A Trace-conditioning Testbed

Banafsheh Rafiee

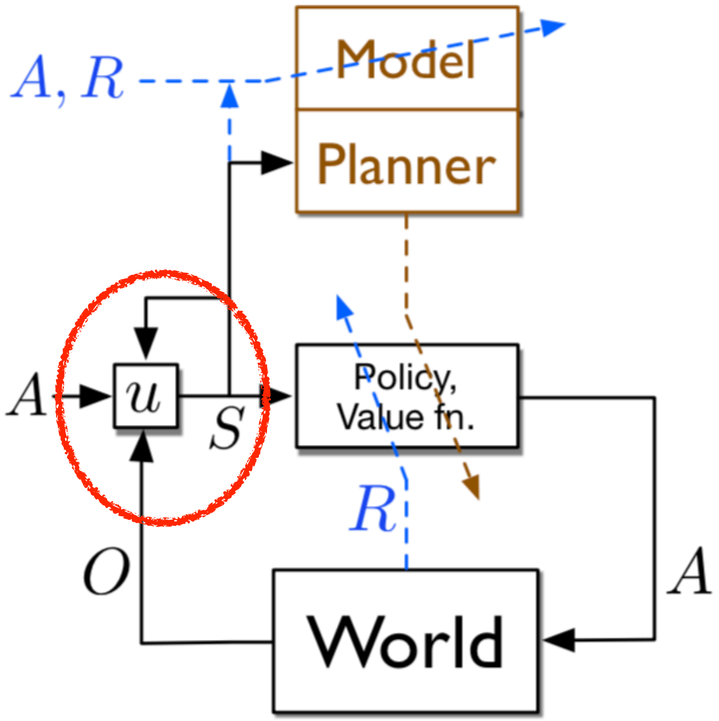
August, 2019

I see three challenges in designing a good way to construct states.

- Representation learning: how to build a representation of the world suitable for learning.
- Partial observability: how to learn if our observations do not carry enough information.
- Learning complex functions: how to handle non-linear functions in linear representation systems.

The part of the agent that construct the state is called the state update.

$$S_{t+1} \doteq u(S_t, A_t, O_{t+1}), \text{ for all } t \geq 0$$



I explore the problem of constructing the state by investigating a simple problem.

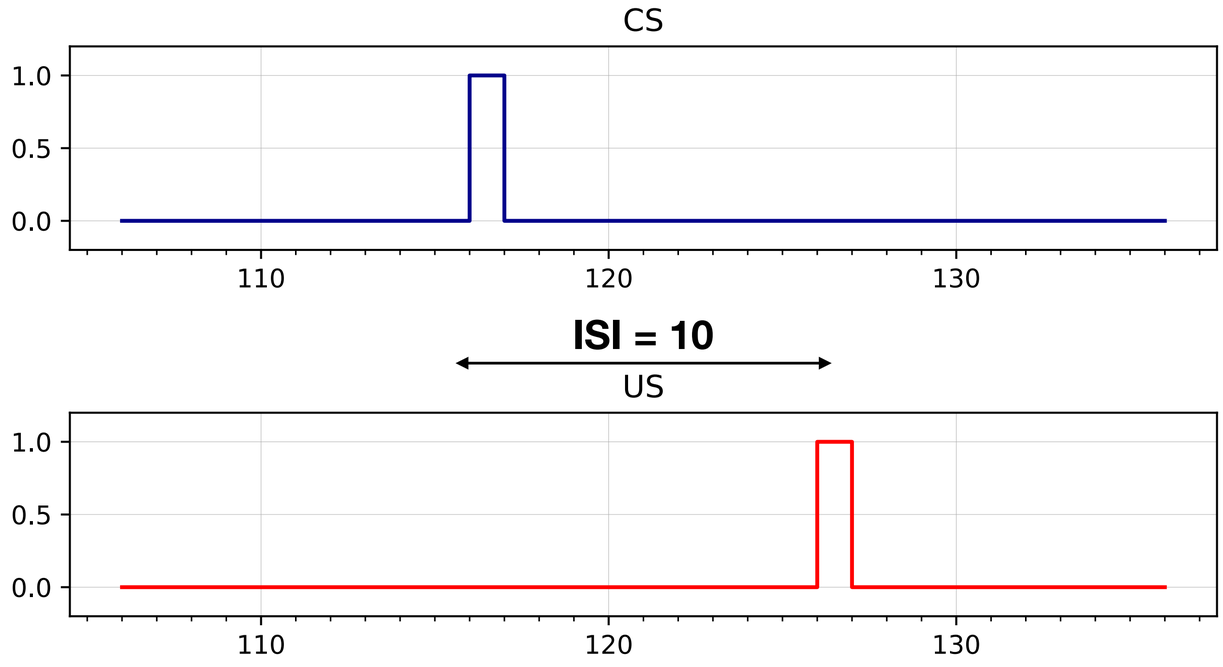
A Trace-conditioning Testbed

The testbed consists of a series of trials.

