

---

# CCNLab: A Benchmarking Framework for Computational Cognitive Neuroscience

---

**Nikhil X. Bhattasali**  
Stanford University  
nikhilxb@cs.stanford.edu

**Momchil S. Tomov**  
Harvard University  
mtomov@g.harvard.edu

**Samuel J. Gershman**  
Harvard University  
gershman@fas.harvard.edu

## Abstract

CCNLab is a benchmark for evaluating computational cognitive neuroscience models on empirical data. As a starting point, its focus is classical conditioning, which studies how animals predict reward and punishment in the environment. CCNLab includes a collection of simulations of seminal experiments expressed under a common API, as well as tools for visualizing and comparing simulated data with empirical data. CCNLab is broad, incorporating representative experiments from different categories of phenomena; flexible, allowing the straightforward addition of new experiments; and easy-to-use, so researchers can focus on developing better models. We envision CCNLab as a testbed for unifying computational theories of learning in the brain. We also hope that it can broadly accelerate neuroscience research and facilitate interaction between the fields of neuroscience, psychology, and artificial intelligence.

## 1 Introduction

Brains are the *de facto* standard for general intelligence [Lake et al., 2017], and many researchers believe that progress in artificial intelligence is intimately intertwined with understanding natural intelligence [Hassabis et al., 2017]. Modern research in neuroscience and psychology increasingly relies on computational models to express theories about how the brain works [Durstewitz et al., 2016, Jonas and Kording, 2017, Linderman and Gershman, 2017, Kriegeskorte and Douglas, 2018, Levenstein et al., 2020, Gershman, 2021]. This has brought the fields studying artificial and natural intelligence even closer, with computational neuroscientists directly borrowing ideas from machine learning [Montague et al., 1996, Yamins et al., 2014, Stachenfeld et al., 2017, Ma and Peters, 2020, Saxe et al., 2020] and vice versa [LeCun et al., 1995, He et al., 2016, Sutton and Barto, 2018]. This perpetuates a “virtuous cycle” in which the science and engineering of intelligence may progress together [Hassabis et al., 2017].

Despite these promising developments, there is a significant rift between machine learning and computational neuroscience in how research is evaluated and compared. Machine learning has benefited from the widespread adoption of publicly available datasets and benchmarks, like ImageNet [Deng et al., 2009] and OpenAI Gym [Brockman et al., 2016]. This has been instrumental for algorithm design, as the strengths and weaknesses of different work can be directly compared, and research is incentivized to perform beyond narrow, domain-specific tasks. In contrast, models in computational neuroscience have often been confined to a small set of phenomena in a specific domain, and due the lack of standards for evaluation, even those models can be difficult to compare.

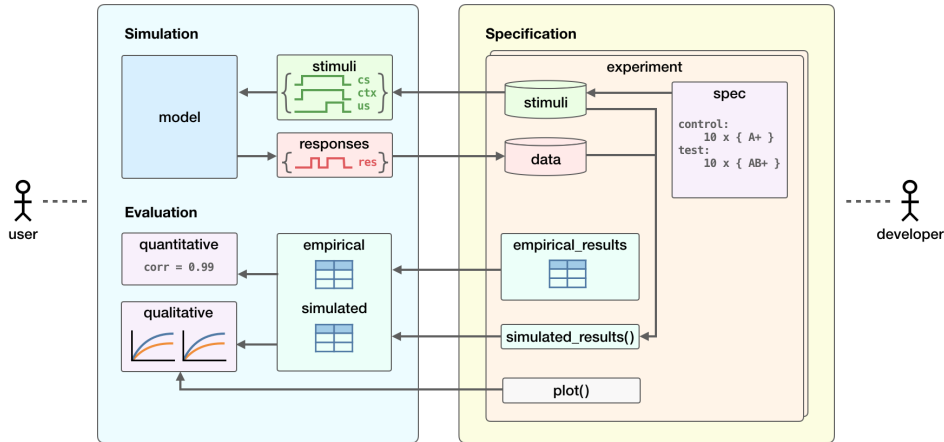


Figure 1: **Overview of CCNLab architecture.** Users can simulate their models on a wide variety of classical conditioning experiments, and the simulated results are evaluated against empirical results from published work. Each experiment is an environment that simulates a schedule of stimuli mirroring a real-world study.

We present **CCNLab** (short for Cognitive Computational Neuroscience Lab), a benchmark for computational models of classical conditioning. Using our Python framework inspired by OpenAI Gym [Brockman et al., 2016], users can simulate their models on a wide variety of classical conditioning experiments, and the simulated results are evaluated against empirical results from published work. Currently, the CCNLab registry includes 30 experiments to simulate the most established classical conditioning phenomena selected from the list provided in [Alonso and Schmajuk, 2012]. Our framework is modular and extensible, allowing developers to easily extend the benchmark with additional experiments from the literature or of their own design.

