Linear Function Approximation

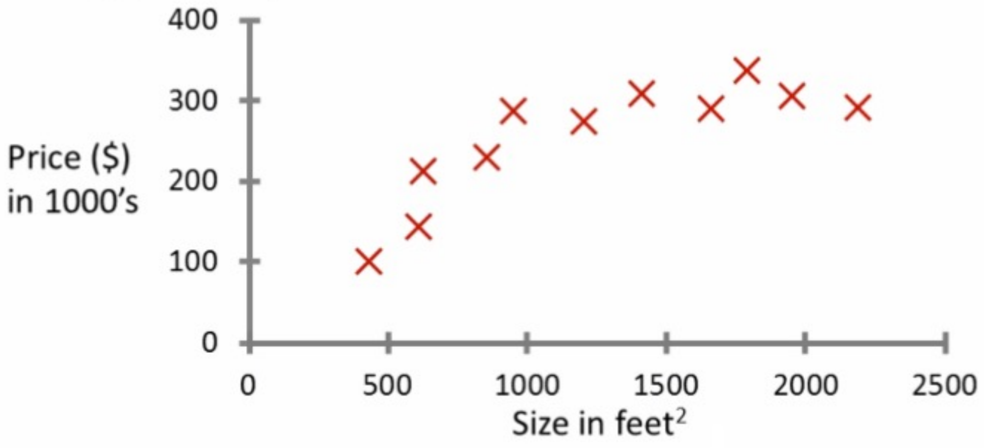
Christopher Mutschler

Value Function Approximation

- Linear Value Function Approximation (careful: non-linear features)

$$\hat{Q}^{\pi}(s, a; w) = \phi(s, a)^T w$$

Housing Price Prediction



Coursera ML Course

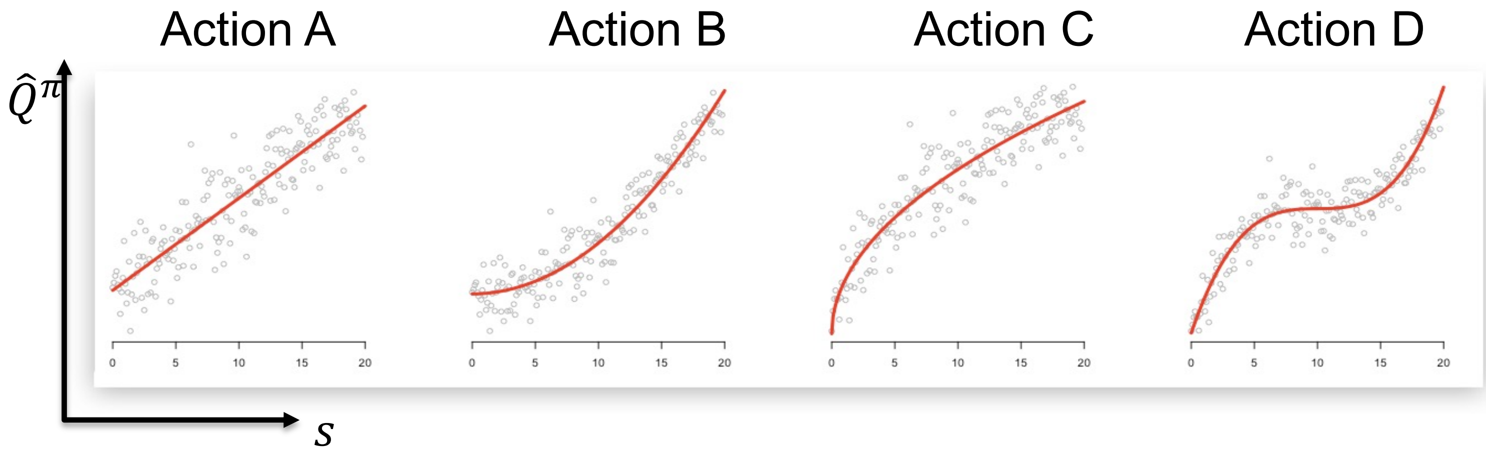
Value Function Approximation

- Linear Value Function Approximation (careful: non-linear features)

$$\hat{Q}^{\pi}(s,a;w) = \phi(s,a)^T w$$

- Example features: Polynomial Basis

$$\phi_i(s,a_l) = \prod_{j=1}^{k} s_j^{c_{i,j}}, \qquad c_{i,j} \in \{0,1,\ldots,n\}, a_l \in \mathcal{A}$$



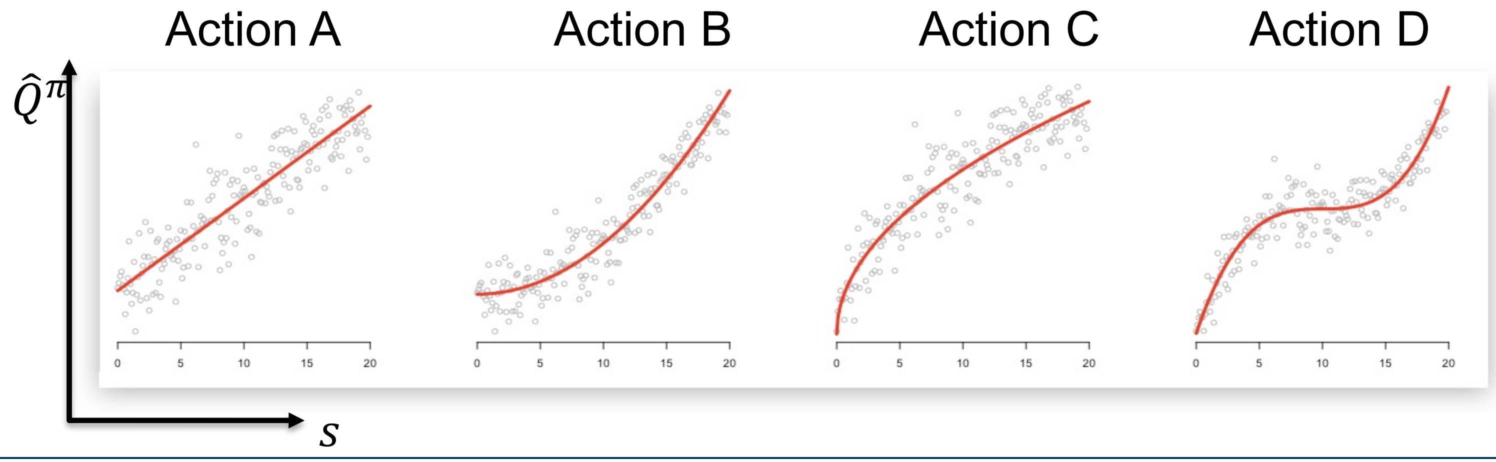
Value Function Approximation

- Linear Value Function Approximation (careful: non-linear features)

$$\hat{Q}^{\pi}(s, a; w) = \phi(s, a)^T w$$

- Example features: Polynomial Basis, for instance:

$$(s_1, s_2)^T \rightarrow (1, s_1, s_2, s_1 s_2)^T$$
$$(s_1, s_2)^T \rightarrow (1, s_1, s_2, s_1 s_2, s_1^2, s_2^2, s_1 s_2^2, s_1^2 s_2, s_1^2 s_2^2)$$