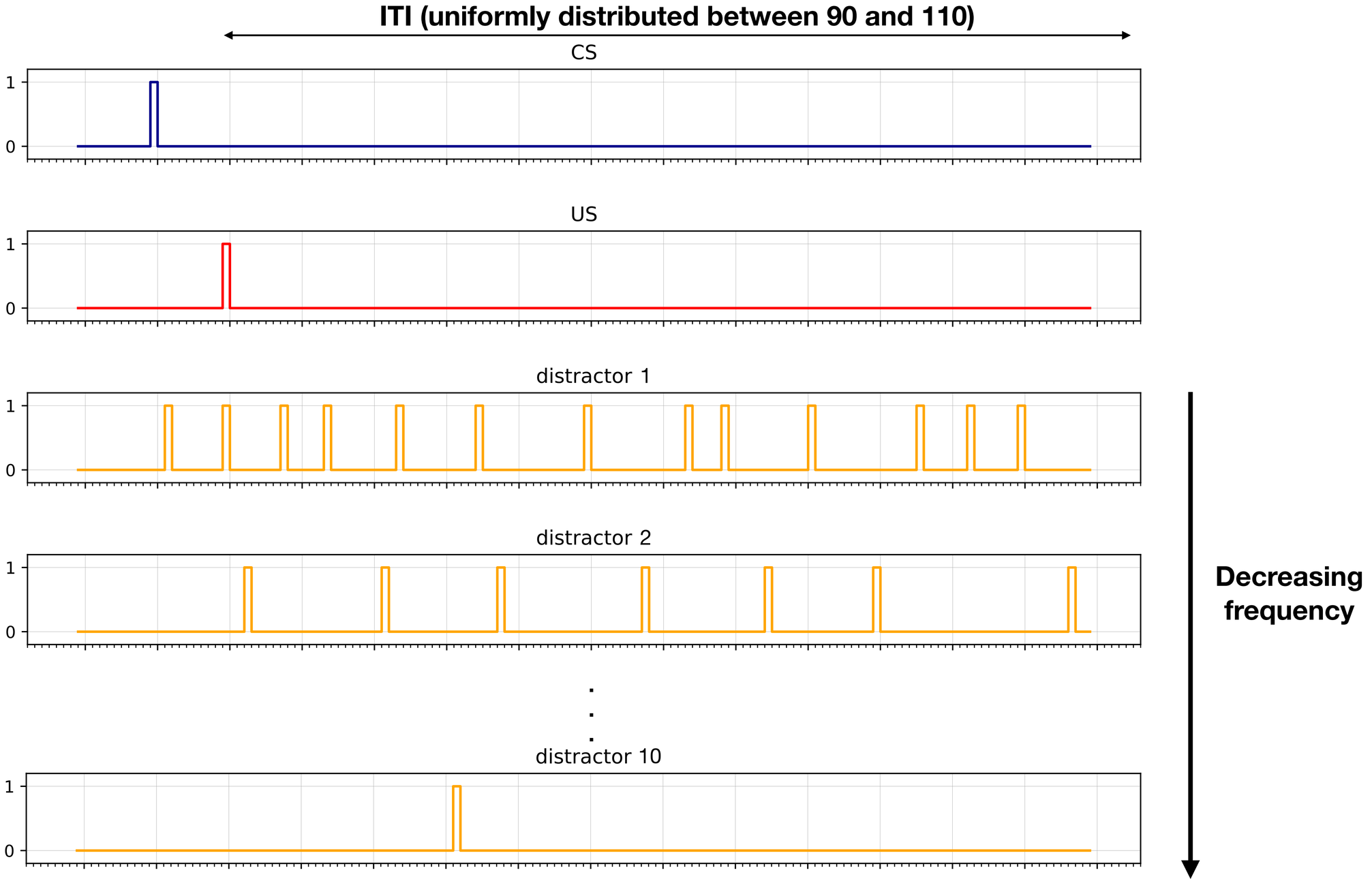
At each trial a sequence of stimuli is presented:

- a US (unconditioned stimulus): is to be predicted **Food**

- a CS (conditioned stimulus): is predictive of the US **Bell ringing**

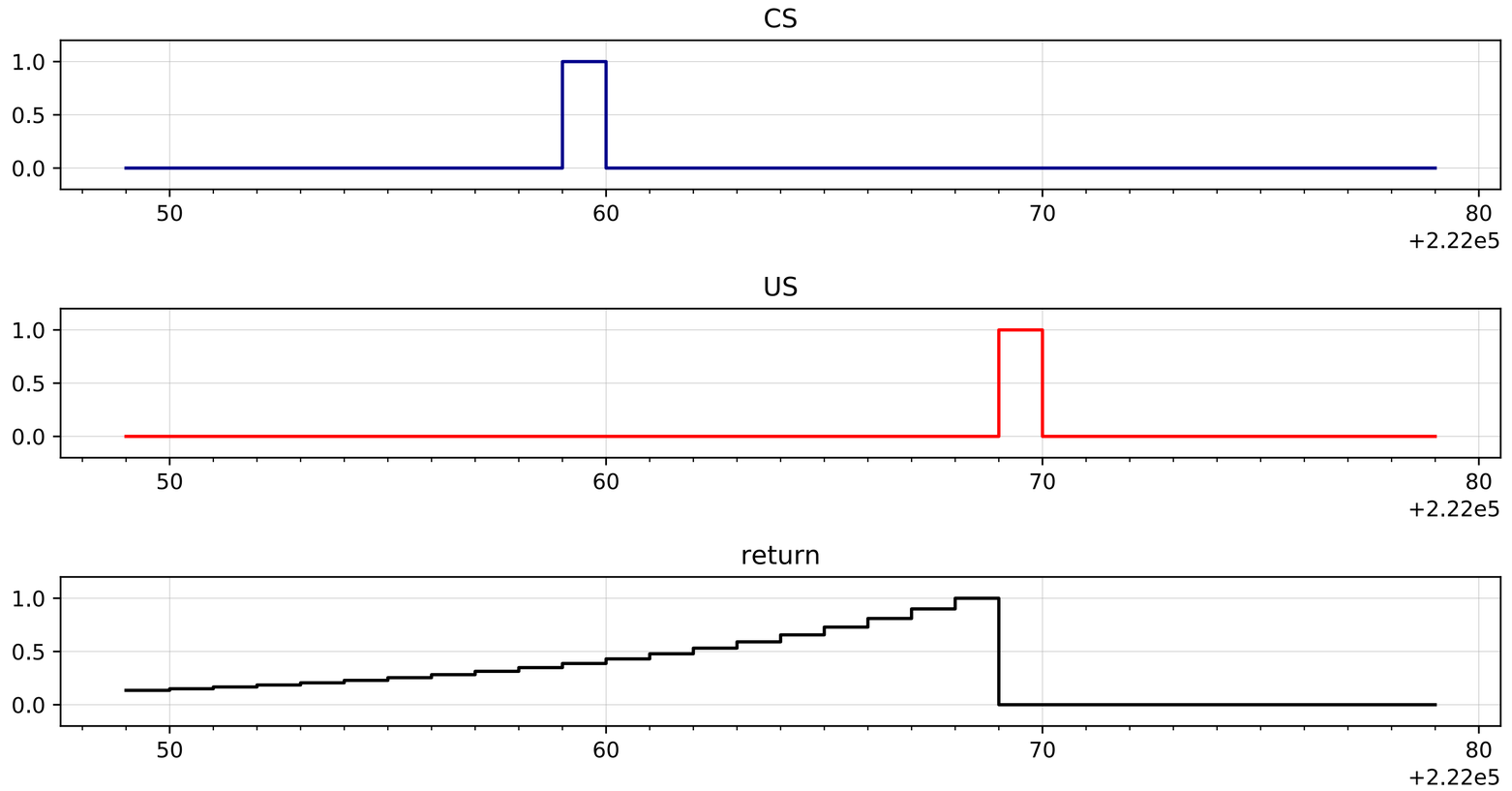


The testbed also includes 10 distractor stimuli happening in a Poisson fashion.



We want to predict the expected discounted sum of the US.

$\gamma = 0.9$



Predicting the US is challenging because of the trace interval.

Trace interval is the empty interval from the CS to the US.

The agent needs to keep some kind of a trace of the CS to be able to predict the US.

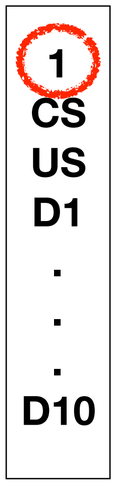
Note that the trace of the CS is different from the eligibility trace.