Classical conditioning has a rich history dating back to [Pavlov, 1927] and exhibits a diverse set of phenomena that have been extensively studied [Alonso and Schmajuk, 2012]. It has attracted much interest for computational modeling [Ertugrul and Tagluk, 2015, Gershman, 2015, Kutlu and Schmajuk, 2012b], and is closely related to reinforcement learning and optimal sequential decision making [Niv, 2009]. These factors make it an excellent starting point for introducing benchmarks into computational neuroscience and encouraging reproducible, rigorous evaluation of models. To that end, we perform a series of baseline experiments to evaluate existing methods on this benchmark. Our analysis reveals that, while our baselines are able to reproduce certain effects well, they lack the ability to generalize to a wider range of phenomena. We conclude by suggesting future extensions and research directions for computational modeling guided by publicly available benchmarks.

Our code for the benchmarks and baseline experiments has been open-sourced under the MIT License and is available at: <https://github.com/nikhilxb/ccnlab>.

## 2 Background

### 2.1 Classical Conditioning

A critical feature of intelligence is the ability to adapt in response to positive and negative feedback. The branch of machine learning that studies how agents can take actions that maximize reward is known as reinforcement learning [Sutton and Barto, 2018]. In neuroscience and psychology, learning based on rewards and punishments is studied through the paradigms of classical (Pavlovian) conditioning and instrumental (operant) conditioning. Classical conditioning studies how animals learn to predict outcomes in their environment, while instrumental conditioning studies how animals learn to select actions that lead to better outcomes.

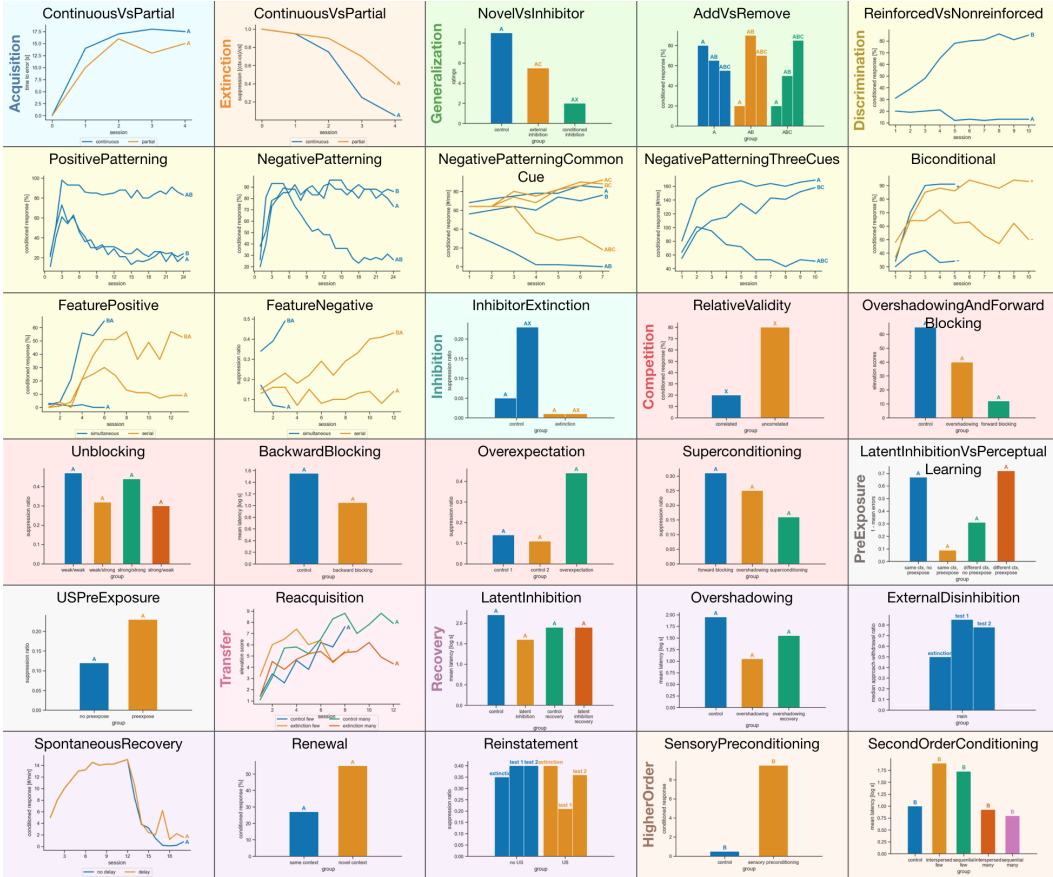


Figure 2: **Summary of simulated experiments.** The registry currently includes 30 experiments to simulate the most established classical conditioning phenomena selected from the list provided in [Alonso and Schmajuk \(2012\)](#) (see Appendix A for details). In the plots, experimental groups are separated by color, and stimuli are separated into individual bars/lines labeled with stimulus names.