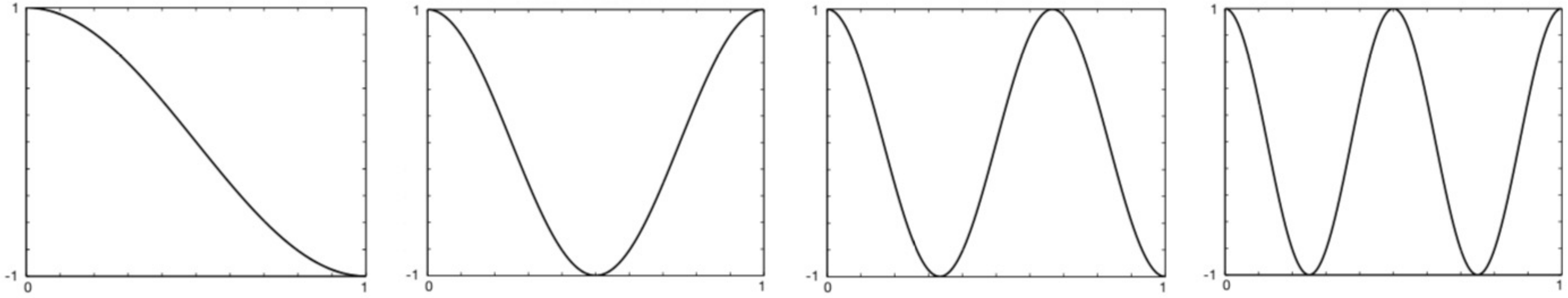
Value Function Approximation

- Linear Value Function Approximation (careful: non-linear features)

$$\hat{Q}^{\pi}(s, a; w) = \phi(s, a)^T w$$

- Example features: Fourier Basis

$$\phi_i(s, a_j) = \cos(i\pi s), \qquad s \in [0,1], a_j \in \mathcal{A}$$



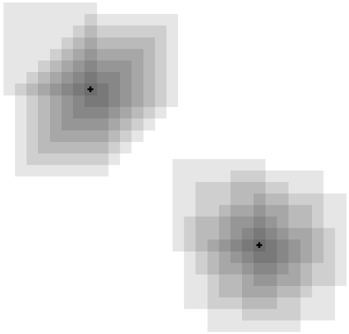
Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. MIT press.

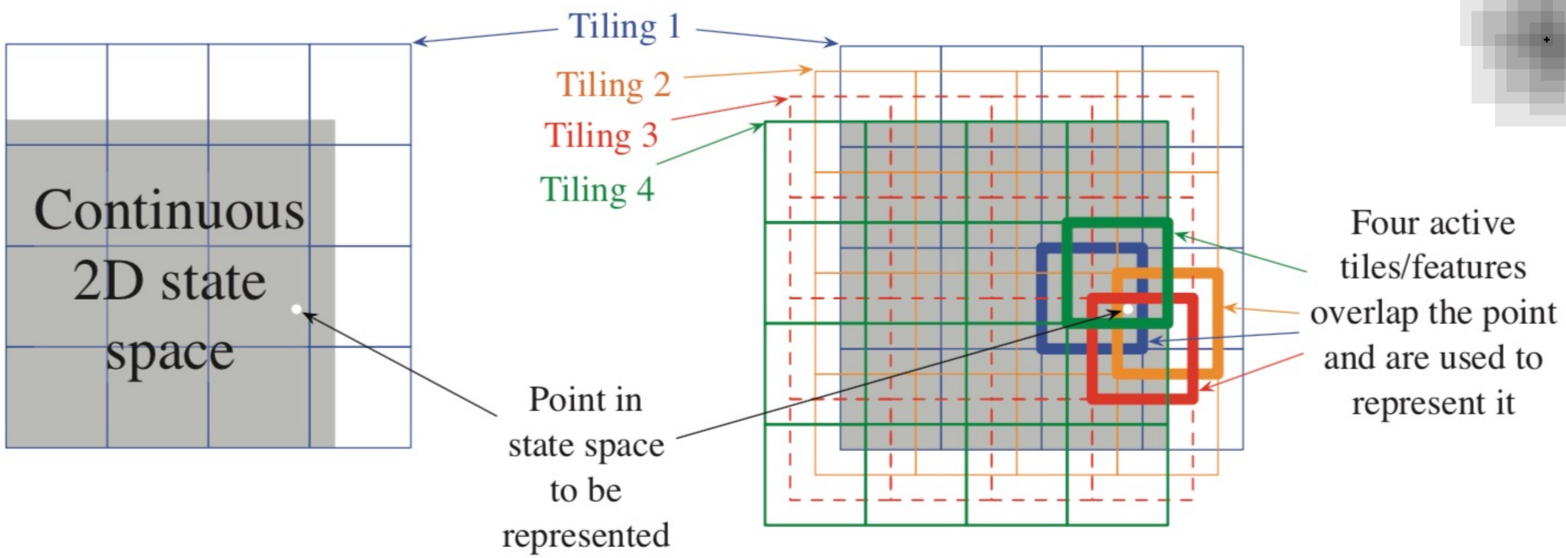
Value Function Approximation

- Linear Value Function Approximation (careful: non-linear features)