A simple solution method

Algorithm: TD(λ) with $\lambda = 0.9$

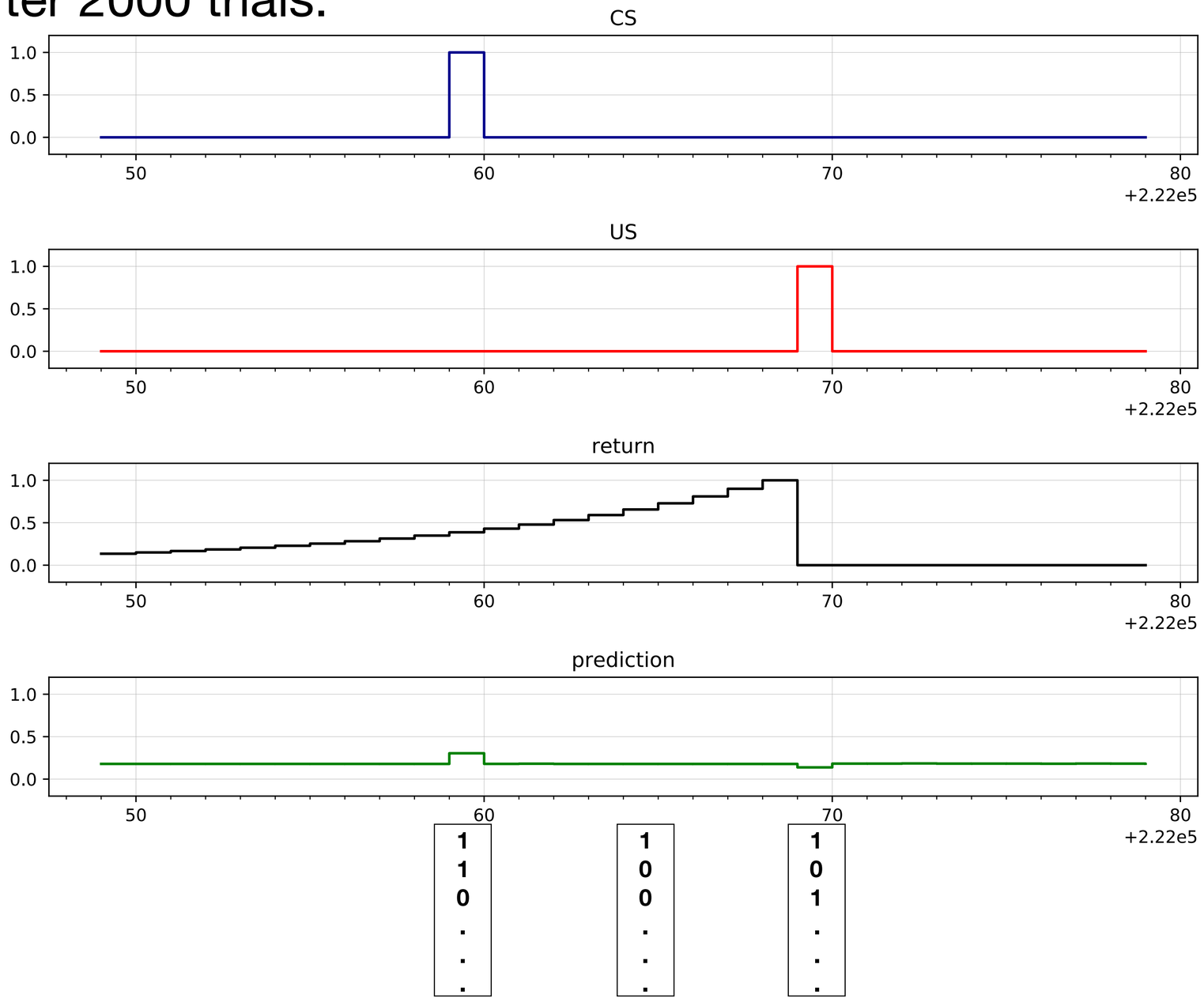
Feature representation: The presence representation

- has one feature for each stimuli.



