies

5

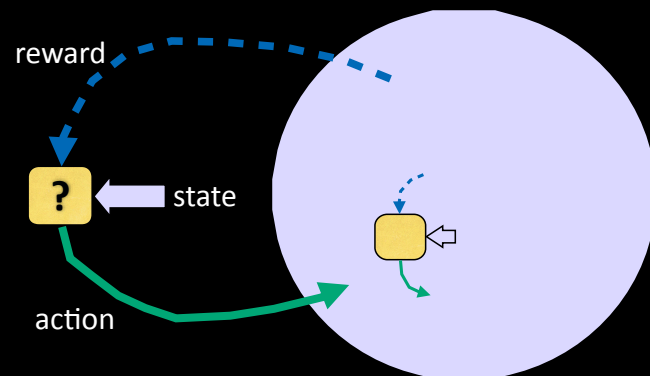
## hovering... inverted!



Inverted autonomous helicopter flight via reinforcement learning, Andrew Y. Ng, Adam Costes, Mark Diel, Varun Ganapathi, Jamie Schulte, Ben Tse, Eric Berger and Eric Liang. In *International Symposium on Experimental Robotics*, 2004.  
URL: <http://heli.stanford.edu/>

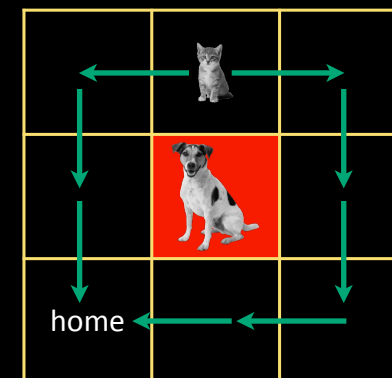
6

## the problem



7

## toy problem

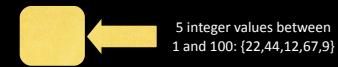


8

# state and action spaces

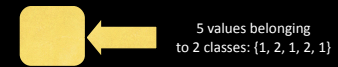
- size of these spaces can be **quite large**
- specifying the spaces is crucial in designing a good learning agent

9



size of state space =  $100 \times 100 \times 100 \times 100 \times 100$

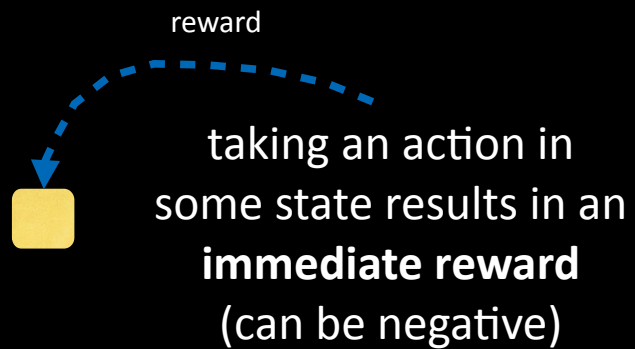
can quantise state space differently



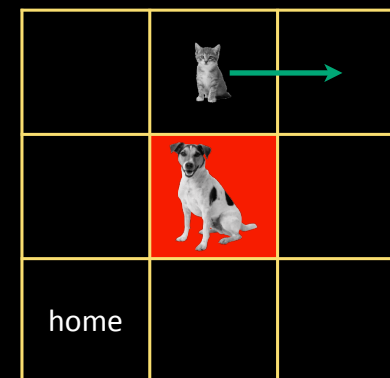
size of state space =  $2 \times 2 \times 2 \times 2 \times 2$