$$\hat{Q}^{\pi}(s,a;w) = \phi(s,a)^T w$$

- Example features: Tile Coding

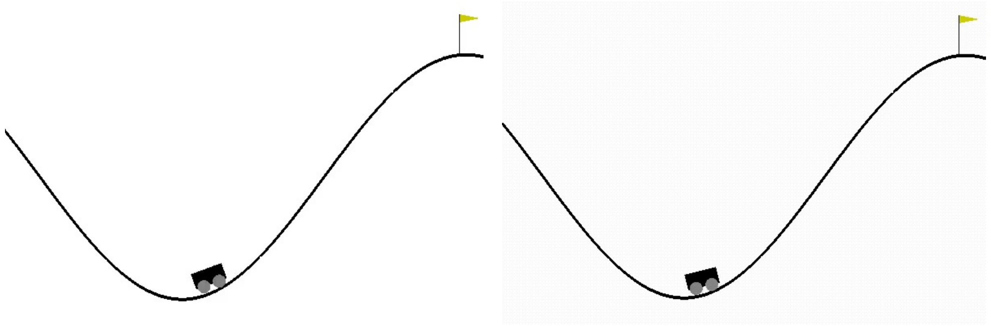


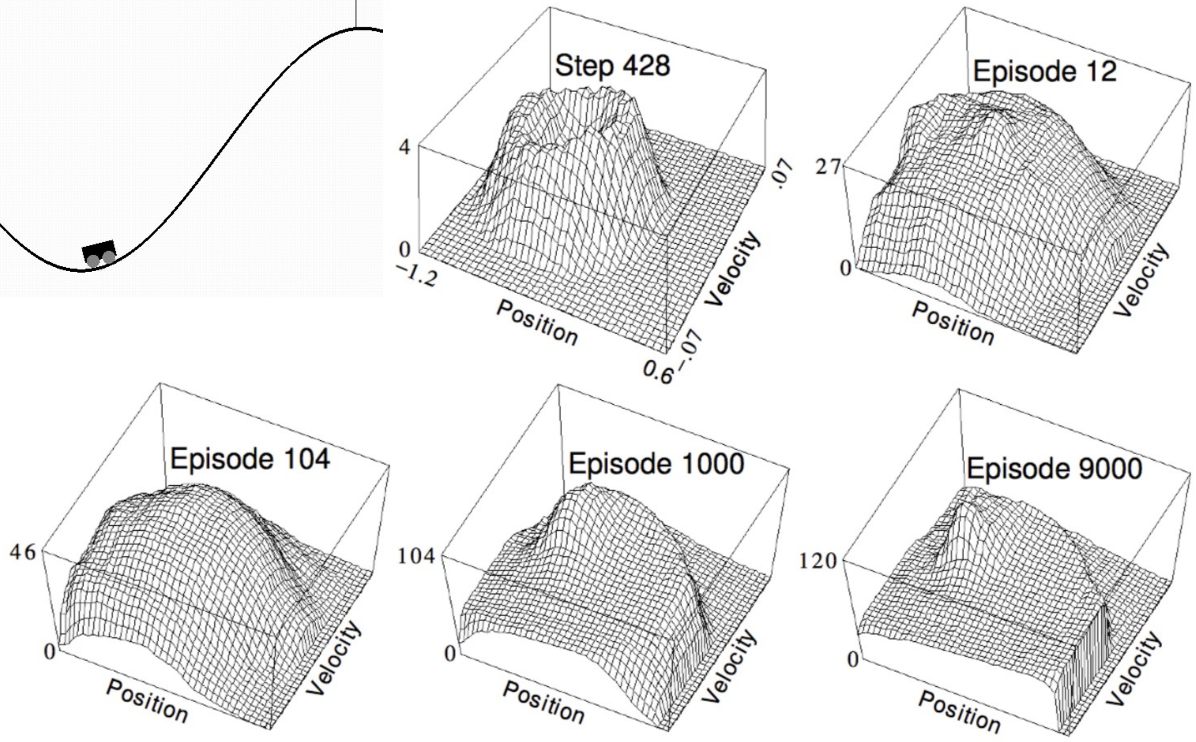


Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. MIT press.

Value Function Approximation

- Linear Sarsa with Coarse Coding[1] in Mountain Car





David Silver. 2016.

[1] *„Tile Coding with circles“*

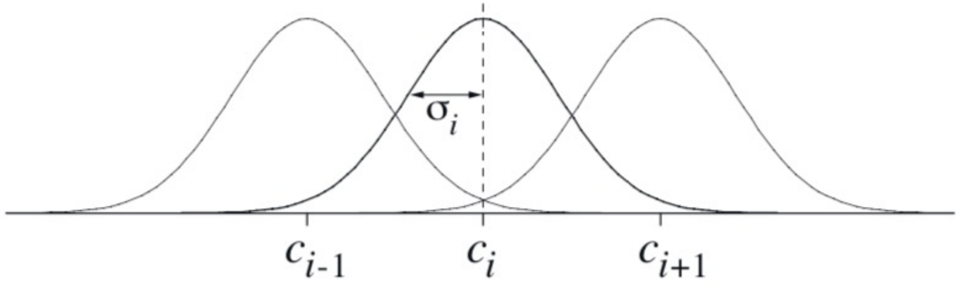
Value Function Approximation

- Linear Value Function Approximation (careful: non-linear features)

$$\hat{Q}^{\pi}(s, a; w) = \phi(s, a)^T w$$

- Example features: Radial Basis Functions (RBFs)

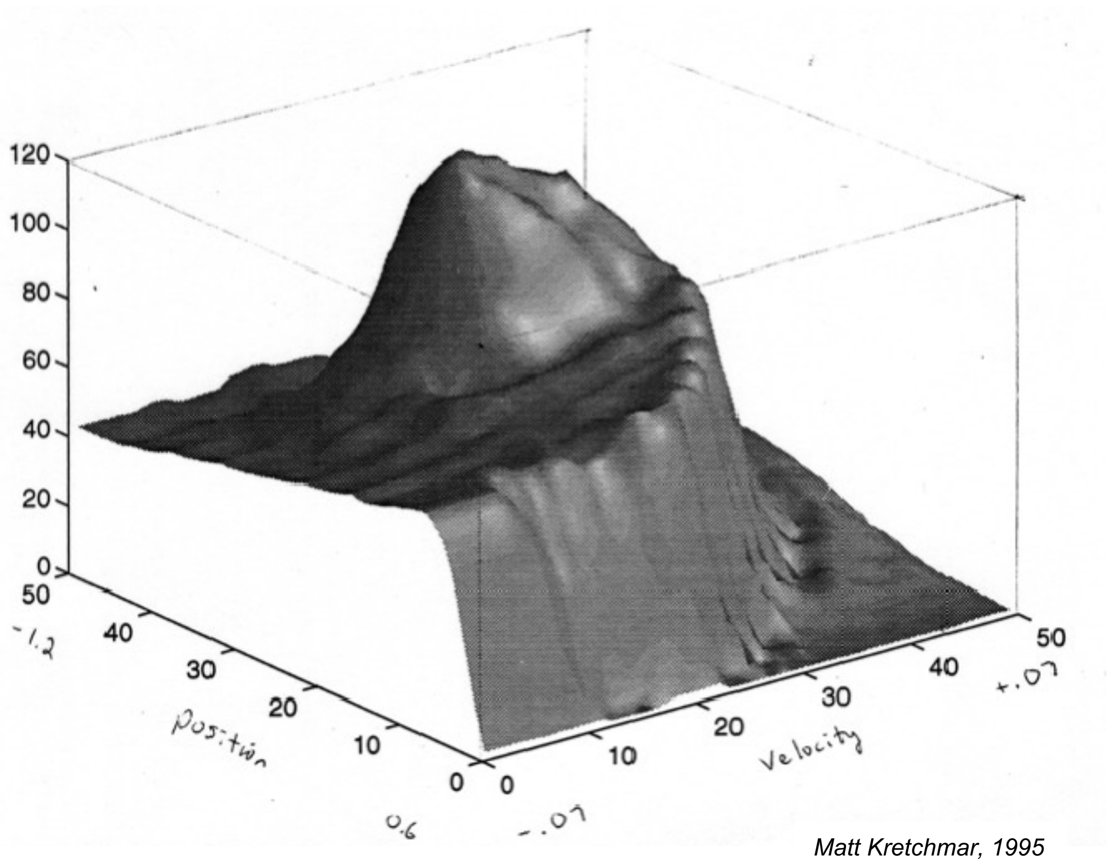
$$\phi_i(s, a_j) \doteq \exp\left(-\frac{\|s - c_i\|^2}{2\sigma_i^2}\right), \qquad a_j \in \mathcal{A}$$