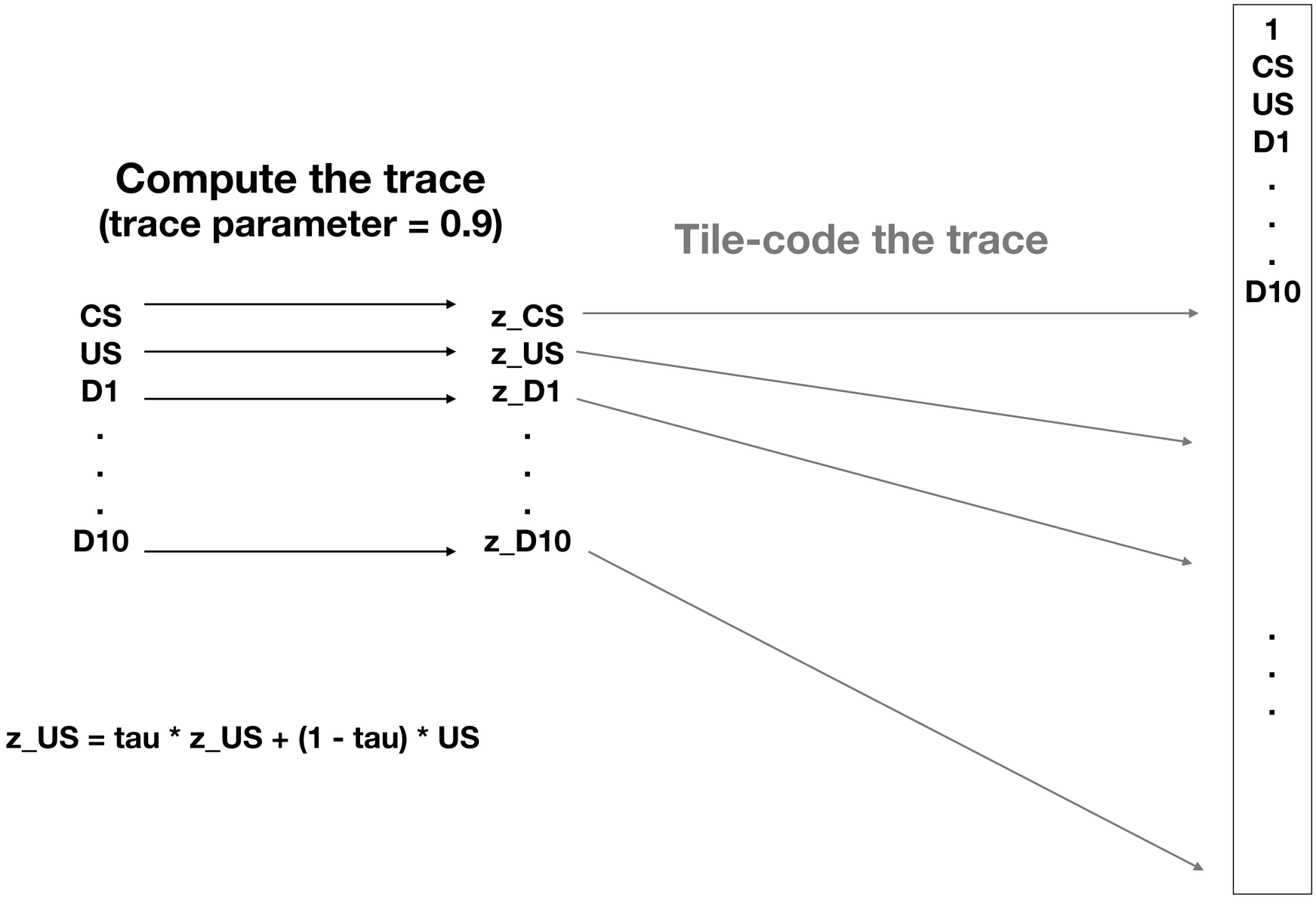
Presence representation is not sufficient.

After 2000 trials:



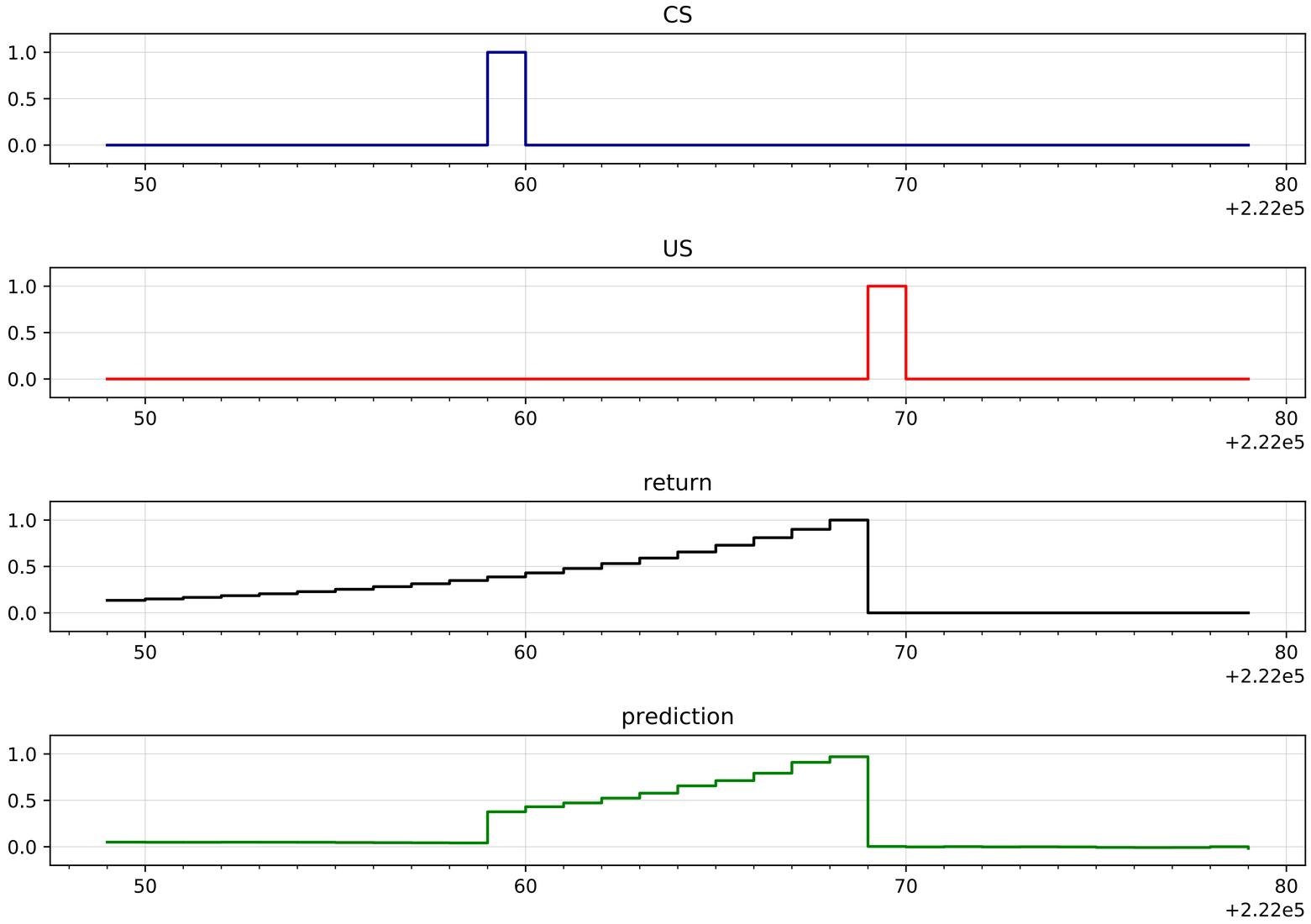
Note that we used eligibility traces.