In a typical classical conditioning experiment, the subject is presented with one or more conditioned stimuli (CS, e.g., a tone) followed by an unconditioned stimulus (US, e.g., food), which elicits an unconditioned response (UR, e.g., salivating) prior to training. After repeated pairing of the CS and the US over the course of multiple trials, the subject may begin to respond to the CS with what is known as the conditioned response (CR, e.g., salivating), as in the classic example of Pavlov’s dog salivating in response to the sound of the bell [\[Pavlov, 1927\]](#). The CR is a standard measure in classical conditioning experiments, and different schedules of CS stimuli result in different CR behavior, producing the wide range of classical conditioning phenomena that has been extensively cataloged [\[Alonso and Schmajuk, 2012\]](#).

## 2.2 Computational Modeling of Classical Conditioning

Reinforcement learning is deeply rooted in the study of conditioned behavior [\[Niv, 2009\]](#), dating back to Thorndike’s seminal law of effect [\[Thorndike, 1898\]](#) and Skinner’s principle of reinforcement [\[Skinner, 1935\]](#) in the early 20th century. By midcentury, the first formal treatments of animal learning were beginning to emerge in the field of mathematical psychology [\[Bush and Mosteller, 1951\]](#), paving the way for the famous Rescorla-Wagner model [\[Rescorla, 1972\]](#). The Rescorla-Wagner model provided a formal account of a wide number of puzzling phenomenon in classical conditioning, such as blocking (the phenomenon where predictable rewards are poor reinforcers) and overshadowing (the phenomenon where salient stimuli tend to form stronger associations). It was this line of research that inspired the computer scientists Sutton and Barto to develop the foundational concepts of modern reinforcement learning theory [\[Sutton and Barto, 2018\]](#). In the following decades,

a wealth of behavioral and neural evidence has further cemented the links between reinforcement learning theory and animal learning [Niv 2009], most notably the discovery that dopamine neurons in the mammalian midbrain convey signals that closely correspond to the reward prediction errors prescribed by reinforcement learning theory [Schultz et al. 1997]. These findings have in turn spurred further research in reinforcement learning [Hassabis et al. 2017], leading to some remarkable recent success of reinforcement learning algorithms across a wide number of domains, from board games and video games [Mnih et al. 2015, Silver et al. 2017, Schrittwieser et al. 2020] to robotics [Haarnoja et al. 2018, Akkaya et al. 2019] to self-driving cars [Toromanoff et al. 2020].

Following the Rescorla-Wagner model, many other computational models of classical conditioning have emerged to explain a additional phenomena: the Pearce-Hall model posits that surprising outcomes increase learning rates [Pearce and Hall 1980]; Temporal Difference models account for the temporal structure of neural learning signals [Schultz et al. 1997]; the Kalman Filter casts learning as Bayesian inference and accounts for effects of outcome uncertainty [Kakade 2001]. More recently, Temporal Difference models operating on belief states have accounted for dopamine firing in the face of partial observability [Starkweather et al. 2017], and distributional reinforcement learning has accounted for the diversity of dopamine firing in the face of different reward distributions [Dabney et al. 2020]. Notably, these models account for overlapping yet distinct sets of phenomena. Despite promising recent work combining aspects of these models [Kutlu and Schmajuk 2012b, Gershman 2015], a unifying computational theory of animal learning is still lacking.

### 2.3 Benchmarks in Computational Neuroscience

In machine learning, it has long been recognized that standardized benchmarks and datasets are essential to the development of algorithms [Deng et al. 2009, Brockman et al. 2016]. However, in psychology and neuroscience, to the best of our knowledge there still do not exist standard benchmarks for comparing theories of animal learning. [Alonso and Schmajuk 2012] provides a list of well-established classical conditioning phenomena. Subsequent work like [Kutlu and Schmajuk 2012b] has sought to evaluate models based on this list. However, a public benchmark encoding these phenomena has not until now been created.

## 3 Technical Details

CCNLab provides a unified framework for specifying and simulating classical conditioning experiments. An **experiment** in CCNLab is analogous to an RL-style environment, as it simulates the interaction between the agent model and the world. Specifically, each experiment simulates the schedule of stimuli presented in a real-world classical conditioning experiment drawn from peer-reviewed academic research, and enables evaluation of the model’s **simulated results** with the published **empirical results**.

