No HW this week so...

- Start thinking about whether any questions you’ve had could be addressed in final projects (tentatively)
- Get started with Emergent tutorials: build your own network etc.
- Linked on class website: http://ski.cog.brown.edu/cogsim.html

Sequential & Temporally-Delayed Learning

1. The Problem.

Error-driven + Hebbian: Solve tasks, learn systematic representations, generalize to new stimuli.

What’s left?...

Time!

Currently: networks learn immediate consequence of a given input.

• What if current input only makes sense as part of a sequence (e.g., language, social interactions)?
• What if the consequence of this input comes later (e.g., school/work)?

Representing Context for Sequence Learning

How do we do it?

We represent the context, not just the current input.

For example:

- The girl picked up the pen.
- The pig raced around the pen.
- My favorite color is purple.

We represent the context, not just the current input.

How does the brain do it?

How would we get our models to do it?

Add layers to keep track of context (prefrontal cortex; hippocampus...).

An Example Task

BTXSE
BPVPSE
BTSXXTVVE
BPTVPSE
BSSXSE

Which of the following sequences are allowed?:

- BTXXTTVVE
- TSXSE
- VVSXE
- BSSXSE

What if the consequence of the input comes later?

Impulse control; social interactions?

Current reward: immediate consequence of a given input.

Time

What if...

Representation generation to new stimuli?

Forward + Feedback: social basis, learn something...
An Example Task

BTXSE, BPVPSE, BTSXXTVVE, BPTVPSE, BTXXTTVVE

TSXSE, VVSXE, BSSXSE

We implicitly learn such grammars (e.g., pressing buttons faster to letters that follow grammar).

Time & Sequences

Currently: networks learn immediate consequence; a given input

Next:

What if the consequence of a single input comes later in time (next extended sequence of inputs (context)?

What if current input only makes sense as part of a sequence?

Extended: networks learn immediate consequence of a given input

We implicitly learn such grammars (e.g., pressing buttons faster to letters that follow grammar).

BXXE, BYXE, BEXXE, BXXE, BXXE
Why Copy the Hidden Representation?

Simple Recurrent Network (SRN):

- This hidden representation serves to disambiguate the input.
- Captures information from the previous state.
- Captures forward information by means of a context layer.

Simple Recurrent Network Summary:
Why Copy the Hidden Representation?

- Just copying last hidden state at time t-1 is analogous to a stack, but doesn’t allow the network to revisit past items.
- Copying the hidden layer lets the network hold on to an arbitrarily large number of items – even though it only ever looks at the last hidden state at each time step.
- The network learns how strongly to hold on to past items.
Simple SRN story is not flawless

• How is hidden → "copy" function implemented biologically?

• During settling, context must be actively maintained (ongoing hidden activity has no effect on context).

• Assumes all context is relevant: What if distracting information presented in middle of sequence? Want to only hold on to relevant context.

→ Stay tuned for specialized biological/computational mechanisms for updating / gating vs. robust maintenance of context.

Motivating Motivation

Why does anyone go to grad school? (or, why do we ever do anything besides eat, sleep, have sex, etc)?

e.g., Why am I here today, instead of lying on a beach in Mexico, drinking mojitos and reading a good book?

Challenge: make a neural network model that decides to go to grad school!

The Motivational Bootstrap

Puzzle: Motivations must be built-in (else we would likely die)

But if purely fixed, there would be no art, science, etc.

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Need a motivational bootstrap: Learning on top of built-ins.

Puzzle: Motivations must be built-in (else we would die).

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.

Puzzle: Motivations must be built-in (else we would die).

Need a motivational bootstrap: Learning on top of built-ins.

Culture & social drives provide cumulative shaping of learning.
Temporally-delayed Learning & Reinforcement

Reinforcement often delayed from the action(s) that lead to it: need to "span the gap".

Key ideas: We want to predict rewards consistently over time. This process leads us to learn what events are associated with rewards, earlier and earlier back in time.

We use the Temporal Differences (TD) algorithm (Sutton & Barto, Dayan, Montague, ...).

Reinforcement Biology (somewhat outdated...)

Striasomes
Matrisomes
a) Basal Ganglia
b) Frontal Cortex

Midbrain areas send dopamine (DA) to modulate cortex & basal ganglia: Substantia Nigra (SN) & Ventral-Tegmental Area (VTA).

