Sharif University Of Technology

Advanced Topics in Neuroscience

Simulation 05 (Motivation and Classical Conditioning)

Instructor: Dr. Ali Ghazizadeh

Zahra Kavian - 98102121

Contents

1 RW rule **2**
- 1.1 Conditioning Paradigms . . . 2
  - 1.1.1 Pavlovian . . . 2
  - 1.1.2 Extinction . . . 2
  - 1.1.3 Partial . . . 2
  - 1.1.4 Blocking . . . 3
  - 1.1.5 Inhibitory . . . 3
  - 1.1.6 Overshadow . . . 4
- 1.2 Part 2 . . . 4

2 Kalman Filter **5**
- 2.1 Dayan and Yu Paper Result . . . 5
  - 2.1.1 Blocking Conditioning . . . 5
  - 2.1.2 Unblocking Conditioning . . . 5
  - 2.1.3 Backward Blocking Conditioning . . . 5
- 2.2 Process Noise vs Measurement Noise effect . . . 6
- 2.3 Kalman Gain at Steady state . . . 8
- 2.4 Error Effect on Uncertainty . . . 8
- 2.5 Reward and Punishment Conditioning . . . 8

3 Unknown Uncertainty (Adaptive Factor) **9**

Section 1: RW rule

1.1 Conditioning Paradigms

1.1.1 Pavlovian

Paradigm	Pre-Train	Train	Result
Pavlovian		s $\rightarrow$r	s $\rightarrow' r'$

We use $RW(RescolaWagner)$ rule and simulate this paradigm. During training, stimulus lead to reward. The state value gradually reach the expected reward and λ be zero. After that the value doesn't change.

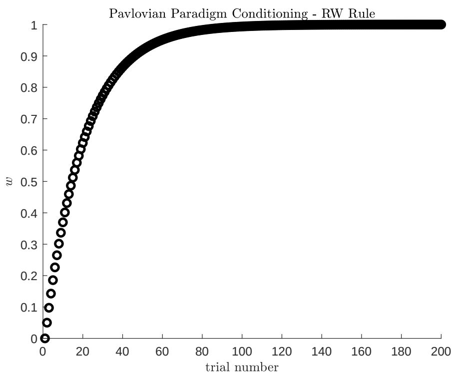


Figure 1: Pavlovian paradigm

1.1.2 Extinction

Paradigm	Pre-Train	Train	Result
Extinction	s $\rightarrow$r	s $\rightarrow$0	s $\rightarrow' 0'$

As you see in figure 2, in pre-train the, state value increase to each the expected reward, but during the training, the stimulus doesn't lead to reward and state value decrease gradually and be zero.

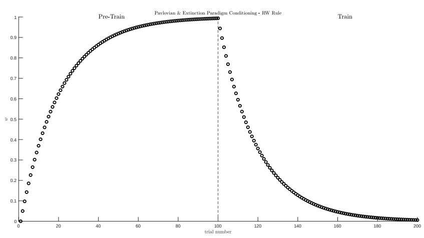


Figure 2: Extinction paradigm

1.1.3 Partial

Paradigm	Pre-Train	Train	Result
Partial		s $\rightarrow$r s $\rightarrow$0	s $\rightarrow \alpha' r'$

During training, some stimulus lead to reward and some of them not. At the end, value reach the percentage of the expected reward as is shown in figure 3.

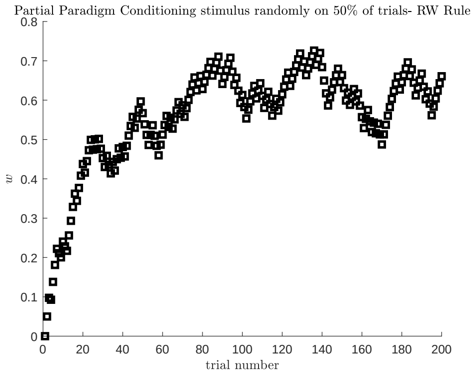


Figure 3: Extinction paradigm

1.1.4 Blocking

Paradigm	Pre-Train	Train	Result
Blocking	$s_1 \rightarrow$r	$s_1 + s_2 \rightarrow$r	$s_1 \rightarrow$'r' $\quad s_2 \rightarrow$'0'

In pre train, the first stimulus lead to reward. In training, two stimulus simultaneous lead to reward. At the end, the first stimulus value reach the expected reward and the second reach the zero value, because the the state value reach the reward in pre-training and λ is zero during training.