Currently, the CCNLab registry includes 30 experiments (Figure 2) to simulate the most established classical conditioning phenomena selected from the list in [Alonso and Schmajuk 2012]. These phenomena have been collected over a diverse range of conditioning preparations (e.g., visual/auditory conditioning, taste aversion, eyeblink conditioning, fear conditioning). Across experiments, the CSs were shapes, sounds, lights, flavors, and odors; the USs were food, shock, and mechanical stimulation; and both CSs and USs covered a wide range of intensities and durations. The inter-stimulus intervals (ISI) ranged from seconds to minutes; and the inter-trial intervals (ITI) ranged from seconds to days.

An overview of CCNLab architecture is provided in Figure 1. In the following sections, we describe the technical details of CCNLab, separating information most relevant for **users** (who are primarily concerned with evaluating the performance of their models in experiment simulations) and **developers** (who are interested in extending the benchmark with additional experiments).

### 3.1 Simulation and Evaluation

Users may run their models on experiments using the provided Jupyter notebooks [Kluyver et al. 2016] or custom Python scripts. Each experiment object encapsulates functionality for: (1) generating the stimuli that serve as model inputs, (2) storing model outputs (i.e., conditioned responses), (3) summarizing model outputs in a format that is directly comparable with empirical results gathered from published work, and (4) plotting simulated and empirical results. Here is an example of the API:

```

import ccnlab.benchmarks.classical as classical
import ccnlab.evaluation as evaluation

# Select experiments to run, filtering by name using glob syntax.
for exp in classical.registry('*'):

    # Experiments often have multiple groups, each shown different stimuli.
    for g, group in exp.stimuli.items():

        # Users are free to decide how to initialize their models and how many
        # instances to allocate per group.
        for instance in range(N):
            model = YourModelHere()

            # Each group is shown a sequence of trials. At each timestep in a
            # trial, the model input consists of the conditioned stimuli (cs),
            # context (ctx), and unconditioned stimulus (us); the model output
            # is a response value.
            for i, trial in enumerate(group):
                for t, timestep in enumerate(trial):
                    cs, ctx, us = timestep
                    response = model.act(cs, ctx, us)
                    exp.data[g][i][t]['response'].append(response)

            # Simulation results can be compared to empirical results from published
            # work, via evaluation metrics (quantitative) and plots (qualitative).
            empirical = exp.empirical_results
            simulated = exp.simulated_results()
            score = evaluation.correlation(empirical, simulated)
            exp.plot()

```

**Input Representation** At each timestep in a trial, the environment provides:

1. `cs`: A list of active stimuli (string ids) and their magnitudes (positive real-valued). For most experiments, the magnitudes are either 0 or 1.
2. `ctx`: The active context (string id). The context remains the same throughout a trial.
3. `us`: The unconditioned stimulus magnitude (positive real-valued). For most experiments, the magnitude is either 0 or 1.

Alternatively, the `cs` and `ctx` are available as one-hot vectors with dimensions equal to the stimuli and context space, respectively. The 3 components of the observation are provided separately as each has unique semantic meaning. By definition, the US evokes a response in untrained subjects while the CS do not [Pavlov, 1927]. Moreover, multiple studies have demonstrated the importance of context for modulating conditioning effects to the CS [Alonso and Schmajuk, 2012, Gershman, 2017]. It is therefore common for models of classical conditioning to treat these input signals differently.

**Output Representation** At each timestep in a trial, the model should provide a response value (real-valued) indicating the strength of the conditioned response. Due to the diversity of conditioning preparations used across experiments, a degree of standardization is needed to enable the same model to perform across preparations. In particular, following [Kutlu and Schmajuk, 2012b], the response value should represent the abstract CR itself, regardless of whether it is experimentally measured as an increase or decrease of some behavior. For example, in an appetitive preparation like eyeblink conditioning where the CR is measured as an increase in blinking activity, a lower response value represents *lower* blinking activity and a higher response value represents *higher* in blinking activity, as expected. Conversely, in an aversive preparation like bar pressing where the CR is measured as a reduction in bar pressing activity, a lower response value of represents *baseline/higher* bar pressing activity and a higher response value represents *reduced/lower* bar pressing activity.

**Experiment Parameters** Each experiment simulates the schedule of stimuli presented in its real-world counterpart. Due to the diversity of conditioning preparations used across experiments, the number of conditioning trials vary significantly. By default, the experiments provide a number of trials in each phase proportional to the original work or the simulations performed in [Kutlu and Schmajuk \[2012b\]](#). However, it is possible to change the number of trials per session, if desired.