in the toy problem? 9

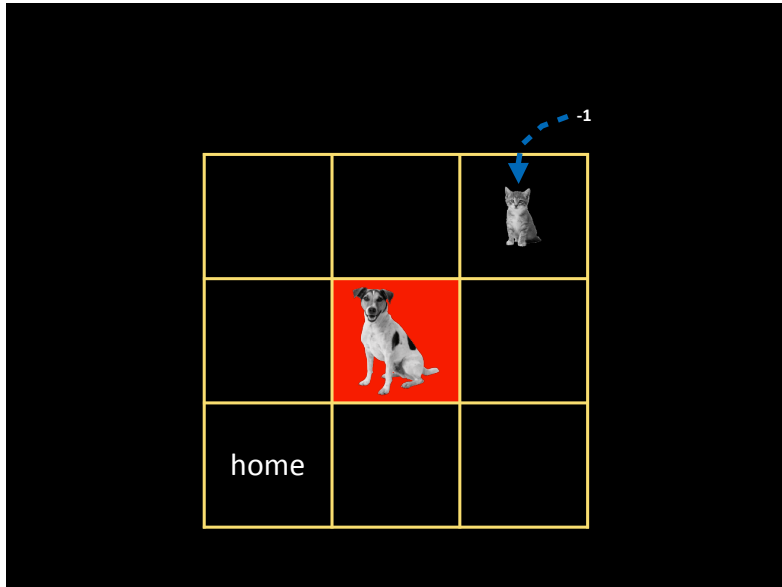
10



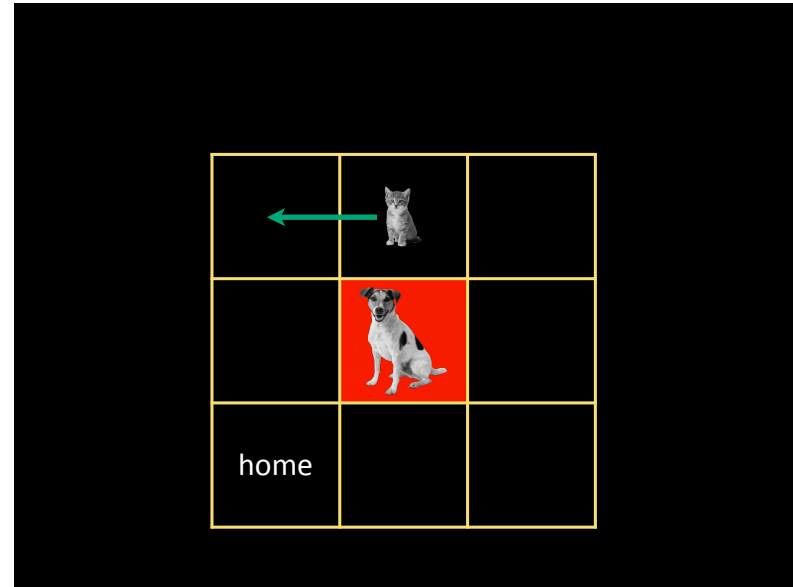
11



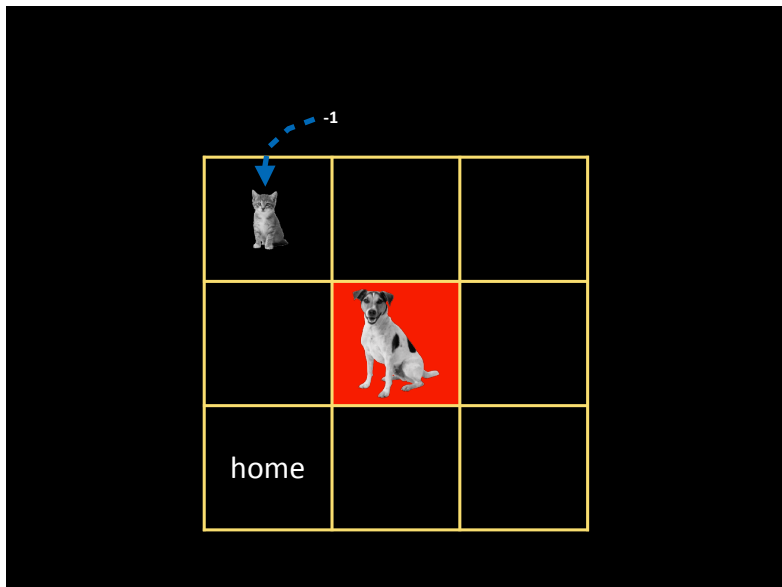
12



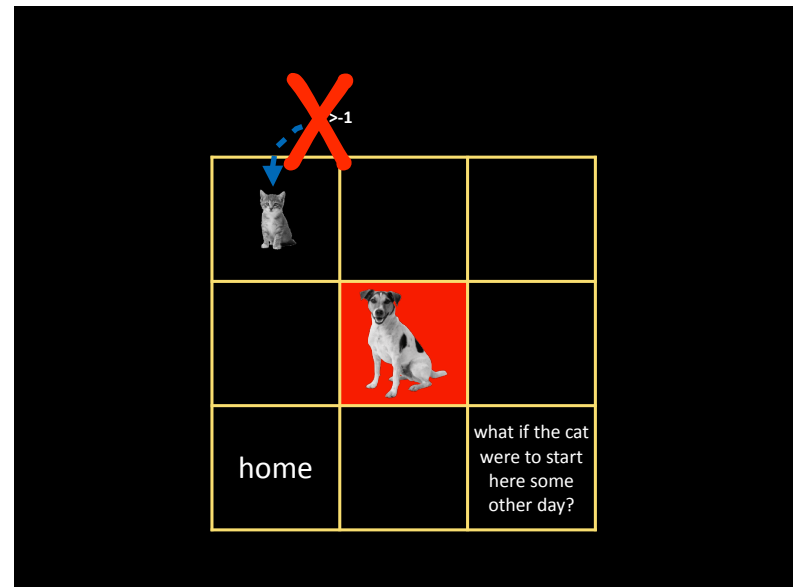
13



14



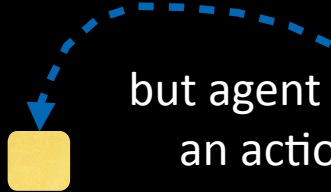
15



16

reward system should tell  
the agent:  
**what to achieve**  
rather than how to achieve

17

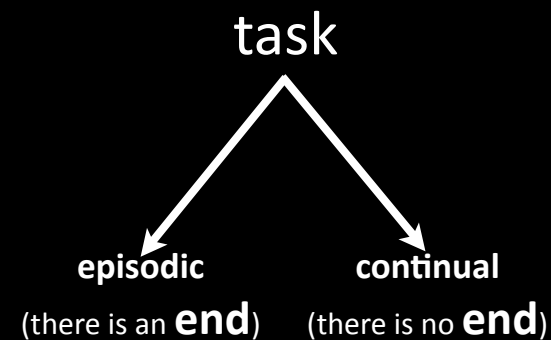
reward?  
  
but agent has to choose  
an action based on  
**expected “long term”  
reward** (cumulative  
reward in the long run)

18

**expected “long term”  
reward** (cumulative  
reward in the long run)



19



20

## episodic

(there is an **end**)

agent taking **finite (say 5) steps** till the end...

should act based on the  
**average of the following**

$$R_0 = r_1 + r_2 + r_3 + r_4 + r_5$$

21

## continual

(there is no **end**)

agent can continue acting for **infinite steps**  
**in time...**

should **discount** future rewards and act based on  
the **average of the following**

$$R_0 = r_1 + \gamma r_2 + \gamma^2 r_3 + \gamma^3 r_4 + \gamma^4 r_5 + \dots$$

22

## discount

future reward is probably **more**  
**uncertain** than immediate reward

shortsighted?  
 $\gamma=0$

$$0 \leq \gamma \leq 1$$

farsighted?  
 $\gamma=1$

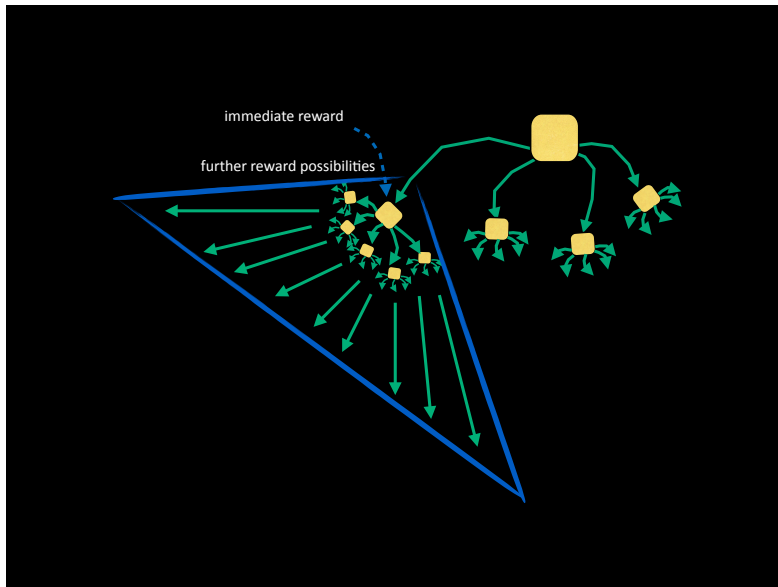
$$R_0 = r_1 + \gamma r_2 + \gamma^2 r_3 + \gamma^3 r_4 + \gamma^4 r_5 + \dots$$

23

$$R_0 = \sum_{k=0}^T \gamma^k r_{k+1}$$

$$E \left\{ R_t = \sum_{k=0}^T \gamma^k r_{t+k+1} \right\}$$

24



25

$$E \left\{ R_t = \sum_{k=0}^T \gamma^k r_{t+k+1} \right\}$$

26

but these expected rewards are **not known to agent** beforehand!