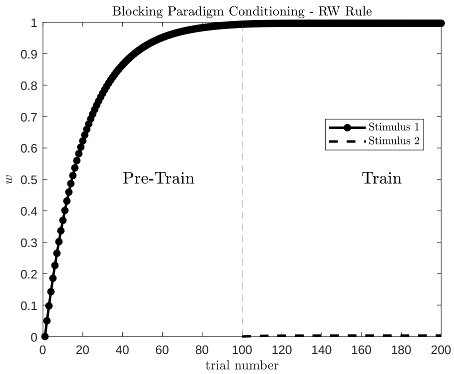


Figure 4: Extinction paradigm

1.1.5 Inhibitory

Paradigm	Pre-Train	Train	Result
Blocking		$s_1 + s_2 \rightarrow$0 $\quad s_1 \rightarrow$'r'	$s_1 \rightarrow$'r' $\quad s_2 \rightarrow$'-r'

In training, one stimulus lead to reward, but the reward is not given when use two stimulus. So one stimulus get the reward value and other consider as a inhibitory stimulus and get the negative reward value.

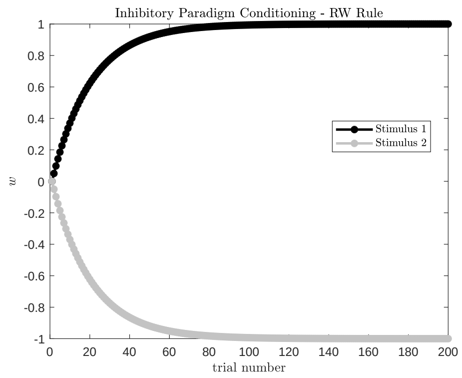


Figure 5: Extinction paradigm

1.1.6 Overshadow

Paradigm	Pre-Train	Train	Result
Blocking		$s_1 + s_2 \rightarrow$r	$s_1 \rightarrow \alpha_1' r' \quad s_2 \rightarrow \alpha_2' r'$

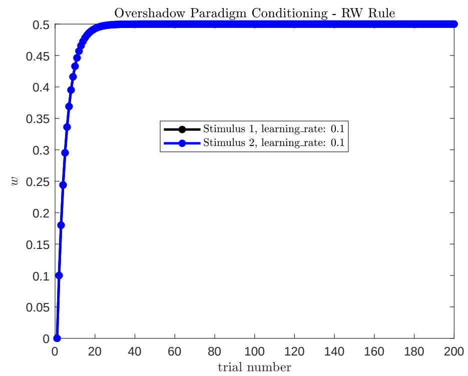


Figure 6: Extinction paradigm

1.2 Part 2

In overshadow paradigm, if two stimulus have a different learning rate, their final value will be different. The stimulus with higher learning rate (faster) reach the more state value.

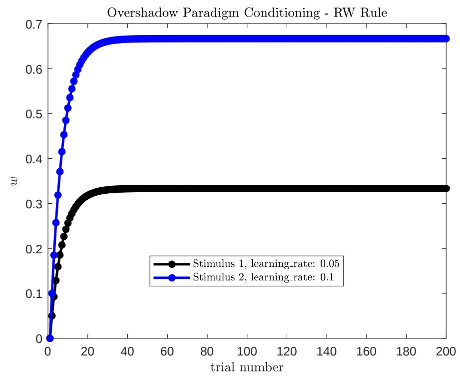


Figure 7: Extinction paradigm

Section 2: Kalman Filter

RW rule couldn't explain some paradigm such as the *Secondary* or *Backward Blocking*. So we implement Kalman filter

2.1 Dayan and Yu Paper Result

2.1.1 Blocking Conditioning

As you see in figure 8, the learning second stimulus is blocked by the first stimulus, so its value reach zero as time pass.

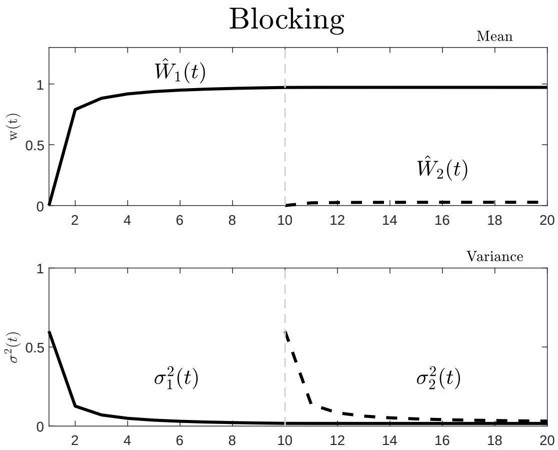


Figure 8: Blocking paradigm. Upper: the average estimated weight, lower: the weight variance.