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

Value Function Approximation

- Linear Sarsa with Radial Basis Functions in Mountain Car



Matt Kretchmar, 1995

Value Function Approximation

- Why Linear VFA?
- Convergence of Control Algorithms

Algorithm	Table Lookup	Linear	Non-linear
Monte-Carlo Control	✓	(✓)	✗
SARSA	✓	(✓)	✗
Q-learning	✓	✗	✗

(✓) = chatters around near-optimal value function

Value Function Approximation (VFA)

- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA?

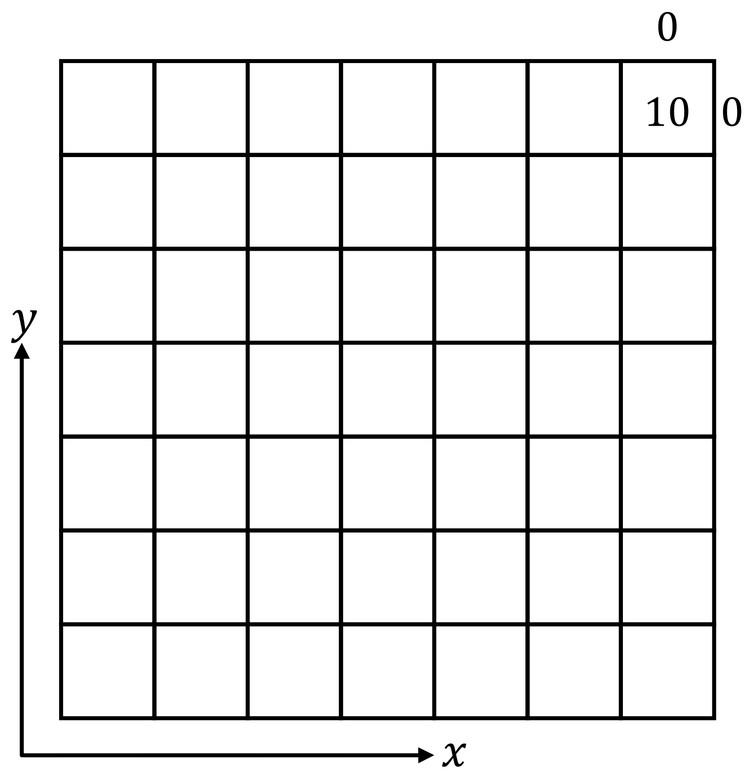
Value Function Approximation (VFA)

- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
     → **YES!** (But the proof is out of the scope of this class…)

Value Function Approximation (VFA)

- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA? → **YES!**

- Example Gridworld problem:
  - No obstacles,
  - deterministic actions (UDLR)
  - no discounting
  - reward is -1 everywhere except $+10$ at goal.



Value Function Approximation (VFA)

- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA? → **YES!**

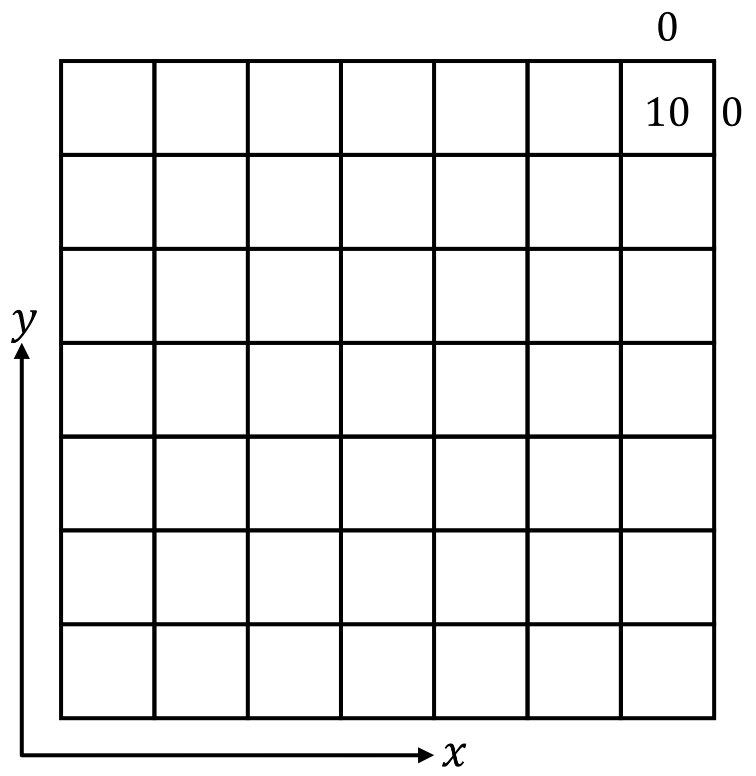
- Represent state s by a feature vector:

$$\phi(s) = \begin{bmatrix} 1 \\ x \\ y \end{bmatrix}$$

- Perform linear VFA:

$$\hat{V}(s; w) = \phi(s)^T w = \begin{bmatrix} 1 & x & y \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix}$$