27

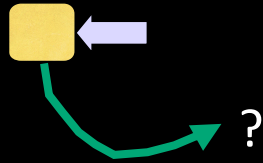
whether they are known or not, the agent has to act somehow!

**how to act/action selection?**

**how to get to know/estimate?**

28

action selection?



**values** of each possible action  
in the current state?

**expected reward** for  
carrying out the action is its **value**

29

but what are  
**these values?**

<<expected rewards are not known>>  
<<actions based on expected rewards>>

these expected rewards  $E\{R_t\}$  are to be  
**estimated by agent**  
**whilst acting!**

30



	a	b	c
1	2	0	1
2	3	0	-1
3	-5	6	2
4	2	3	1
.	.	.	.
.	.	.	.
.	.	.	.
n	7	8	7

agent maintains **values**  
for actions within each state

selects actions using these values based on a  
**“policy”**

31



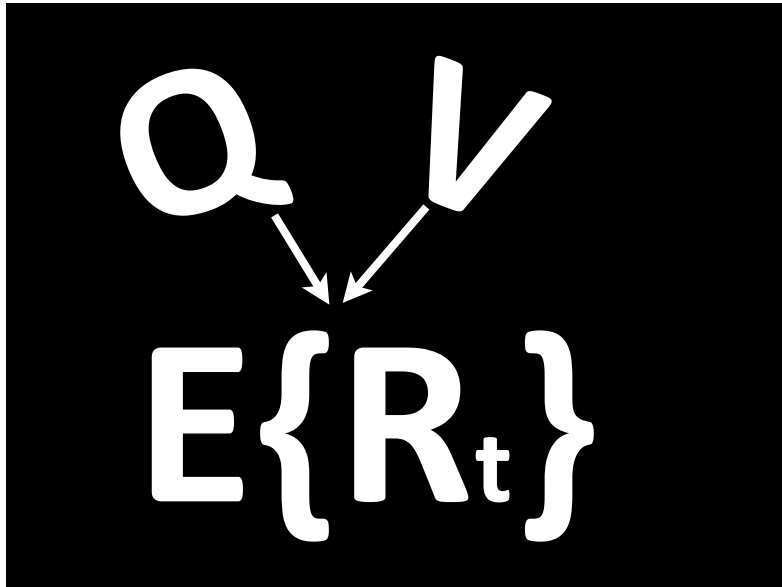
1	2
2	3
3	-5
4	2
.	.
.	.
.	.
n	7

agent maintains **state values**

selects actions using these values based on a  
**“policy”**

32





33

## policy?

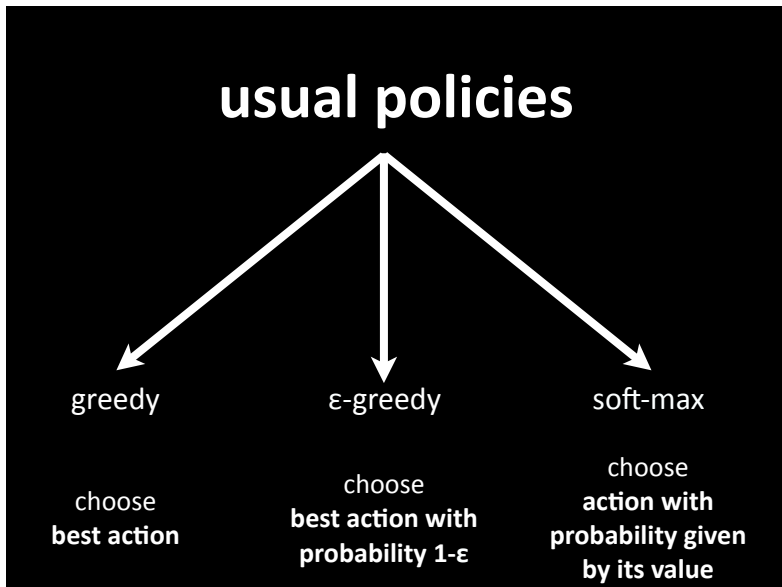
probability of choosing  
an action, given a state

$\pi$ 

$Q^\pi(s, a)$

$V^\pi(s)$

34



35

## exploration vs. exploitation

policy = multi-armed bandit strategy




Yamaguchi先生, [http://en.wikipedia.org/wiki/File:Las\\_Vegas\\_slot\\_machines.jpg](http://en.wikipedia.org/wiki/File:Las_Vegas_slot_machines.jpg)

Learning to be different: Heterogeneity and efficiency in distributed smart camera networks, P. R. Lewis, L. Esterle, A. Chandra, B. Rinner, and X. Yao, In Proceedings of the IEEE Conference on Self-Adaptive and Self-Organizing Systems (SASO), IEEE, 2013.  
 (forthcoming) Static, dynamic and adaptive heterogeneity in socio-economic distributed smart camera networks, P. R. Lewis, L. Esterle, A. Chandra, B. Rinner, J. Torresen, and X. Yao, ACM Transactions on Autonomous and Adaptive Systems (TAAS), ACM, 2014.