SN & VTA are controlled by specific cortical & BG areas. These other areas are like an "Adaptive Critic" (AC), which evaluates stimuli & actions for their rewarding value. The critic can then train an "Actor" system to execute rewarding actions.

- SN & VTA are connected by specific cortical BG areas
- These areas are like an "Adaptive Critic" (AC), which evaluates stimuli & actions for their rewarding value.
- The critic can then train an "Actor" system to execute rewarding actions.

The Actor-Critic Framework

Actor (executes actions)
Critic (evaluates actions)

VTA dopamine neuron recordings
VTA firing moves from responding to reward to anticipating it at the instruction.

Basic Data: VTA DA Neural Firing in Conditioning

Before: After:
Temporal Difference Learning: Equations

Value function, sum of discounted future rewards:

\[ V(t) = \langle \gamma^0 r(t) + \gamma^1 r(t+1) + \gamma^2 r(t+2) \ldots \rangle \] (1)

Recursive definition:

\[ V(t) = \langle r(t) + \gamma V(t+1) \rangle \] (2)

Error in predicted reward (from previous to next time-step):

\[ \delta(t) = (r(t) + \gamma \hat{V}(t+1)) - \hat{V}(t) \] (3)

Update value estimate:

\[ \hat{V}(t) \leftarrow \hat{V}(t) + \alpha \delta(t) \] (4)

\[ \alpha = \text{learning rate} \]

TD and Dopamine Relationship

Schultz, Dayan & Montague, 1996, Science

Model: CS at t=2, US at t=16

a) TD Error

\[ \begin{array}{ccccccccc}
0 & -0.2 & -0.4 & -0.6 & -0.8 & -1 & 0 & 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 20 \\
\end{array} \]

Time

b) TD Error

\[ \begin{array}{ccccccccc}
0 & -0.2 & -0.4 & -0.6 & -0.8 & -1 & 0 & 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 20 \\
\end{array} \]

Time

c) TD Error

\[ \begin{array}{ccccccccc}
0 & 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 20 \\
\end{array} \]

Time

Network Implementation

\[ \begin{align*}
\text{Stimuli} & \quad \rightarrow \quad \text{Hidden} \\
V(t) & \quad \leftarrow \quad V(t+1) + r(t) + \gamma \delta(t)
\end{align*} \]

Phase-based Implementation

\[ \begin{align*}
\text{Stimulus} & \quad \rightarrow \quad \text{Hidden} \\
V(1) & \quad \leftarrow \quad V(2) + \gamma V(1) + r(1) + \gamma \delta(1) \\
V(2) & \quad \leftarrow \quad V(3) + \gamma V(2) + r(2) + \gamma \delta(2) \\
\end{align*} \]

TDRewInteg = TDRewPred + ExtRew

\[ \text{Minus phase: TDRewInteg clamped to prev plus phase value.} \]

\[ \text{Plus phase: TDRewInteg settles via weights} \]

Learning signal \( \delta \) trains prediction for previous time step. (eligibility traces needed)

Exploration: \[ \text{rl cond.proj} \]

\begin{align*}
\text{Input} & \quad \rightarrow \quad \text{Hidden} \\
\text{time} & \quad \rightarrow \quad \text{Hidden} \\
\text{tone} & \quad \rightarrow \quad \text{Hidden} \\
\text{light} & \quad \rightarrow \quad \text{Hidden} \\
\text{odor} & \quad \rightarrow \quad \text{Hidden}
\end{align*} \]

Complete Serial Compound (CSC) input representation

\[ \text{Not realistic, but good for demonstration. This assumption can be relaxed without changing core ideas (e.g. Ludvig et al, 2008).} \]

\[ \text{This approach is widely used in Sutton & Barto, Montague et al, etc} \]

\[ \text{Used in combination with dopamine, etc.} \]

Value function sum of discounted future rewards

\[ \text{Update value estimate:} \]

\[ \hat{V}(t) \leftarrow \hat{V}(t) + \alpha \delta(t) \] (4)

\[ \alpha = \text{learning rate} \]
Exploration: \[ r_i \text{proj} \]

Standard TD:
\[
\hat{V}(t) = \sum_i w_i x_i(t) \quad [x_i \text{are inputs: tone, light}]
\]

Here: passed thru activation function – has to surpass threshold, subject to inhibitory competition from other value reps.