$$\hat{V}(s; w) = w_0 + w_1 x + w_2 y$$



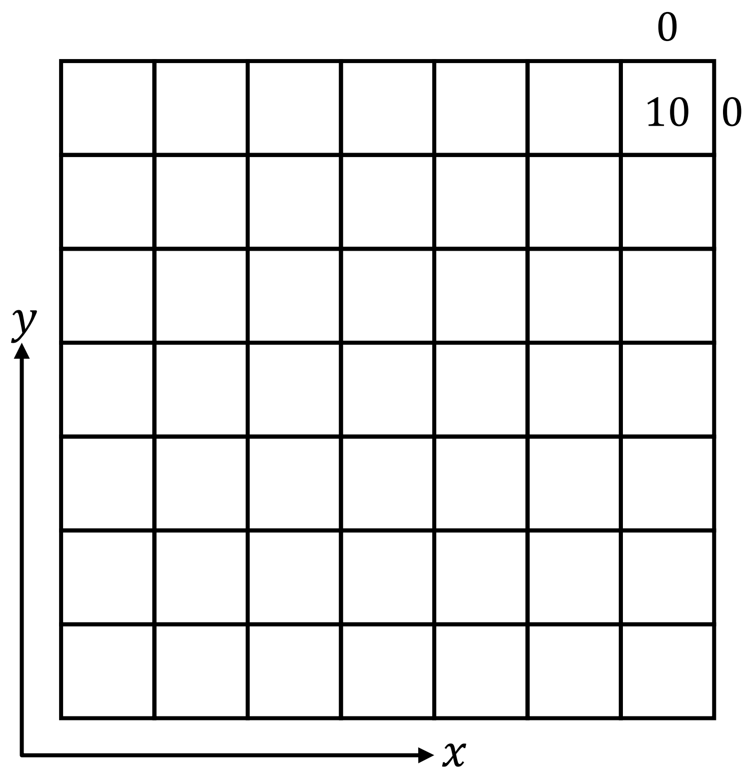
Value Function Approximation (VFA)

- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA? → **YES!**

- Is there a good linear approximation? → **YES**

$$\hat{V}(s;w) = \begin{bmatrix} 1 & x & y \end{bmatrix} \begin{bmatrix} 10 \\ -1 \\ -1 \end{bmatrix}$$
$$= 10 - x - y = 10 - |x + y|$$

- Note: Manhattan Distance.



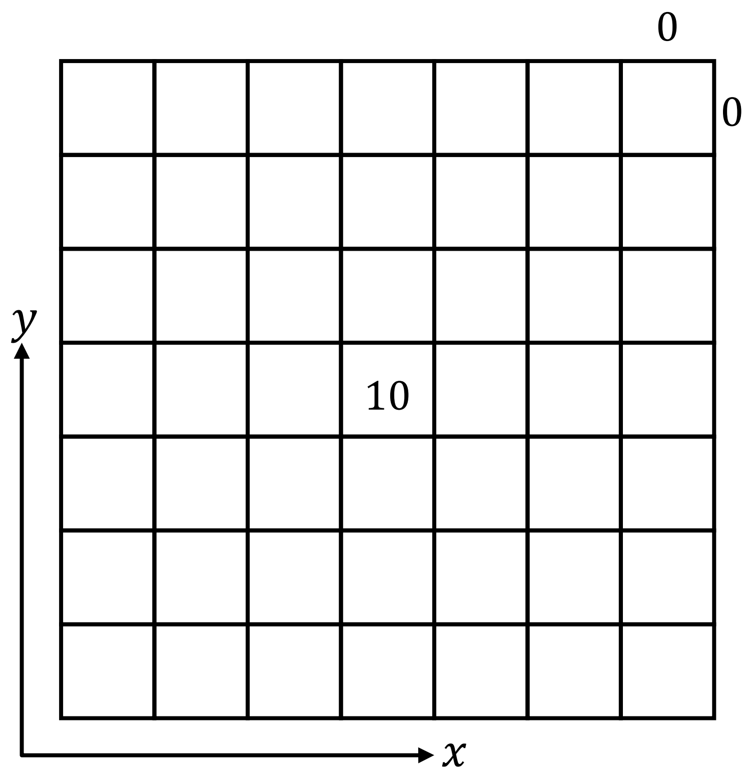
Value Function Approximation (VFA)

- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA? → **YES!**

- What if the reward changes (see Fig.)?
- Linear VFA:

$$\begin{aligned}\hat{V}(s;w) &= \phi(s)^T w \\ &= \begin{bmatrix}1 & x & y\end{bmatrix}\begin{bmatrix}w_0 \\ w_1 \\ w_2\end{bmatrix} \\ &= w_0 + w_1 x + w_2 y\end{aligned}$$

- Is there a good linear approximation?
  → **NO!**



Value Function Approximation (VFA)

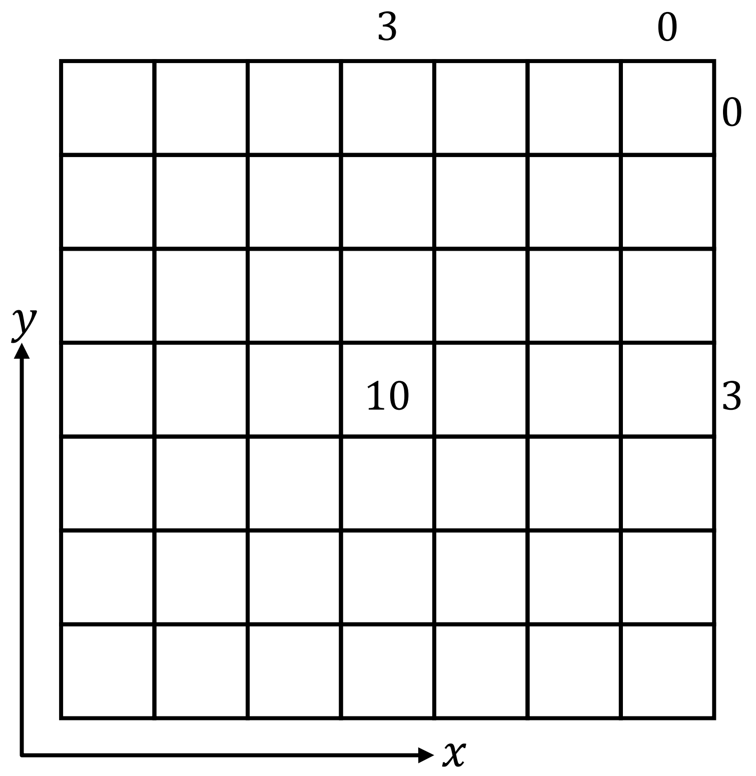
- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA? → **YES!**