**Evaluation Metrics** We follow the approach of evaluating models using scale-invariant, ordinal measures of the quality of fit between empirical and simulated results. Following [Kutlu and Schmajuk \[2012b\]](#), we use Pearson’s correlation coefficient  $r$ , which yields a value between -1 and 1 reflecting a linear correlation between empirical and simulated results. A special case is when the empirical results contain only 2 data points, in which case we use the ratio (smaller to larger) of ratios between the data points, which yields a value between 0 and 1.

### 3.2 Specification

Developers may extend the benchmark with additional experiments from the literature or of their own design. CCNLab provides a collection of data structures and functions to make it easy to implement experiments, including: (1) an abstract syntax library for specifying experimental stimuli, (2) processing functions for computing conditioned responses and suppression ratios from the raw data, and (3) plotting functions to generate line or bar graphs.

Here is an example of the API, which has been simplified for clarity (for more details, refer to the code repository):

```
import pandas as pd
import ccnlab.benchmarks.classical.core as cc

@cc.registry.register
class Acquisition_ContinuousVsPartial(cc.ClassicalConditioningExperiment):
    def __init__(self, n=64, prob=0.5):
        # Specify stimuli structure for each experimental group using the
        # abstract syntax.
        super().__init__({
            'continuous': cc.seq(
                cc.trial('A+'),
                repeat=n, name='train'
            ),
            'partial': cc.seq(
                cc.sample({ cc.trial('A+'): prob, cc.trial('A-'): 1 - prob }),
                repeat=n, name='train',
            ),
        })

        # Encode empirical results and configure plotting.
        self.empirical_results = pd.DataFrame(
            columns=['group', 'session', 'A'],
            data=[ ... ]
        )
        self.plots = [ lambda df, ax: cc.plot_lines(df, ax=ax, x='session') ]

    # Transform raw model responses to the same format as empirical results.
    def simulated_results(self):
        df = self.dataframe(lambda x: {
            'A': cc.conditioned_response(x['timesteps'], x['response'], ['A']),
        })
        return cc.trials_to_sessions(df, self.trials_per_session)
```

**Abstract Syntax Library** Each experiment presents a schedule of stimuli consisting of multiple trials per group. To facilitate the specification of stimuli structure, we developed an abstract syntax library that allows classical conditioning experiments to be expressed in a consistent way. By



composing a sequence of nodes, developers may specify the structure using an abstract syntax tree, which can then be compiled into the a sequence of trials, each consisting of multiple timesteps. The syntax closely conforms to standard classical conditioning notation, using the following nodes:

- **Stimulus**: Leaf node specifying the presentation of a stimulus (string), its magnitude (float), and its start and end timesteps in a trial (ints).
- **Trial**: Compound node specifying a trial consisting of the CS (list of `Stimulus`), CTX (string), and US (`Stimulus`);
- **Sample**: Compound node specifying probabilities with which to choose each of its children.
- **Sequence**: Compound node specifying a sequence of nodes, how many times to repeat the sequence, and a name (used for naming different phases of the schedule).

**Empirical Results** The published work that the experiments simulate present their results in the form of summary statistics. Typically, the measure of interest is the conditioned response or a suppression ratio, and this measure is plotted over trials/sessions (line plots) or over groups (bar plots). For data collection, we relied on numerical tables in published work if available; otherwise, we digitally enlarged the plots and used a grid overlay to more precisely estimate the measure values.

**Simulated Results** In order to compare simulated results to the empirical results, the raw model responses must first be transformed into the same summary statistics as the empirical results. We are given a set of active conditioned stimuli  $cs_t$  and model responses  $response_t$  for each timestep  $0 \leq t < T$  in a single trial.

For appetitive preparations, the conditioned response to a stimulus  $x$  is the average response during the presentation of  $x$ .

$$\text{ConditionedResponse}(x) = \frac{\sum_{t:cs_t=x} response_t}{\sum_{t:cs_t=x} 1}$$

For aversive preparations, the suppression ratio to a stimulus  $x$  is the ratio of (reduced) responding during the presentation of  $x$  compared to baseline responding. As described in Section 3.1, the abstract response value must be inverted to correspond to aversive CR behavior. We use the maximum response value for inversion as in Kutlu and Schmajuk [2012b].

$$\text{SuppressionRatio}(x) = \frac{\sum_{t:cs_t=x} (\max_t response_t) - response_t}{\sum_t (\max_t response_t) - response_t}$$

Finally, after computing the measure for each trial, the measures are aggregated across subjects through averaging. If necessary, consecutive spans of trials are aggregated into sessions through averaging.