A/B Testing

36

**estimation?**

<<use currently visible rewards to update values of where you are coming from>>

the current state (or state-action pair) has an **estimated value** (say zero/random initially), which can be **used together with  $r_{t+1}$**  to **update value** of previous state (or state-action pair)

37

i.e.

fraction of (currently visible rewards - old value)

+

old value

↓

new value

(1-fraction) old value + fraction curr. vis. rewards

38

immediate reward  $r_{t+1}$

further reward possibilities

$Q(s, a)$

$r_{t+1} + \gamma Q(s', a')$

$r_{t+1} + \gamma Q(s_{t+1}, a_{t+1})$

$r_{t+1} + \gamma E\{R_{t+1}\}$

$R_t = \sum_{k=0}^{T-1} \gamma^k r_{t+k+1}$

39

$$V(s) \leftarrow V(s) + \mu(r + \gamma V(s') - V(s))$$

e.g.

$$Q(s, a) \leftarrow Q(s, a) + \mu(r + \gamma Q(s', a') - Q(s, a))$$

40

e.g. update  
a lookup table maintaing  
expected rewards

	a	b	c
1	2	0	1
2			-1
3	5		2
4	2	3	1
·	·	·	·
·	·	·	·
n	7	8	7

	1	2
1		
2		
3		
4		
·	·	·
·	·	·
n		7

41

$$Q(s,a) \leftarrow Q(s,a) + \mu(r + \gamma Q(s',a') - Q(s,a))$$

let's play with a version of the  
above update rule:

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

42

indicates a' to be the action  
with maximum value in next  
state s'

let's play with a version of the  
above update rule:

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

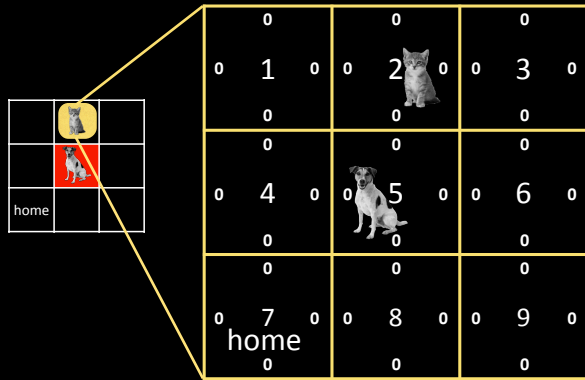
43

our toy problem  
lookup table

	N	S	E	W
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0

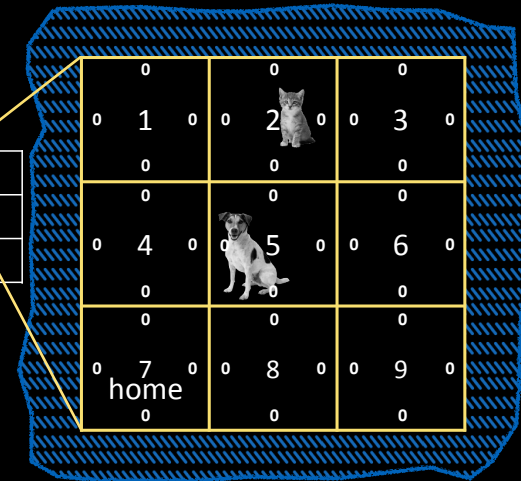
44

# our toy problem lookup table



45

reward  
structure?



move...

to any cell except 5 and 7:  
-1

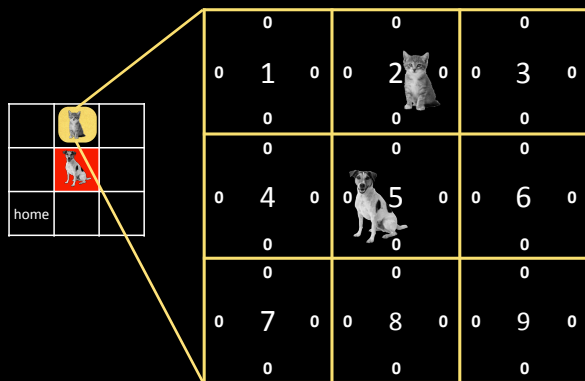
out of bounds:  
-5

to 5:  
-10

to 7/home:  
10

46

let's fix  $\mu = 0.1$ ,  $\gamma = 0.5$

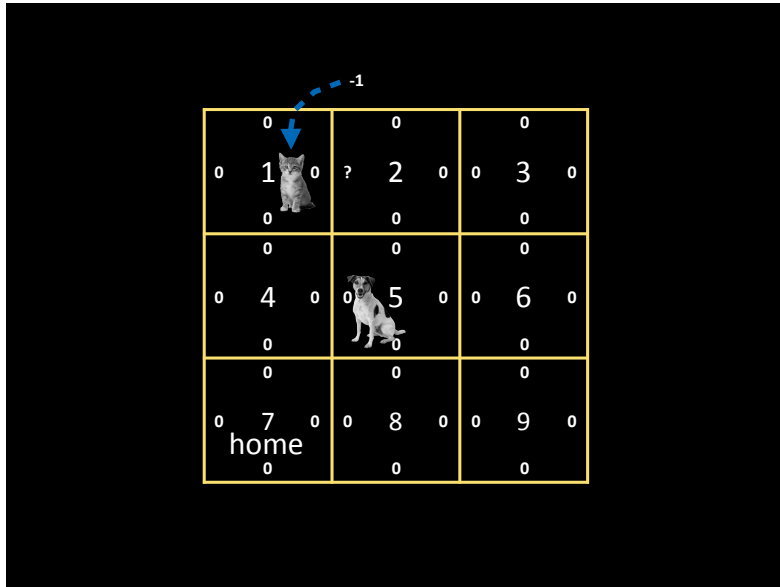


47

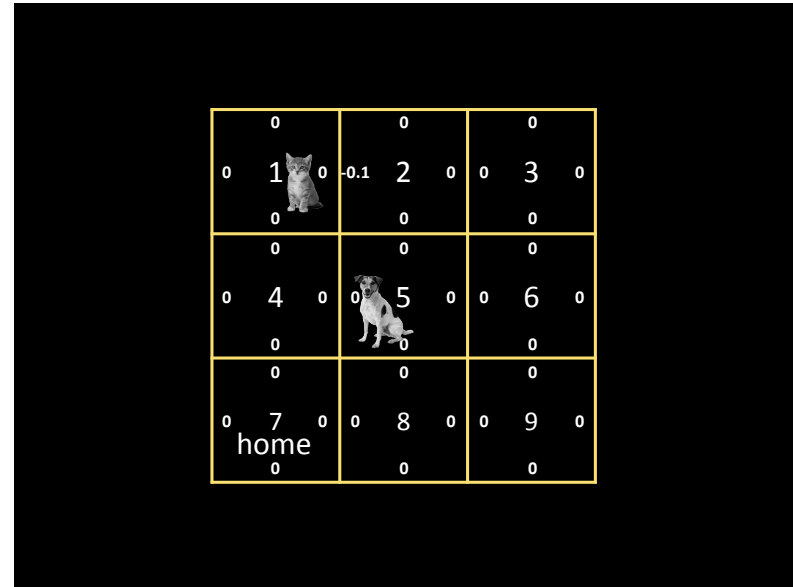


episode 1 begins...