- Linear VFA with a new feature $\boldsymbol{z}$:

$$\hat{V}(s; w) = \phi(s)^T w$$
$$= \begin{bmatrix} 1 & x & y & z \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \end{bmatrix}$$
$$= w_0 + w_1 x + w_2 y + w_3 z$$

- Is there a good linear approximation now?
  → **YES!**

$$z = |3 - x| + |3 - y|$$

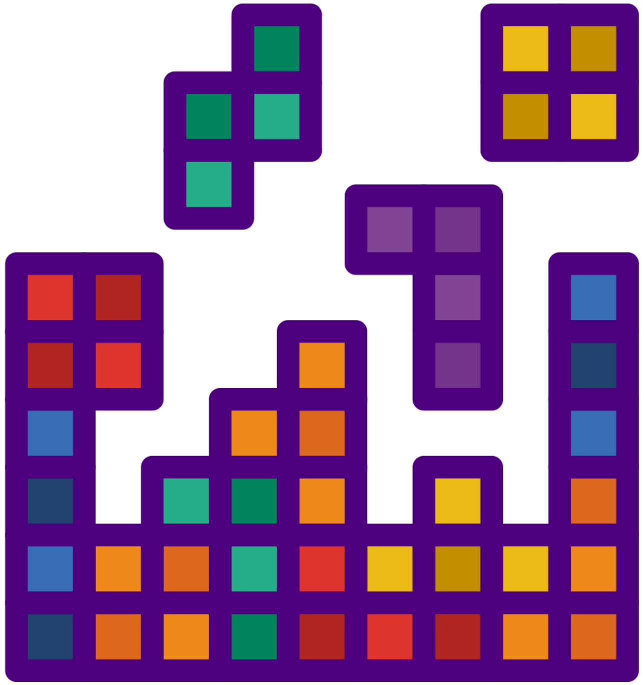


Value Function Approximation (VFA)

- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA? → **NO!**

Feature	Description
Landing Height	Height of last piece is added.
Eroded Piece Cells	#rows eliminated in the last move multiplied with the #bricks eliminated from the last piece added.
Row Transitions	#horizontal full to empty or empty to full transitions between the cells on the board.
Columns Transitions	Same thing for vertical transitions.
Holes	#empty cells covered by at least one full cell.

Thierry, C. and Scherrer, B.: Improvements on Learning Tetris with Cross-Entropy. 2010.



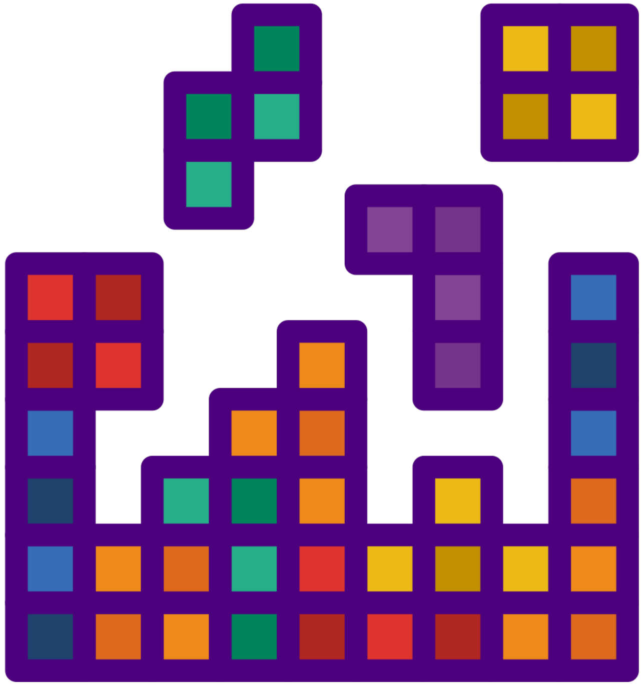
https://www.flaticon.com/free-icon/tetris_1006985

Value Function Approximation (VFA)

- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA? → **NO!**

Feature	Description
Board Wells	$\sum_{w \in wells} (1 + 2 + \cdots + depth(w))$
Column Height	Height of the pth column of the board.
Column Difference	Absolute difference $\lvert h_p - h_{p+1} \rvert$ between adjacent columns.
Maximum Height	Maximum pile height: $\max_p h_p$.
Holes	Number of empty cells covered by at least one full cell.

Thierry, C. and Scherrer, B.: Improvements on Learning Tetris with Cross-Entropy. 2010.

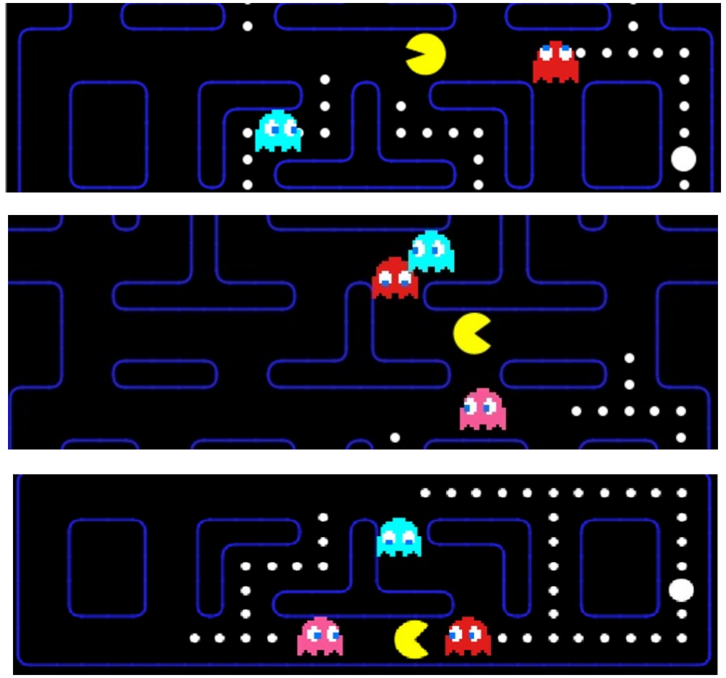


https://www.flaticon.com/free-icon/tetris_1006985

Value Function Approximation (VFA)

- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA? → **NO!**

Feature	Description
Nearby Walls	Existence of wall to all four wind directions.
Nearest Target	Direction of the nearest *target* where it is preferable for the Ms. Pac-Man to move.
Nearby Ghosts	Existence of a ghost to all four wind directions.
Existance of Escape	Describes if Ms. Pac-Man can move freely or if she is trapped.



Tziortziotis, N. and Tziortziotis, K. and Blekas, K.: Play Ms. Pac-Man Using an Advanced Reinforcement Learning Agent. 2014.