DA and Timing: Late and Early Rewards

Learning Theory: Blocking (Behavior)

Bao et al, 2001, Nature

Learning Theory: Blocking (Dopamine)

Waelti et al, 2001, Nature

Blocked stimulus Control (not blocked) stim

Ventral striatum = DA enriched, correlates with TD PE=Critical

Dorsal str only activated when Actor involved (instrumental)

DA and Learning: Auditory Cortex

O'Doherty et al, 2004, Science

TD prediction error and human functional imaging

Exploration (1, cond proj)
The Basal Ganglia and Action Selection

• BG selectively facilitates one command while suppressing others (Mink, 1996; Frank et al, 2001; Gurney et al, 2001; Brown et al, 2004)
• Occurs in parallel for multiple BG-frontal circuits: from motor to cognitive actions (Frank, 2005; O’Reilly & Frank, 2006; Houk, 2005)

Reinforcement Learning and Dopamine

Satoh et al (2003): DA bursts/dips depending on monkey’s action (not just Pavlovian)

Burst/Pause correlations with Rew Prediction Errors

Bayer et al, 2007, JNeurophys

Optical phasic DA stimulation causally induce conditioning (Tsai et al, 2009, Science)
DA neuron spiking during reinforcement task in humans (Zaghoul et al., 2009, Science)

Striato-Cortical Functional Circuitry

Response1 Response2

GPe

D1

D2

excitatory
inhibitory
modulatory

striatum

SNc

thalamus

pre/motor cortex

Go NoGo

STN

SNr/GPi

excitatory
inhibitory
modulatory

pre/motor cortex

tonically active

SNr/GPi

Go

“Disinhibition”

Direct

Indirect thalamus

SNr/GPi
Striato-Cortical Functional Circuitry

- GPe
- D1
- D2
- excitatory
- inhibitory
- modulatory
- striatum
- SNc
- Indirect thalamus
- pre/motor cortex
- NoGo
- Direct
- Go
- SNr/GPi

DA Feedback Effects on BG Learning

- SNc
- GPe
- D2
- excitatory
- inhibitory
- modulatory
- striatum
- Indirect
- Direct thalamus
- pre/motor cortex
- D1
- NoGo
- Go
- LTP
- LTD
- SNr/GPi

DA effects on BG Learning: Positive PE


Neural Model of BG and DA

Integrates a wide range of physiological data into a single coherent framework; can then explore dynamics in learning & decision making

\[
\begin{align*}
\langle \beta_e \rangle &= \langle \beta_w \rangle = \langle \beta_m \rangle = \langle \beta_{\text{syn}} \rangle = \langle \beta_{\text{inh}} \rangle = \langle \beta_{\text{out}} \rangle \\
\gamma_e &= \gamma_w \\
\sigma_{\text{exc}} &= \sigma_{\text{inh}} \\
\Delta w_{ij} &\approx \left( x_i y_j - x_t y_j \right)
\end{align*}
\]
Frontal Cortex

GPe

STN

D2−D1+

excitatory

inhibitory

modulatory

SNc

striatum

SNr/GPi

* SNr activity (Jiang, Stein & McHaffie, 2003):

"Go" "NoGo"

Samejima et al., 2005, Science

Striatal coding for action-values: Go and NoGo cells?

Application to probabilistic reinforcement learning:

Weather Prediction Task

BG Model movies

Selection & Positive Feedback

Selection & Negative Feedback

After more extended training: Cortical response selection

Response Sequencing Trial

Go NoGo thalamus

excitatory

inhibitory

modulatory

See these movies and explanations at

http://ski.cog.brown.edu/BGmodel_movies.html

Application to probabilistic reinforcement learning:

Weather Prediction Task

"Go" "NoGo"

Weather Prediction Task

"Go" "NoGo"

Application to probabilistic reinforcement learning:

Weather Prediction Task

"Go" "NoGo"

Application to probabilistic reinforcement learning:
102
PD Simulation Results: Weather prediction

\[ 0 \quad 50 \quad 100 \quad 150 \quad 200 \]

Trial

\[ 10 \quad 15 \quad 20 \quad 25 \quad 30 \quad 35 \quad 40 \quad 45 \quad 50 \]

Percent Error

Intact

PD

No Indir

Global Nogo

Probabilistic Classification

- Intact nets extracted probabilistic structure by resolving differences in Go/NoGo representations.
- PD nets were impaired due to reduced dynamic range of DA.