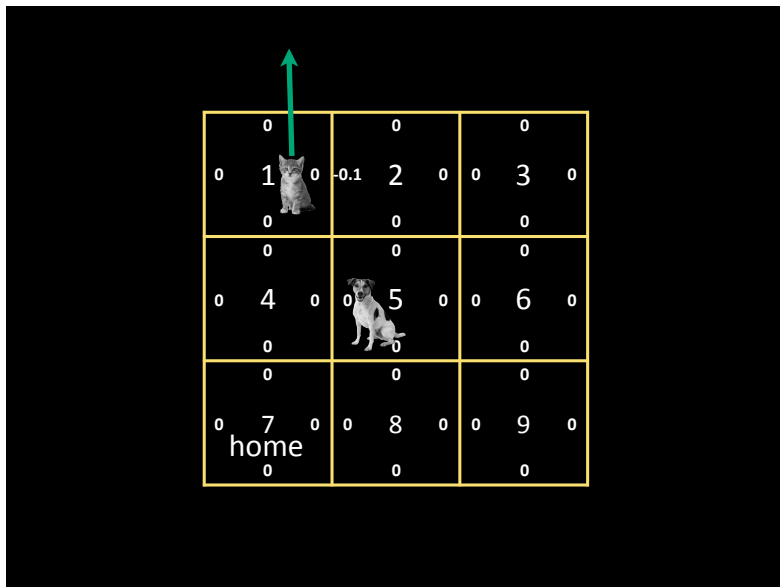
48



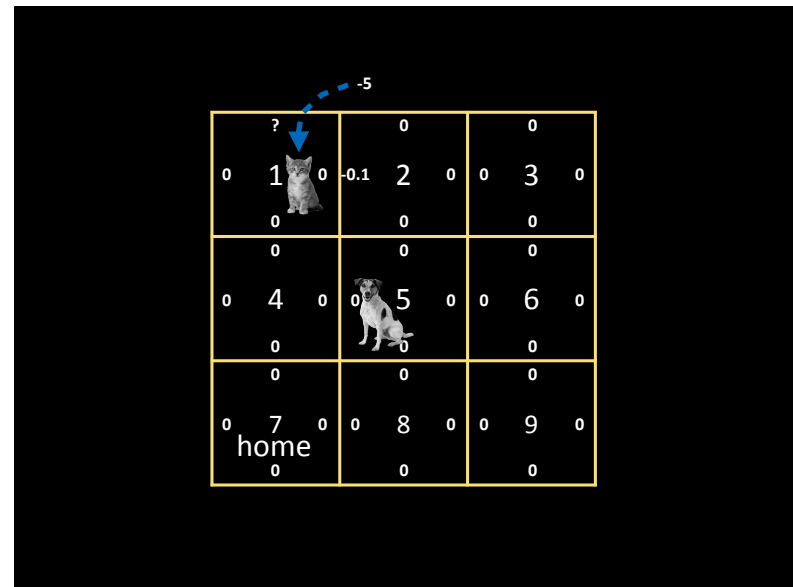
49



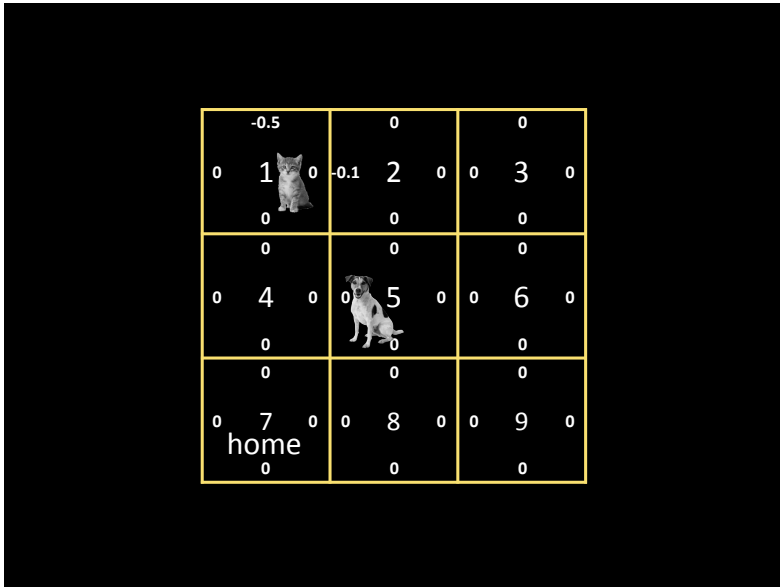
50



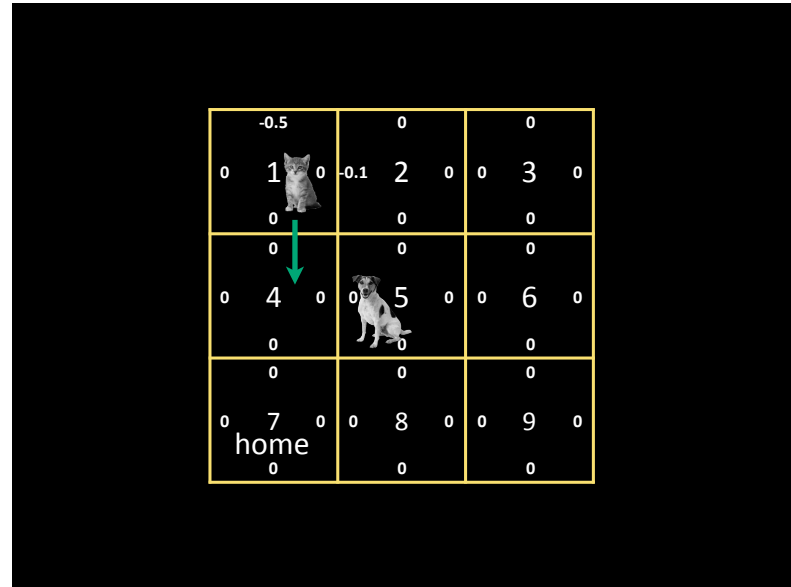
51



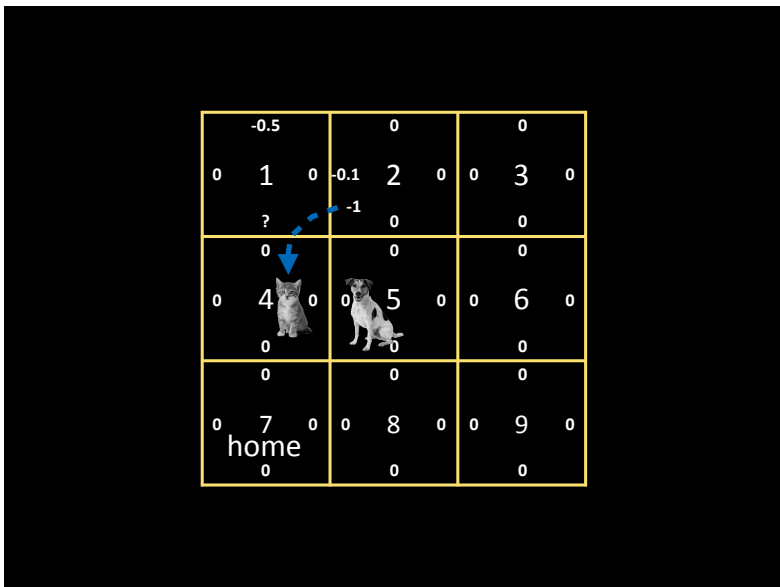
52



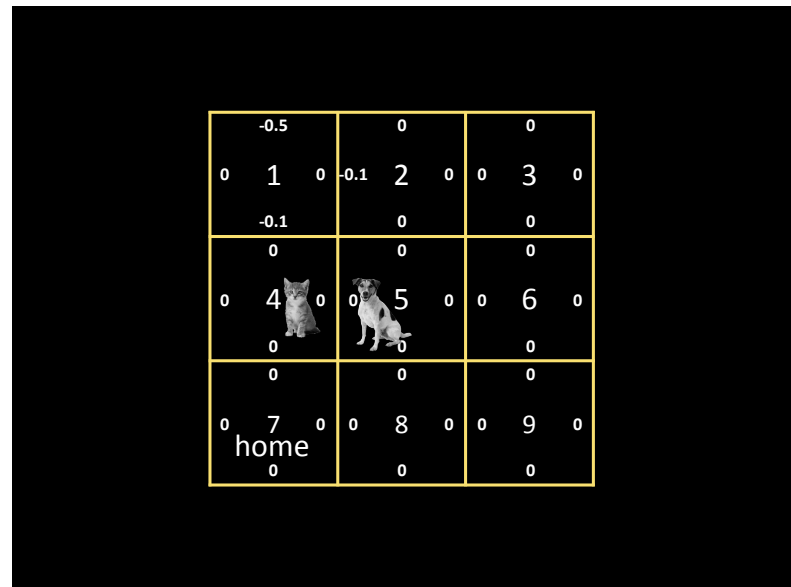
53



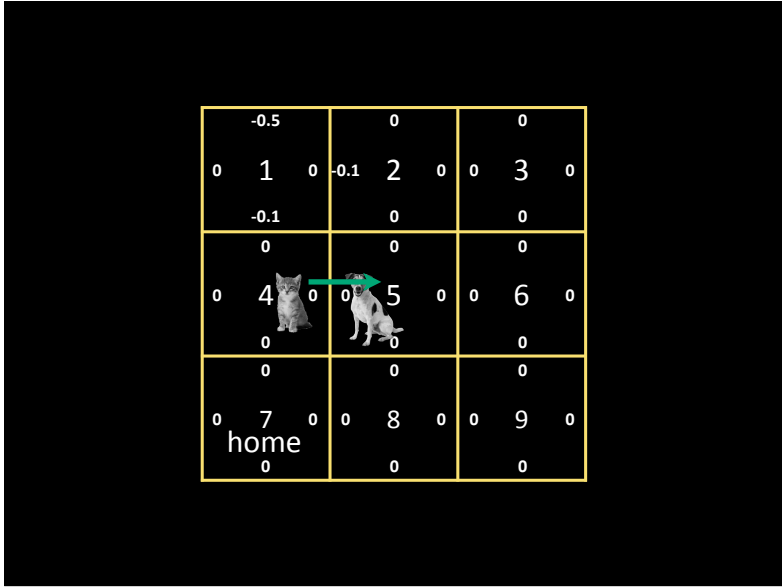
54



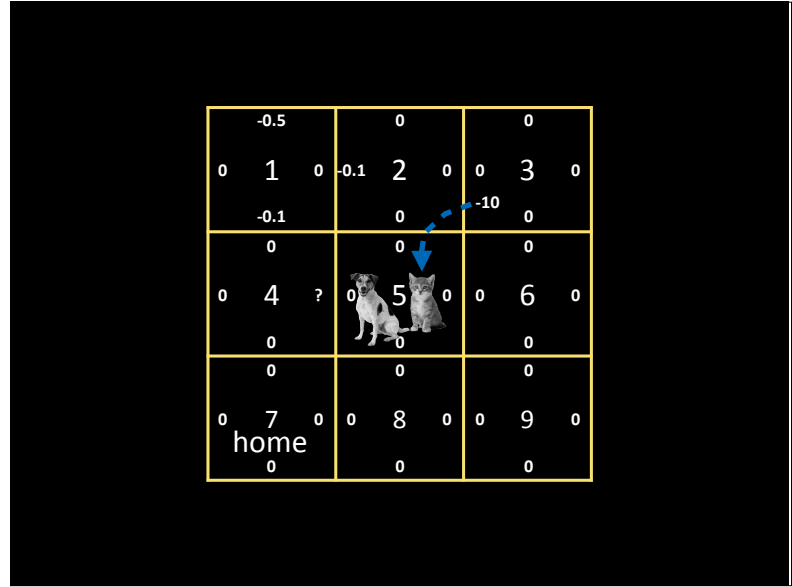
55



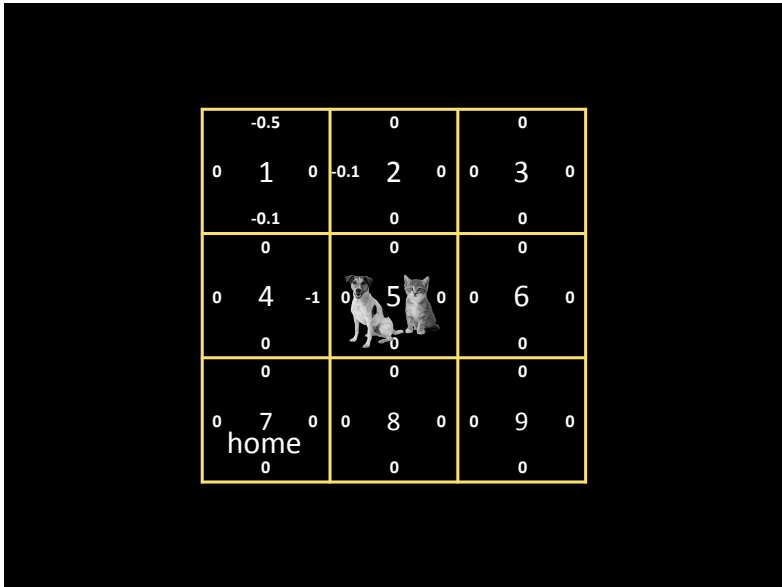
56



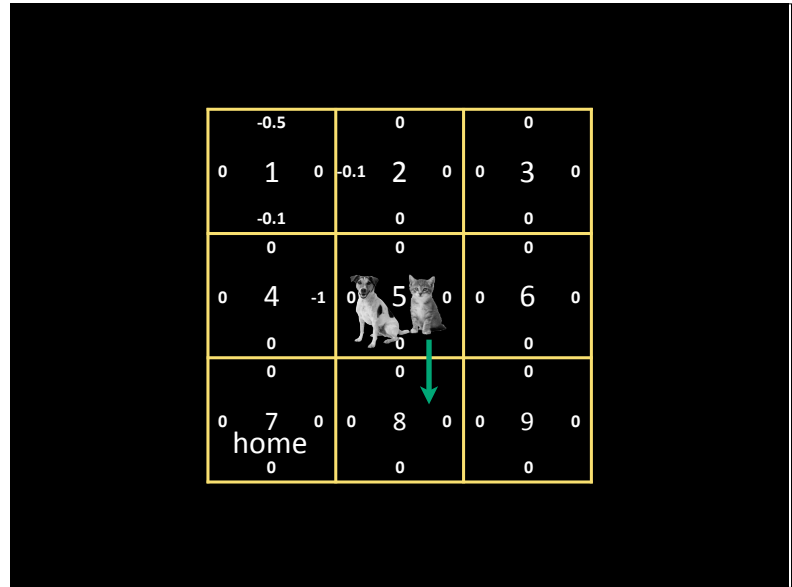
57



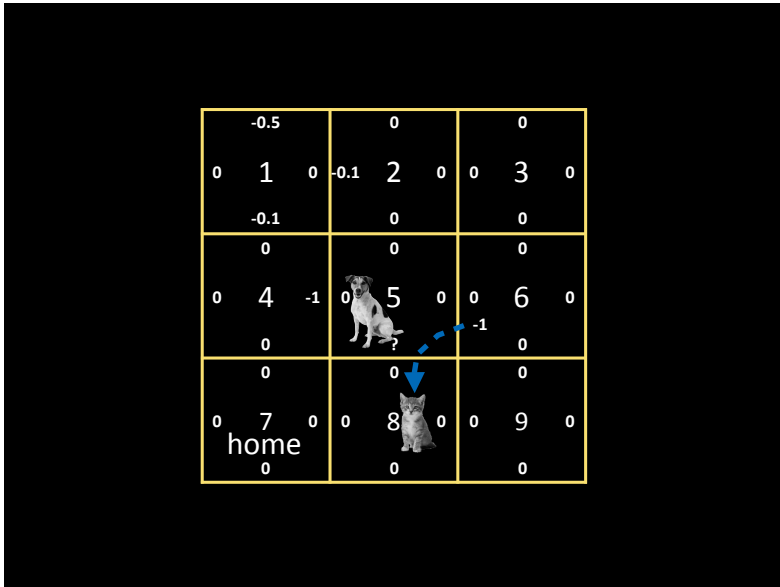
58



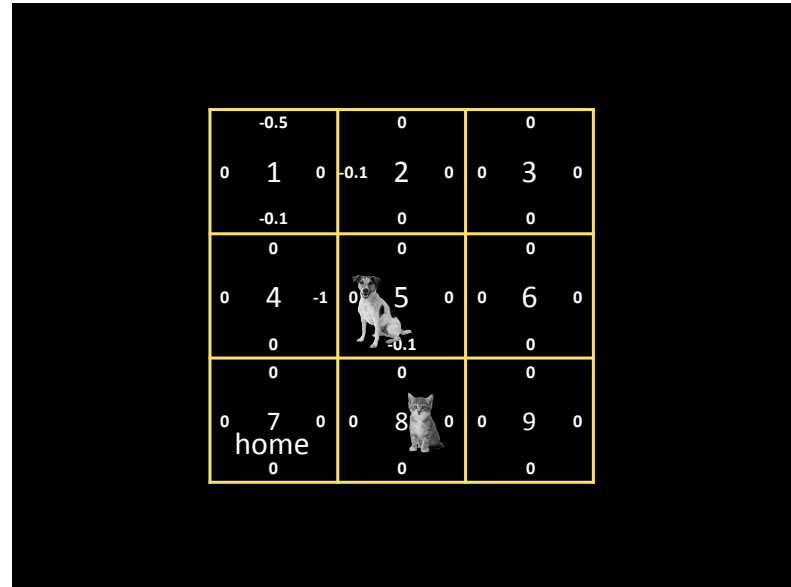
59