## 4 Baselines and Experiments

To illustrate how the benchmark can be used in practice, we applied it to compare 3 classical conditioning baseline models from Gershman [2015]. For more details on the models and their parameters, see Appendix A.

- **Rescorla-Wagner** [Rescorla, 1972]: The predicted reward (US) is a linear combination of the input stimuli (CS), weighted by their associative weights. The weights are updated in proportion to the reward prediction error – the difference between the predicted and actual reward – and how active the stimuli are.
- **Kalman Filter** [Kakade, 2001]: A Bayesian extension of Rescorla-Wagner which learns the covariance of the weights in addition to their mean. The learning rate is dependent on the uncertainty encoded by the covariance.
- **Temporal Difference Learning** [Sutton, 1988]: A temporal extension of Rescorla-Wagner which predicts cumulative future reward instead of immediate reward only.

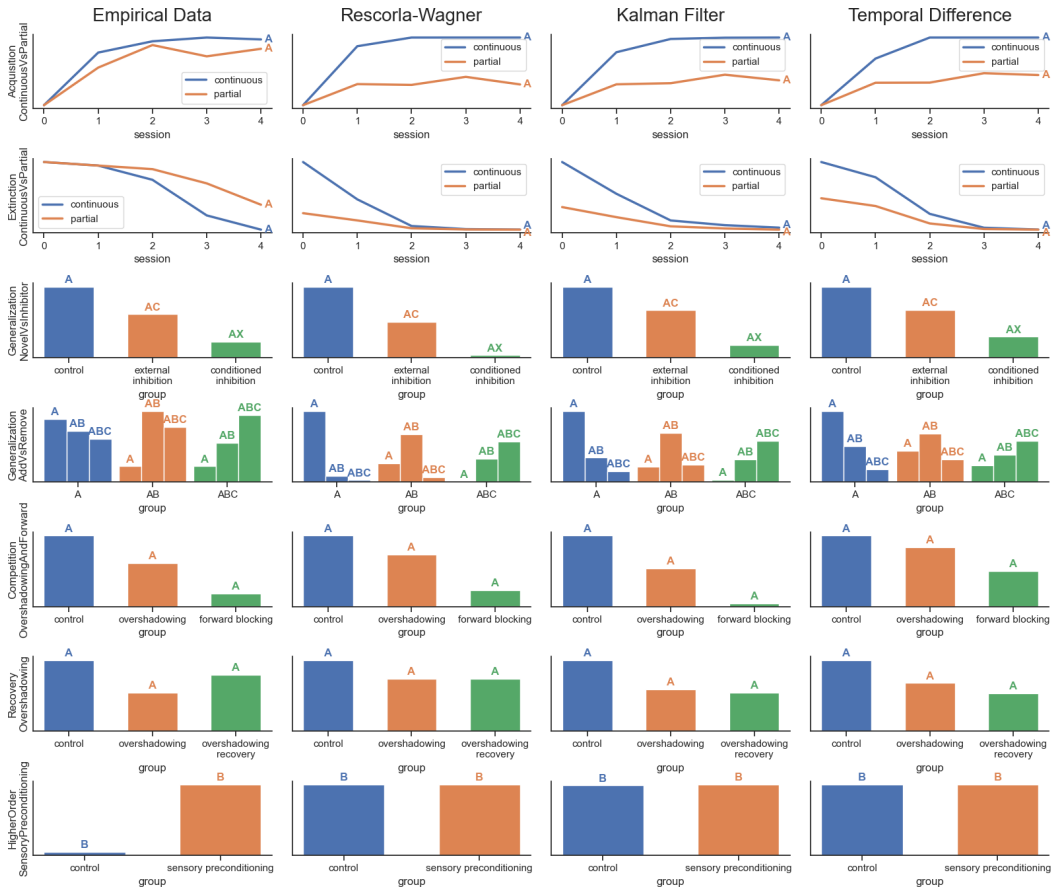


Figure 3: **Plots for selected experiments.** Models can be evaluated qualitatively by comparing plots for empirical and simulated results. Column 1: Empirical results from published work. Columns 2-4: Simulated results for each model. Plot coloring and notation are the same as Figure 2

We evaluated the models based on how well they fit the empirical results for each experiment according to the metrics in Section 3.1. Scores for all experiments are presented in Table 1, and scores averaged across categories are presented in Table 2. We also plotted the empirical and simulated results, for which a subset of experiments are shown in Figure 3.