2.1.2 Unblocking Conditioning

Paradigm	Pre-Train	Train	Result
Blocking	$s_1 \rightarrow$'r'	$s_1 + s_2 \rightarrow$2r	$s_1 \rightarrow r + (1-\alpha)r \quad s_2 \rightarrow \alpha r$

In this condition, th reward is doubled as the second stimulus is presented. As you see in figure 9, at the start time of the second stimulus, its uncertainty is high and its value gradually increase to reach the expected vale (r). At the end, the total average weight is "$2r$".

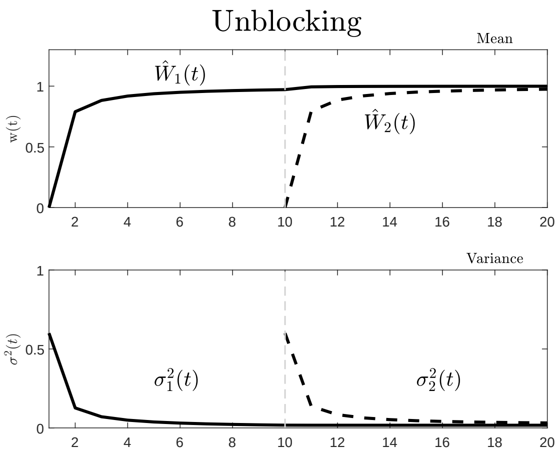


Figure 9: Unblocking paradigm. Upper: the average estimated weight, lower: the weight variance.

2.1.3 Backward Blocking Conditioning

Paradigm	Pre-Train	Train	Result
Blocking	$s_1 + s_2 \rightarrow$'r'	$s_1 \rightarrow$r	$s_1 \rightarrow$'r' $\quad s_2 \rightarrow 0$

One of stimulus lead to reward by itself, so other stimulus doesn't have effect to get the reward and its value decrease to zero. In figure 10, the joint distribution of two stimulus weights are shown. At the first time, two distribution are un-correlated and joint distribution looks a circle. In time equal 9, their weights are anti-correlated and in time equal 19, the first stimulus weight mean is nearly one and the second stimulus weight mean is almost zero.

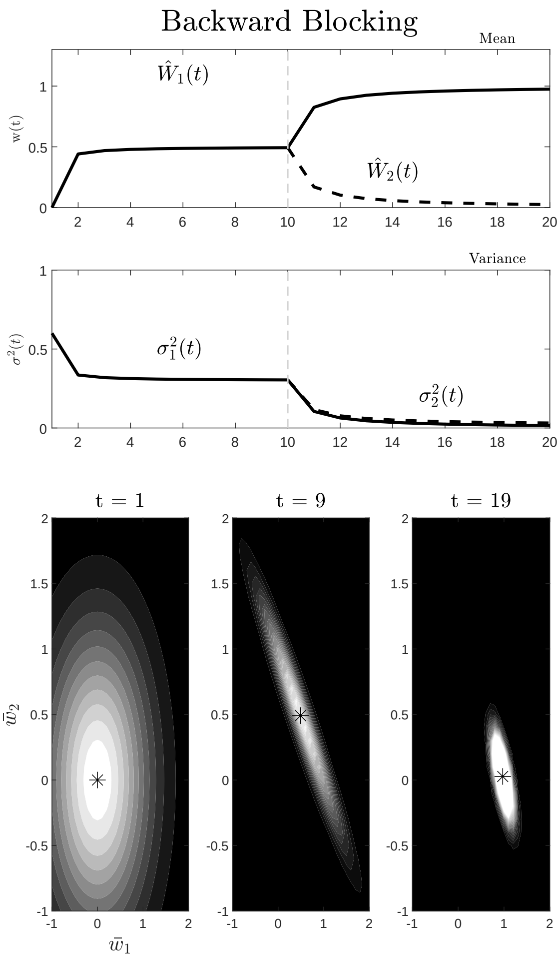
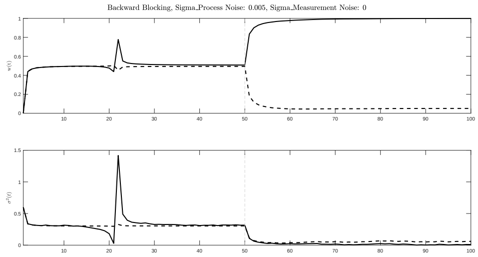


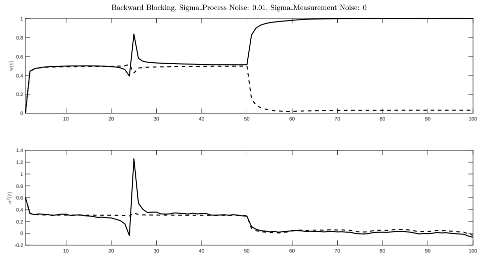
Figure 10: First:Backward Blocking paradigm. Upper: the average estimated weight, lower: the weight variance. Second:Joint Distribution of weights over time.