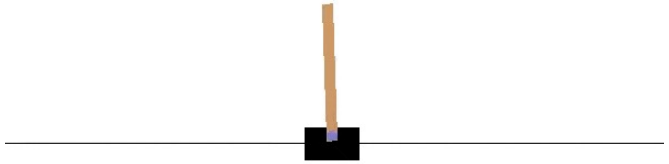
Value Function Approximation (VFA)

- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA? → **NO!**

We applied LSPI with a set of 10 basis functions for each of the 3 actions, thus a total of 30 basis functions, to approximate the value function. These 10 basis functions included a constant term and 9 radial basis functions (Gaussians) arranged in a 3×3 grid over the 2-dimensional state space. In particular, for some state $s = (\theta, \dot{\theta})$ and some action a, all basis functions were zero, except the corresponding active block for action a which was

$$\left(1,\ e^{-\frac{\|s-\mu_1\|^2}{2\sigma^2}},\ e^{-\frac{\|s-\mu_2\|^2}{2\sigma^2}},\ e^{-\frac{\|s-\mu_3\|^2}{2\sigma^2}},\ \dots,\ e^{-\frac{\|s-\mu_9\|^2}{2\sigma^2}}\right)^{\intercal},$$

where the μ_i's are the 9 points of the grid $\{-\pi/4,\ 0,\ +\pi/4\} \times \{-1,\ 0,\ +1\}$ and $\sigma^2 = 1$.



Lagoudakis, M. G. and Parr, R.: Least-Squares Policy Iteration. JMLR. 2003

Value Function Approximation (VFA)

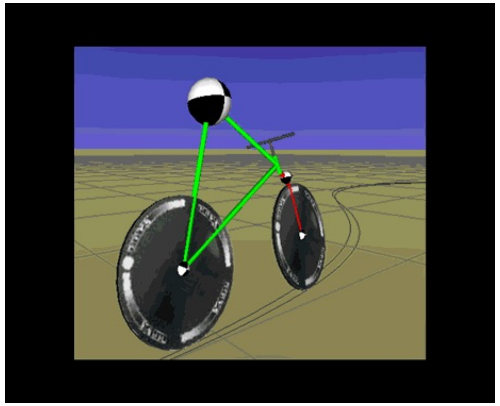
- There are two important questions to answer with VFA:
  1. Can we approximate any V-/Q-value function with a linear FA?
  2. Is it easy to find such a linear FA? → **NO!**

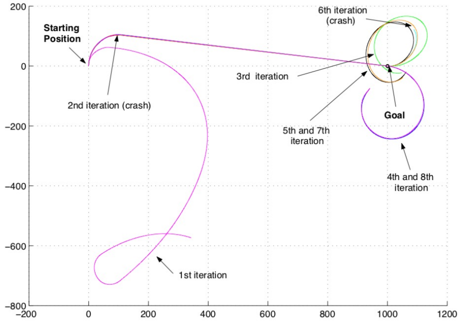
The goal in the *bicycle balancing and riding* problem (Randløv and Alstrøm, 1998) is to learn to balance and ride a bicycle to a target position located 1 km away from the starting location. Initially, the bicycle's orientation is at an angle of 90° to the goal. The state description is a six-dimensional real-valued vector $(\theta, \dot{\theta}, \omega, \dot{\omega}, \ddot{\omega}, \psi)$, where θ is the angle of the handlebar, ω is the vertical angle of the bicycle, and ψ is the angle of the bicycle to the goal. The actions are the torque τ applied to the handlebar (discretized to $\{-2, 0, +2\}$) and the displacement of the rider υ (discretized to $\{-0.02, 0, +0.02\}$). In our experiments, actions are restricted so that either $\tau = 0$ or $\upsilon = 0$ giving a total of 5 actions.[12] The noise in the system is a uniformly distributed term in $[-0.02, +0.02]$ added to the displacement component of the action. The dynamics of the bicycle are based on the model of Randløv and Alstrøm (1998) and the time step of the simulation is set to 0.01 seconds.

The state-action value function $Q(s, a)$ for a fixed action a is approximated by a linear combination of 20 basis functions:

$$\left(1,\ \omega,\ \dot{\omega},\ \omega^2,\ \dot{\omega}^2, \omega\dot{\omega},\ \theta,\ \dot{\theta},\ \theta^2,\ \dot{\theta}^2,\ \theta\dot{\theta},\ \omega\theta,\ \omega\theta^2,\ \omega^2\theta,\ \psi,\ \psi^2,\ \psi\theta,\ \bar{\psi},\ \bar{\psi}^2,\ \bar{\psi}\theta \right)^{\mathsf{T}},$$

where $\bar{\psi} = \pi - \psi$ for $\psi > 0$ and $\bar{\psi} = -\pi - \psi$ for $\psi < 0$. Note that the state variable $\ddot{\omega}$ is completely ignored. This block of basis functions is repeated for each of the 5 actions, giving a total of 100 basis functions (and parameters).





Lagoudakis, M. G. and Parr, R.: Least-Squares Policy Iteration. JMLR. 2003

Value Function Approximation (VFA)

- Idea: Why don‘t we replace linear approximation with NNs?
  - Because theory tells us that this doesn‘t work out

Algorithm	Table Lookup	Linear	Non-linear
Monte-Carlo Control	✓	(✓)	✗
SARSA	✓	(✓)	✗
Q-learning	✓	✗	✗