Unsurprisingly, Rescorla-Wagner is able to account for phenomena it was designed to explain, such as acquisition (Figure 3, Row 1), extinction (Row 2), external and conditioned inhibition (Row 3), generalization with added and removed cues (Row 4), and overshadowing and forward blocking (Row 5). Kalman Filter and Temporal Difference Learning are also able to account for these phenomena well, since they are generalizations of Rescorla-Wagner. In addition, they each account better for other phenomena, such as extinction of inhibition for Kalman Filter; and overexpectation, superconditioning, and higher-order conditioning for Temporal Difference Learning, giving them higher scores overall (Table 2). Notably, all models perform poorly on most experiments in the benchmark, and even show negative correlations on a substantial number of experiments (Table 1). This could be partially explained by the fact that model parameters were not tuned to each experiment, although we found that performance did not change substantially with different parameter settings (data not shown). We take this to highlight the limitations of these standard learning models which are still widely used in neuroscience, suggesting a pressing need for the development and adoption of more general theories of learning.



Table 1: **Scores for all experiments.** Models can be evaluated quantitatively by computing measures of fit between empirical and simulated summary statistics. These scores use the Pearson correlation, except for experiments indicated by \* which use ratio of ratios instead (see Section 3.1). Scores greater than 0.8 are highlighted in bold.

Category	Experiment	Rescorla-Wagner	Kalman Filter	Temporal Difference
Acquisition	ContinuousVsPartial	<b>0.80</b>	<b>0.83</b>	<b>0.85</b>
Extinction	ContinuousVsPartial	0.54	0.57	0.69
Generalization	NovelVsInhibitor	<b>1.00</b>	<b>0.99</b>	<b>1.00</b>
	AddVsRemove	0.60	0.75	0.60
Discrimination	ReinforcedVsNonreinforced	-0.88	-0.88	-0.89
	PositivePatterning	-0.82	<b>0.88</b>	<b>0.89</b>
	NegativePatterning	-0.64	-0.63	0.75
	NegativePatterningCommonCue	0.15	0.02	0.74
	NegativePatterningThreeCues	-0.76	-0.70	0.46
	Biconditional	0.23	0.33	0.68
	FeaturePositive	-0.30	-0.07	0.11
	FeatureNegative	0.33	0.42	0.28
Inhibition	InhibitorExtinction	-0.36	<b>0.99</b>	0.48
Competition	RelativeValidity*	0.00	0.00	0.00
	OvershadowingAndForwardBlocking	<b>0.99</b>	<b>1.00</b>	<b>0.99</b>
	Unblocking	-0.65	-0.65	-0.17
	BackwardBlocking*	0.17	0.12	0.77
	Overexpectation	-1.00	-0.99	<b>0.87</b>
	Superconditioning	-0.77	-0.65	<b>0.88</b>
PreExposure	LatentInhibitionVsPerceptualLearning	0.00	0.00	0.00
	USPreExposure*	<b>0.81</b>	0.62	<b>0.96</b>
Transfer	Reacquisition	0.72	0.74	0.63
Recovery	LatentInhibition	0.00	0.01	0.00
	Overshadowing	<b>0.83</b>	0.78	0.62
	ExternalDisinhibition	0.69	0.58	0.40
	SpontaneousRecovery	<b>0.97</b>	<b>0.93</b>	0.56
	Renewal*	0.00	0.00	0.00
	Reinstatement	-0.71	-0.70	-0.82
HigherOrder	SensoryPreconditioning*	0.00	0.00	0.05
	SecondOrderConditioning	0.01	0.17	0.49

Table 2: **Scores for all categories.** Averages across categories of the scores in Table 1. Scores greater than 0.8 are highlighted in bold.

Category	Rescorla-Wagner	Kalman Filter	Temporal Difference
Acquisition	<b>0.80</b>	<b>0.83</b>	<b>0.85</b>
Extinction	0.54	0.57	0.69
Generalization	<b>0.80</b>	<b>0.87</b>	<b>0.80</b>
Discrimination	-0.34	-0.08	0.38
Inhibition	-0.36	<b>0.99</b>	0.48
Competition	-0.21	-0.20	0.56
PreExposure	0.41	0.31	0.48
Transfer	0.72	0.74	0.63
Recovery	0.30	0.27	0.13
HigherOrder	0.01	0.09	0.27
Overall	0.27	0.44	0.53

## 5 Conclusion and Future Work

We presented CCNLab, an open-source benchmark for computational modeling of classical conditioning. To the best of our knowledge, this is the first benchmark of its kind for evaluating and comparing computational models in neuroscience and psychology. We hope that it will encourage the development of broader neuroscientific theories of learning, as well as the development of powerful artificial intelligence algorithms inspired by the brain. In the following sections, we suggest future directions for the benchmark and for research using the benchmark.