Frank, 2005, J Cog Neurosci

103
What about Dopaminergic Medication Effects in PD?

- Medication actually causes deficits in some tasks: Probabilistic Reversal (Cools et al, 2002).
- Hypothesis: medication blocks the effects of DA dips needed to learn “NoGo”.

104
BG Model Go/NoGo associations, Predictions for Parkinson’s and Medication

Choose A Avoid B

Test Condition

-0.10
0.00
0.10
0.20
0.30
0.40
0.50

Go/NoGo Associations

Intact

Simulated PD

Simulated DA Meds

Probabilistic Selection

105
Parkinson’s Behavioral Results: Probabilistic Selection

Choose A Avoid B

Test Condition

Percent Accuracy

Seniors

PD OFF

PD ON

Probabilistic Selection

Test Performance

Frank, Seeberger & O’Reilly, 2004, Science
(See Cools et al, 2006 for similar findings)

106
Converging Evidence: PD & DA drugs in Healthy Participants

Choose A Avoid B

Test Condition

Percent Accuracy

Seniors

PD OFF

PD ON

Probabilistic Selection

Test Performance

Frank et al, 2004, Science
Frank & O’Reilly (2006), Behavioral Neuroscience

107
Testing the models:

Striatal Genetic Contributions

Striatal Genes: DARPP-32 and DRD2

- DARPP-32: protein concentrated in striatum, activated by D1 receptor stimulation, mediates DA effects on plasticity (Greengard et al, 1999).

Meyer-Lindenberg et al, 2007

Model Prediction:

DARPP-32 gene modulates Go learning.
DARPP-32 Modulates Relative Go learning

N=69

Choose A Avoid B

Test Condition

60 65 70 75 80 85

Accuracy

Go/NoGo Generalization

DARPP-32 gene

Frank, Moustafa, Haughey, Curran & Hutchison (2007), PNAS

Striatal Genes: DARPP-32 and DRD2

• DRD2 gene: D2 receptor in striatum (Hirvonen et al, 2004, 2005)
  – slow integration of BG NoGo signals
  – supports avoidance of most aversive stimulus “on average”

-> Model Prediction:
Avoid-B test phase performance

Sources of Individual Difference: DRD2 Gene

N=69

Choose A Avoid B

Test Condition

60 65 70 75 80 85

Accuracy

Go/NoGo Generalization

DRD2 gene

* Frank, Moustafa, Haughey, Curran & Hutchison (2007), PNAS
  (See Klein et al, 2007 for similar findings)

Want to learn more?
• Check out BasalGanglia.prt.pdf slides on cogsim website
• Check out http://ski.cog.brown.edu/BG_Projects/ for a few Emergent simulations replicating published effects. (papers avail on my website). One possibility to explore for final project..
• Later: BG interactions with PFC in higher level cognition

Extra

The following slides describe a recently developed alternative to TD, called PVLV, which we think is more biologically plausible and computationally powerful. This material is optional for the course.

Q: How do we learn to attach positive/negative valence to environmental stimuli?
A: The same way we learn lots of other stuff: the Delta Rule!

δ_{pv} = r - \hat{V}_{pv}

\hat{V}_{pv}: expected reward based on prior associations
r: reward
δ_{pv}: learning signal

The Problem

The Problem
Q: But what happens when environmental stimulus occurs before reward?

Basic Data: VTA DA Neural Firing in Conditioning

Rew     DA
CS
Dopamine spikes/dips are learning signals!!!

Standard Approach: TD

Error = Temporal Difference = TD:

Problems with TD

- Great algorithm, developed in computer science/machine learning, but is this actually what the brain does?
- Even if so, doesn’t specify how these signals are computed by systems upstream of DA...
- Current reward value is always relative to what happened just before, how much temporal dependency?
- Current reward value is always relative to what happened just before, how much temporal dependency?
- Chaining not seen in neural recordings.
- What determines “discount factor” $\gamma$, biologically?

Input

Output

Before

After

Reward

But what happens when environmental stimulus occurs before
Rat study: simultaneous CS and US DA spike
Pan et al., 2005, Journal of Neuroscience
Inconsistent with standard TD!