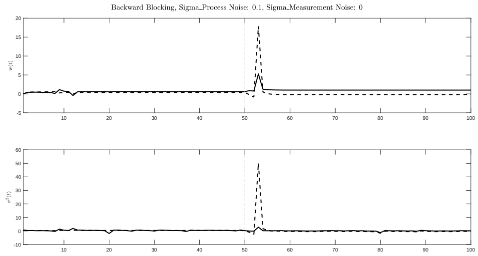
2.2 Process Noise vs Measurement Noise effect

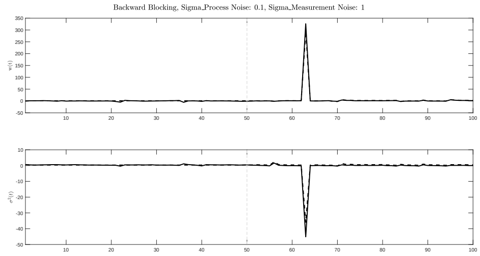
For different process and measurement noise repeat the previous part again.

I use Gaussian process and measurement noise once:

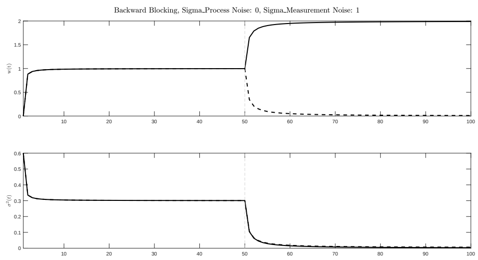


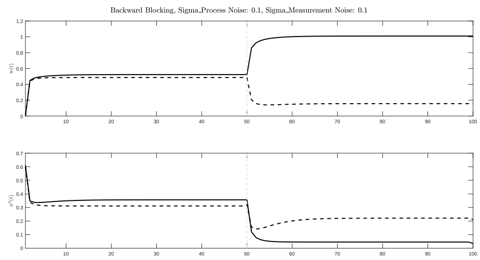


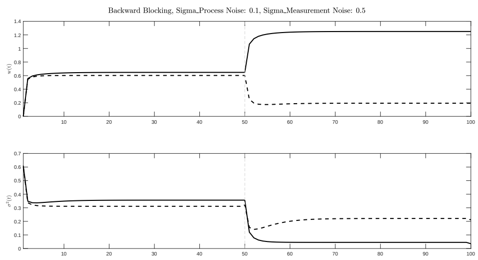


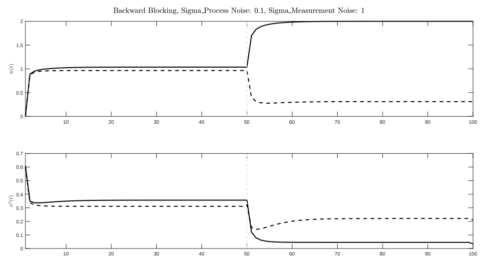


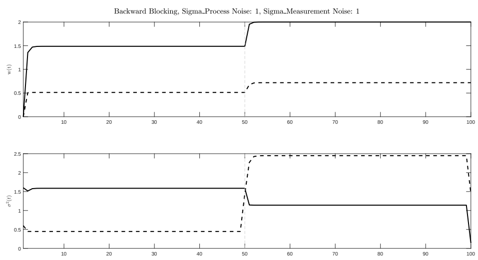
And again consider constant noise:

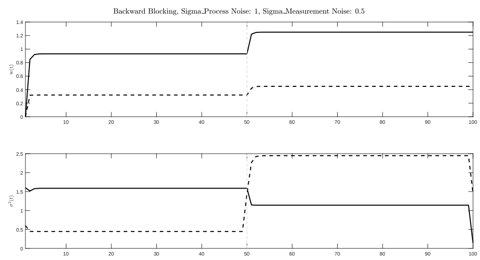


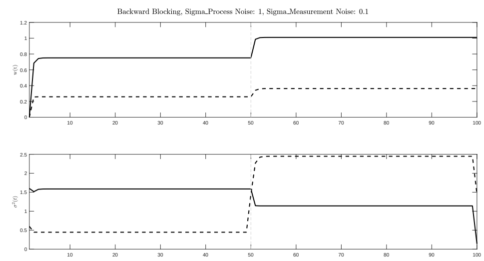












2.3 Kalman Gain at Steady state

$$G = \sum_{\infty} C^T \left(C \sum_{\infty} C^T + V \right)^{-1} \tag{1}$$

The filter gain is only depend on the uncertainty matrix and measurement noise.