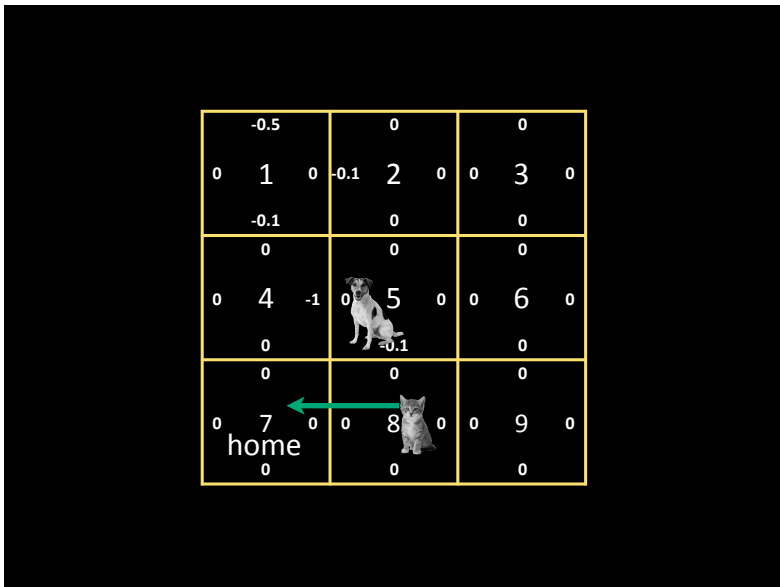
60



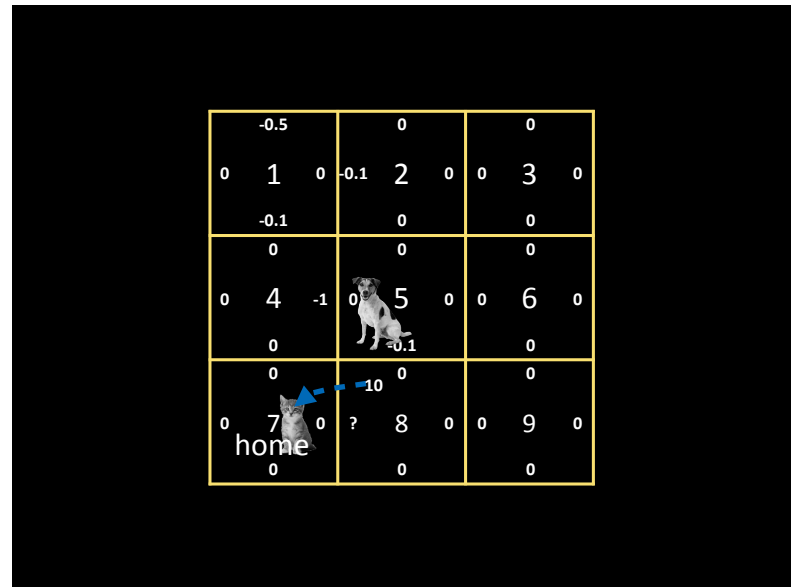
61



62



63



64



-0.5	0	0
0 1 0	-0.1 2 0	0 3 0
-0.1	0	0
0	0	0
0 4 -1	0 5 0	0 6 0
0	-0.1	0
0	0	0
0 7 0	1 8 0	0 9 0
0	0	0

episode 1 ends.

65

let's work out the next episode, starting at state 4

go WEST and then SOUTH

how does the table change?

66

-0.5	0	0
0 1 0	-0.1 2 0	0 3 0
-0.1	0	0
0	0	0
-0.5 4 -1	0 5 0	0 6 0
1	-0.1	0
0	0	0
0 7 0	1 8 0	0 9 0
0	0	0

67

and the next episode, starting at state 3

go WEST -> SOUTH -> WEST -> SOUTH

how does the table change?

68

	-0.5	0	0