(✓) =chatters around near-optimal value function

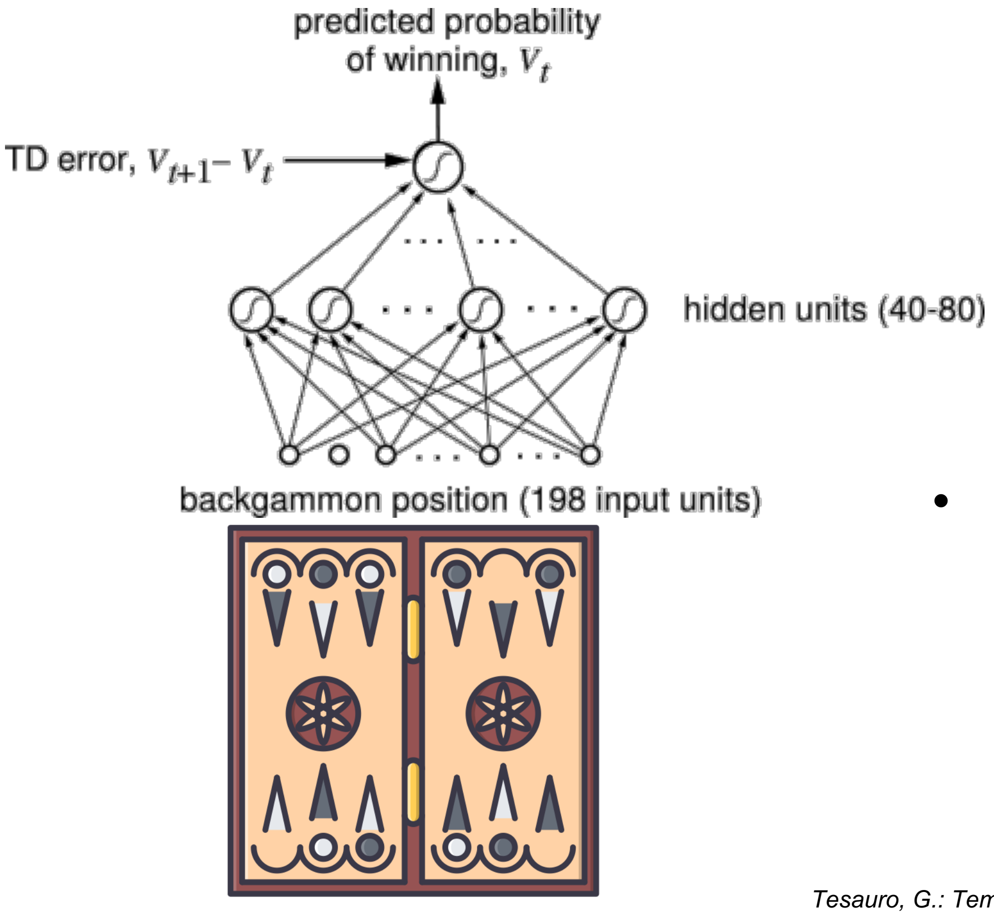
Value Function Approximation (VFA)

- Idea: Why don‘t we replace linear approximation with NNs?
  - Because theory tells us that this doesn‘t work out
  - But in some applications, it did ☺!
- World‘s Best Backgammon Player:
  - Neural network (NN) with 80 hidden units
  - Used RL for 300.000 games of self-play
  - One of the top players in the world!

https://www.flaticon.com/free-icon/backgammon_683899

Value Function Approximation (VFA)

- Parenthesis: Gerry Tesauro‘s TD-Gammon (‘92, ’94, ’95)



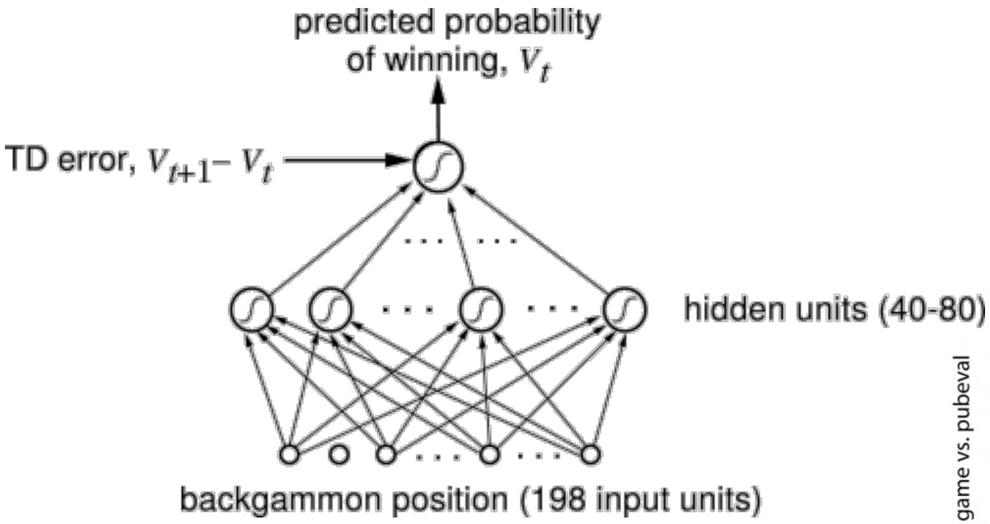
- Rule to update the network weights w:

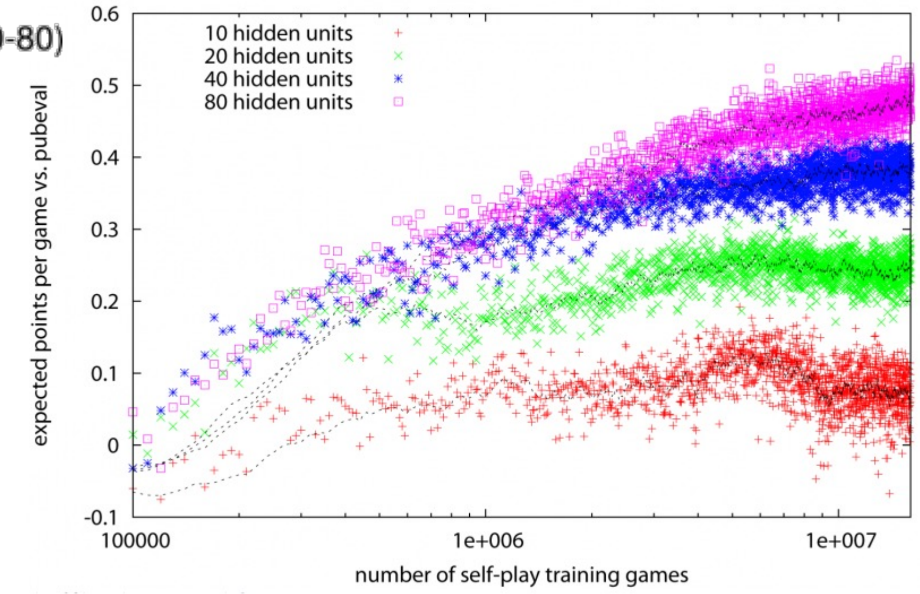
$$w \leftarrow w + \alpha(V_{t+1} - V_t) \sum_{k=1}^{t} \lambda^{t-k} \nabla_w V_k$$

Tesauro, G.: Temporal difference learning and TD-Gammon. 1995.
see also: https://users.auth.gr/kehagiat/Research/GameTheory/12CombBiblio/BackGammon.html

Value Function Approximation (VFA)

- Parenthesis: Gerry Tesauro‘s TD-Gammon (‘92, ’94, ’95)





Tesauro, G.: Temporal difference learning and TD-Gammon. 1995.
see also: https://users.auth.gr/kehagiat/Research/GameTheory/12CombBiblio/BackGammon.html