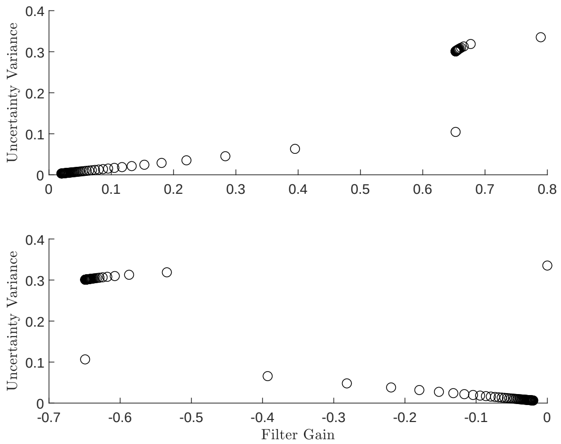


Figure 11: The factor effected on Kalman filter gain.

2.4 Error Effect on Uncertainty

The uncertainty is changed as follows:

$$\sum_{t|t} = \sum_{t|t-1} C^T \left(C \sum_{t|t-1} C^T + V \right)^{-1} C \sum_{t|t-1} \tag{2}$$

The error of each trial has no effect on uncertainty.

2.5 Reward and Punishment Conditioning

Paradigm	Pre-Train	Train	Result
Blocking	s_1 →'r'	s_1 →-r	–

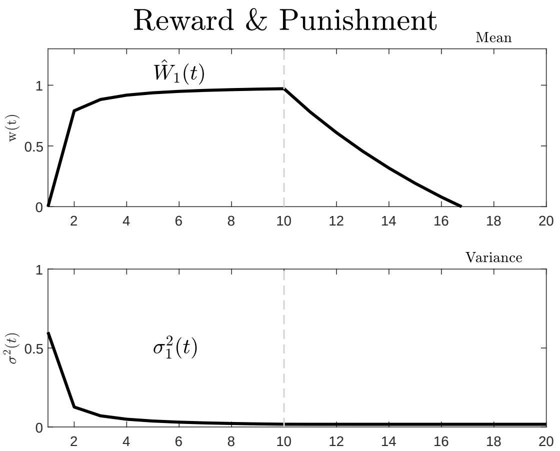


Figure 12: Reward and Punishment Conditioning.

At first, weight increase and reach the reward value rapidly (high uncertainty) but when punishment start, the weight decrease slowly, because at the punishment time, the error value

$(r(t) - x(t).w(t))$ is -2 and the uncertainty still has a small value. So the state weight is high but the learning rate is small. The weight will decrease gradually in time. The main reason is the uncertainty would not change as our environment had changed. So Kalman Filter can not follow changing environments.

Section 3: Unknown Uncertainty (Adaptive Factor)

The defined β parameters change as the environment changes. If it reach an specific threshold, we update an uncertainty. So in this model, we could tract learning new things.

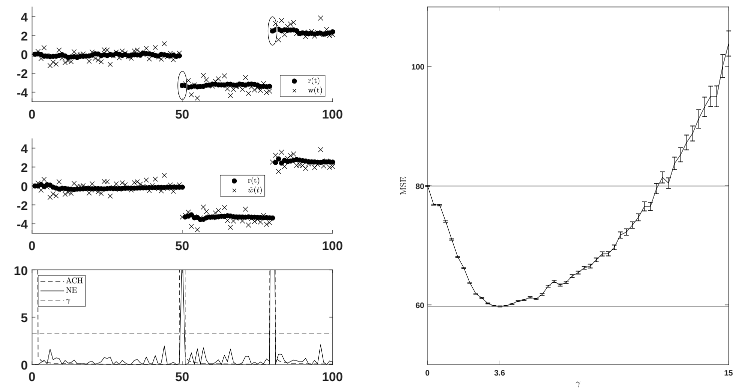


Figure 13: Right top: the simulated weight and noisy related response. Right middle: "the mean estimate $\hat{w}$ from the $ACh - NE$ approximate learning scheme closely tracks the actual w". Right bottom: "Corresponding levels of ACh (dashed) and NE (solid) for the same sequence". Left: "MSE of the approximate scheme as a function of the threshold γ. Optimal γ for this particular setting of parameters is 3.6, for which performance approaches that of the exact algorithm (lower line). When γ is too large or too small, performance is as bad as, or worse than, if $w(t)$ is directly estimated from $r(t)$ without taking previous observation into account (top line)".