0	1	0	-0.1
	-0.1	-1	0
0	0	0	0
-0.5	4	-1	-0.05
	5	0	0
	6	0	0
	1.9	-0.1	0
0	0	0	0
0	7	0	1
	8	0	0
	9	0	0
0	0	0	0

69

what we just saw was  
some episodes of  
**Q-learning**

value updates based on **optimal policy**:  
value of **best next action**

**off-policy learning**

70

**SARSA-learning?**

value updates based on **used policy**:  
value of **the actual next action**

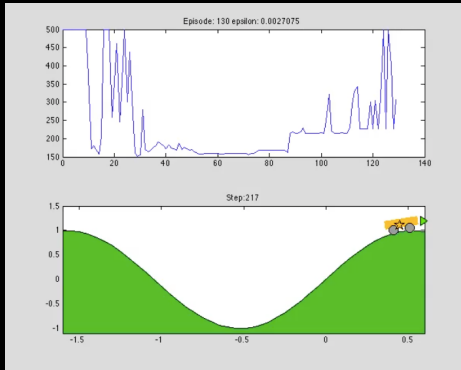
**on-policy learning**

71



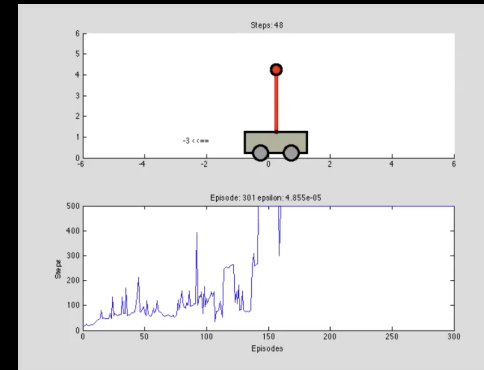
72

## mountain car...



73

## pole balancing...



Pole balancing in reality: <http://www.youtube.com/watch?v=Lc-KLtkDlh8>

74

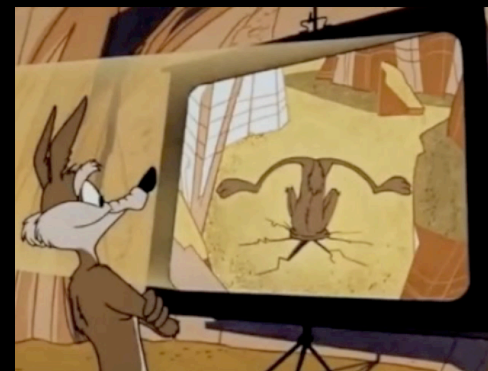
## matlab code for you to play with...

available online for the curious (extremely easy to run):  
[http://jamh-web.appspot.com/  
download.htm#Reinforcement\\_Learning:](http://jamh-web.appspot.com/download.htm#Reinforcement_Learning)

please do e-mail for questions, and if you want to work on  
reinforcement learning research projects:  
[arjun@studix.com](mailto:arjun@studix.com) / [chandra@ifi.uio.no](mailto:chandra@ifi.uio.no)

75

## coyote learning what not to do...



76