A possible solution is to use traces of stimuli in representation and tile code them



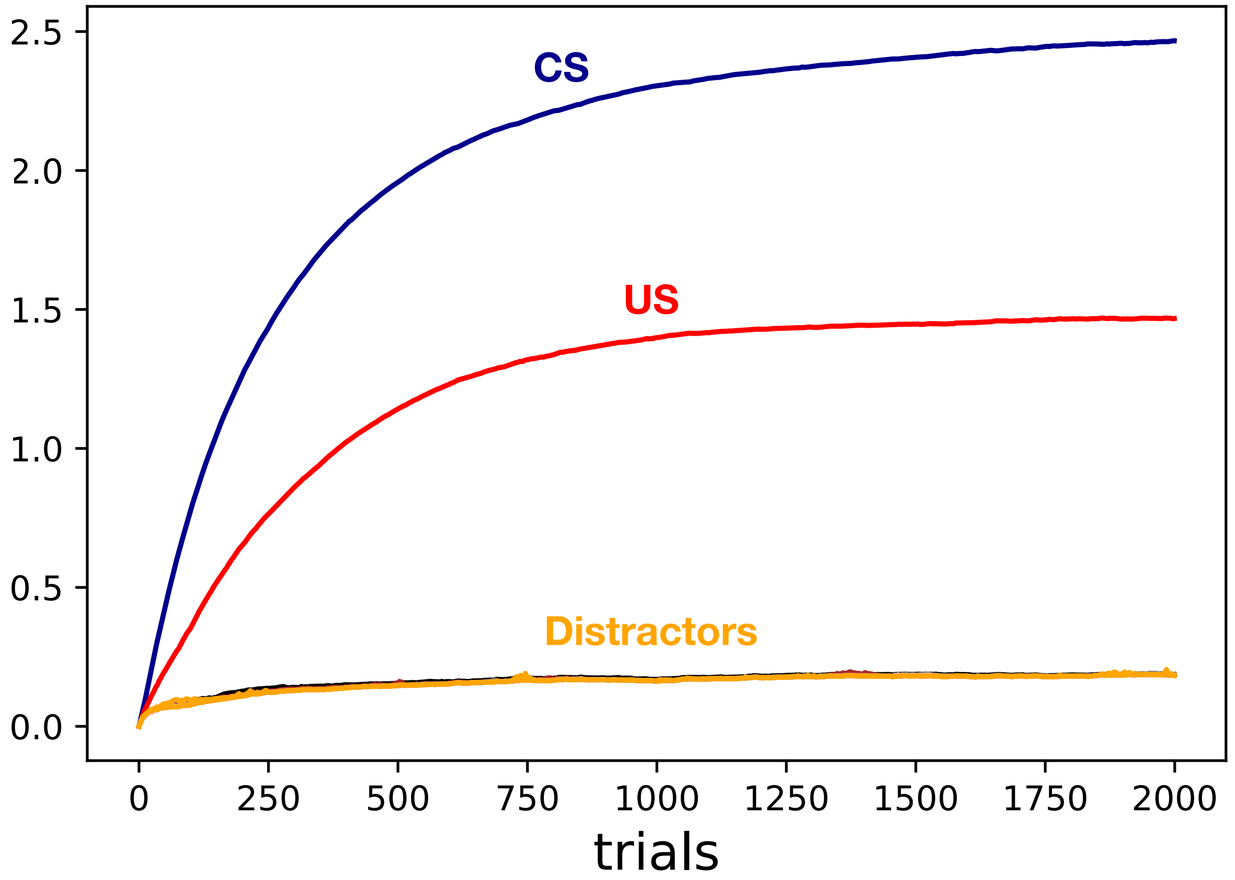
The new representation is sufficient to find a good approximation.