The PVLV Alternative
PVLV = Primary Value, Learned Value
• No reward predictions, just associations!
• No temporal dependencies: DA depends only on current state.
• Uses same basic delta-rule learning as TD (Rescorla-Wagner).

PVLV: Two Separate Mechanisms (PV, LV)

PV: Primary Value
• Trained at each point in time on actual reward value:
  \[ \delta_t = r_t - \hat{V}_t \]
• This uses immediate prediction (\(\hat{V}_t\) of current rew value (\(r_t\))
• Accounts for canceling of DA spike @ rew, and DA dips when no rew received.
• But this doesn't account for predictive DA spikes... (actually results in predictive DA dips!)

LV: Learned Value
• Represents perceived values of stimuli even when there is no current rew expectation.
• Only gets training signal @ rew, or when PV expects some rew.
• (ie learning is filtered by primary PV system.)
• Learns at time of rew, but not at CS onset.
• Generalizes rew values to CS...
• Accounts for DA spikes for stimuli that have previously been associated with reward.

PVLV: Computationally Powerful

Comparison with TD on Random Delays (breaks TD chaining):

\[
\begin{array}{cccccc}
\text{Delay} & 3 & 6 & 12 & \text{Avg DA Value} \\
\text{Rnd Delay, p=.2, TD Disc .95 Lrate .1} & \text{vs} & \text{vs} & \text{vs} & \text{vs} & \text{vs} \\
\text{Rnd Delay, p=.2, PVLV Lrate .005} & \text{vs} & \text{vs} & \text{vs} & \text{vs} & \text{vs} \\
\end{array}
\]

Enables working memory model to learn complex WM tasks.
Diffs: Anatomical (CNA vs. VS; Dorsal vs. Ventral Patch)

Functional (intrinsic timing? LV system cannot train itself).

PVLV accounts for timing data better than TD!

• Data: during transient learning period, both rews and CS elicit activation.
  This accounted for by PV, LV systems operating in parallel.

• TD: predicts chaining back in time from rew to CS.

• Data: delayed rewards cause dips @ usual time, then spikes.
  This accounted for by both TD and PV.

• Data: early rewards cause spikes, then dips @ usual time.
  This accounted for by PV (spike), PV (dip), but TD only accounts for spike.

More Key Predictions from PVLV

Excitatory

Inhibitory

CNA

VS

patch

VS

patch

Stimuli (CS)

DA

VTA/SNc

Cereb.

(CNA) Pavlovian conditioning (e.g., Killcross et al. ’97).

NAc (patch/shell) = Extinction (Ferry et al. ’00; Annett et al., 89), Blocking (data?).

NAc (matrix/core) = Basic actions (OR’s, approach, avoid).

CNA can’t train itself: No 2nd order conditioning.

PVLV provides computationally motivated architecture.

Conclusions

PVLV provides computationally motivated architecture that seems to fit with biology & behavioral data.

These learning mechanisms enable arbitrary stimuli/goals to be plugged into our fixed set of built-in motivational drives.

Something motivates every generated mental-state, always!

PVLV, WM, and DA
Two Separate Mechanisms (PV, LV)

**PV learning:**
\[ \delta_{pv} = r - \hat{V}_{pv} \]

or
\[ \delta_{pv} = PV_{e} - PV_{i} \]

\[ \Delta w_i = \epsilon x_i \delta_{pv} \]

**LV learning (filtered by PV):**
\[ \Delta w_i = \begin{cases} \epsilon (r - \hat{V}_{lv}) x_i & \text{if } \hat{V}_{pv} > \theta_{pv} \text{ or } r_t > 0 \\ 0 & \text{otherwise} \end{cases} \]

**Global DA (PV dominates):**
\[ \delta_t = \begin{cases} \delta_{pv} & \text{if } \hat{V}_{pv} > \theta_{pv} \text{ or } r_t > 0 \\ \delta_{lv} & \text{otherwise} \end{cases} \]

\[ \Delta v = \begin{cases} \delta_{pv} & \text{if } \hat{V}_{pv} > \theta_{pv} \text{ or } r_t > 0 \\ \delta_{lv} & \text{otherwise} \end{cases} \]

**LV Extras**

- DA spikes only observed @ CS onset, don’t continue throughout delay until reward. Problem for PV?
- Solution: PV system has synaptic depression, accommodates to constant sensory inputs; only perceives values of stim that were not present in last time step.
- This is also important for PFC learning. (stay tuned)