After 2000 trials:

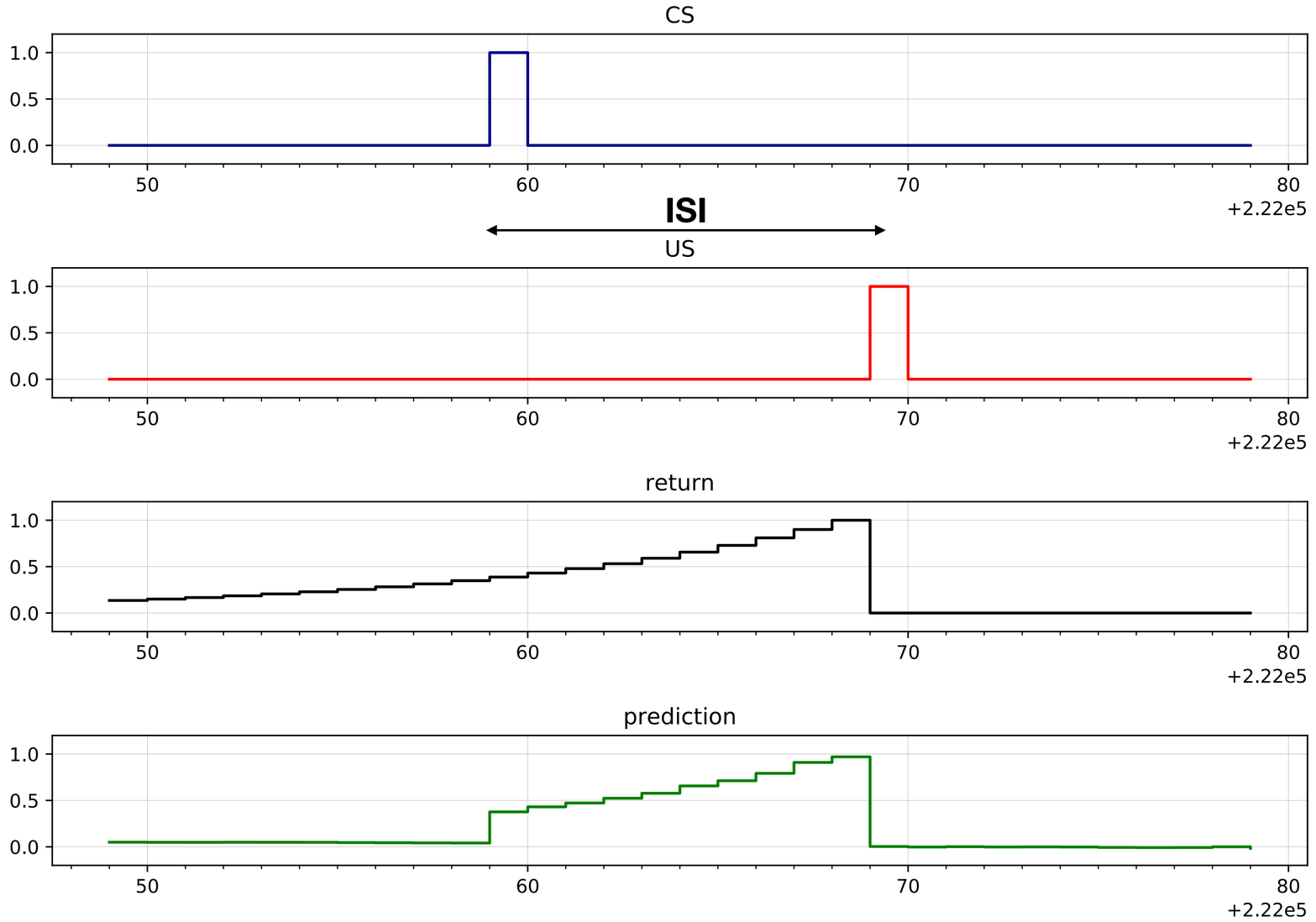


With the new representation, the learning algorithm associates high weights with the features corresponding to CS and US and low weights with the features corresponding to the distractors.

Sum of absolute weights of features associated with traces of stimuli

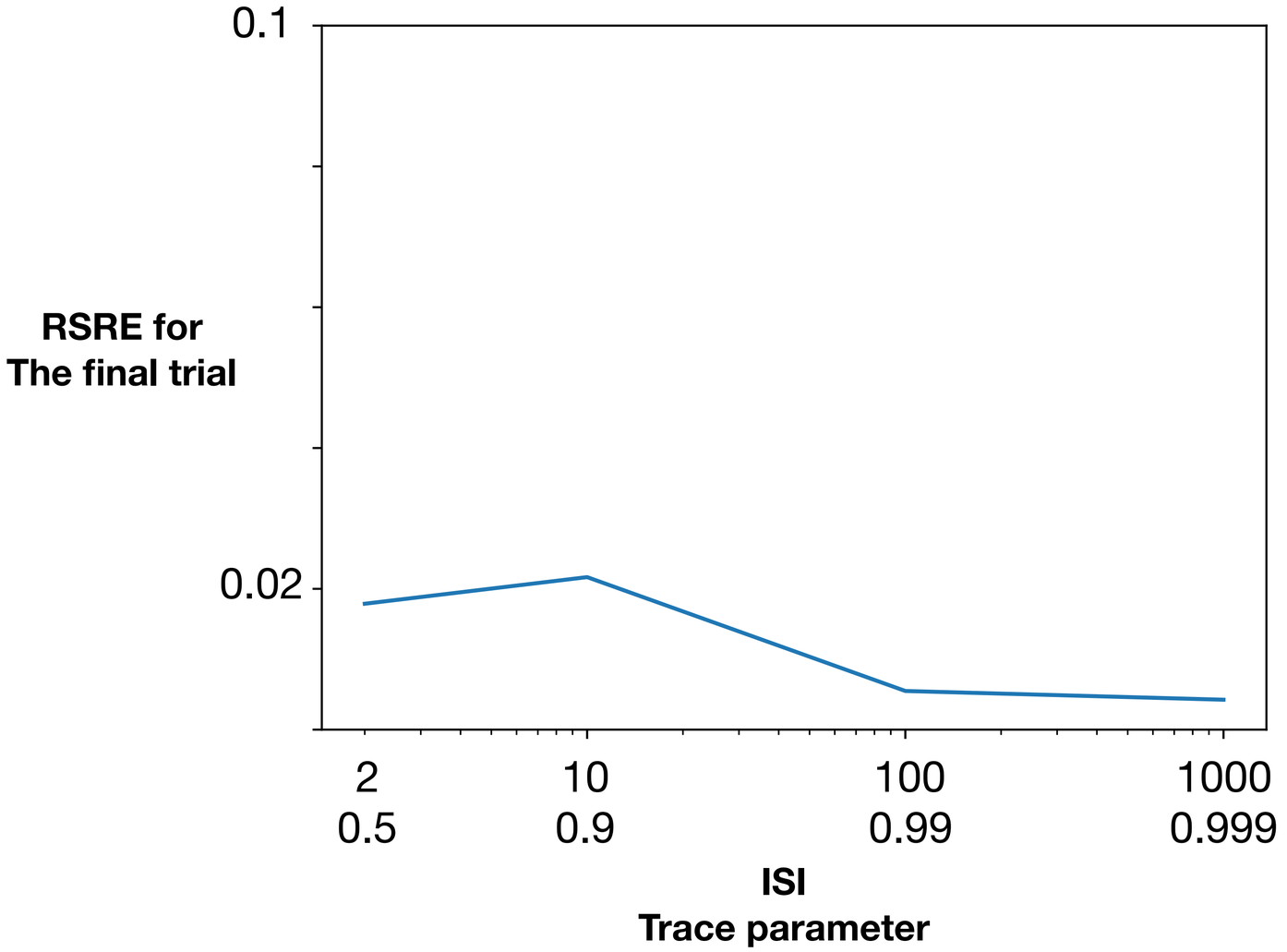


What happens for higher values of ISI?



Performance measure = root squared return error (RSRE) at CS onset

For larger values of ISI we need longer traces.



Adjusting the trace parameter with ISI

How successful was tile-coding the traces?

There are three levels of success to this problem:

- We are able to represent the answer 
- We can find the answer efficiently 

We should compare to natural competitors like LSTMs

- We can discover the useful features 

Eventually we want to do discovery

Take home messages

The problems of state representation, partial observability, and learning nonlinear functions all involve the state-update function.

The trace-conditioning testbed is useful for investigating the state-update function.

Tile coding the traces is natural way for enhancing the state representation.

Future work should investigate the efficiency of tile coding the traces and address the discovery problem.

Questions?