**Benchmark Extensions** While the presented benchmark is a step forward for the rigorous evaluation of computational classical conditioning models, there are a number of extensions that would improve it. (1) Currently, it contains a selection of experiments demonstrating the most well established phenomena. We expect the registry of experiments to grow over time with contributions from the authors and the community, adding experiments that capture a wider breadth of phenomena, especially regarding multi-modal combination and temporal effects that are presently missing. (2) There is significant diversity in the types of conditioning preparations reflected by the experiments. For more a standardized comparison, it would be useful to capture the phenomena in a single preparation, or show the same phenomena across multiple preparations. (3) Currently, we only allow a single US and UR, but the model should likely be extended handle multiple. (4) While current experiments test the fit of simulated and empirical behavioral data, it is ultimately important for models of classical conditioning to also capture neural data.

**Research Directions** (1) From our experiments it is evident that computational models of classical conditioning still fail to explain a breadth of phenomena. We hope that the introduction of this benchmark will encourage future research to consider a broader range of phenomena. (2) We hope that CCNLab is only the first step towards widespread benchmarking in computational neuroscience. In the future, we expect that benchmarks for different domains will be added, for instance to capture the diverse phenomena of instrumental conditioning and short-term/working memory [Oberauer et al., 2018]. (3) Finally, we believe that it will continue to be productive for the fields of artificial intelligence and computational neuroscience to build on each other’s insights, simultaneously approaching the project of intelligence from both a scientific and engineering standpoint.

## Acknowledgments and Disclosure of Funding

This work was supported by grants from the NIH (U19 NS113201-01), the Toyota Corporation, the Air Force Office of Scientific Research (FA9550-20-1-0413), and the Center for Brains, Minds and Machines (CBMM), funded by NSF STC award CCF-1231216. The funders had no role in study design, data collection and analysis, decision to publish or preparation of the manuscript. The authors have no competing interests to report.

## References

- I. Akkaya, M. Andrychowicz, M. Chociej, M. Litwin, B. McGrew, A. Petron, A. Paino, M. Plappert, G. Powell, R. Ribas, et al. Solving rubik’s cube with a robot hand. *arXiv preprint arXiv:1910.07113*, 2019.
- E. Alonso and N. A. Schmajuk. Special issue on computational models of classical conditioning guest editors’ introduction. *Learning & Behavior*, 40(3):231–240, 2012.
- W. Bellingham, K. Gillette-Bellingham, and E. Kehoe. Summation and configuration in patterning schedules with the rat and rabbit. *Animal Learning & Behavior*, 13:152–164, 1985.
- S. Bottjer. Conditioned approach and withdrawal behavior in pigeons: Effects of a novel extraneous stimulus during acquisition and extinction. *Learning and Motivation*, 13:44–67, 1982.
- S. Brandon, E. H. Vogel, and A. R. Wagner. A componential view of configural cues in generalization and discrimination in pavlovian conditioning. *Behavioural Brain Research*, 110:67–72, 2000.
- G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.
- W. Brogden. Sensory pre-conditioning. *Journal of Experimental Psychology*, 25:323–332, 1939.

- R. R. Bush and F. Mosteller. A mathematical model for simple learning. *Psychological review*, 58(5):313, 1951.
- M. M. Campolattaro, K. M. Schnitker, and J. H. Freeman. Changes in inhibition during differential eyeblink conditioning with increased training. *Learn Behav*, 36(2):159–165, May 2008.
- W. Dabney, Z. Kurth-Nelson, N. Uchida, C. K. Starkweather, D. Hassabis, R. Munos, and M. Botvinick. A distributional code for value in dopamine-based reinforcement learning. *Nature*, 577(7792):671–675, 2020.
- J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- A. Dickinson, G. Hall, and N. Mackintosh. Surprise and the attenuation of blocking. *Journal of Experimental Psychology: Animal Behavior Processes*, 2:313–322, 1976.
- D. Durstewitz, G. Koppe, and H. Toutounji. Computational models as statistical tools. *Current Opinion in Behavioral Sciences*, 11:93–99, 2016.
- O. Ertugrul and M. Tagluk. A review of computational classical conditioning models. *American Journal of Psychology and Behavioral Sciences*, 2:36–44, 03 2015.
- S. J. Gershman. A unifying probabilistic view of associative learning. *PLoS Comput Biol*, 11(11):e1004567, 2015.
- S. J. Gershman. Context-dependent learning and causal structure. *Psychonomic Bulletin & Review*, 24:557–565, 2017.
- S. J. Gershman. Just looking: the innocent eye in neuroscience. *Neuron*, 2021.
- N. Grahame, R. Barnet, L. M. Gunther, and R. Miller. Latent inhibition as a performance deficit resulting from cs—context associations. *Animal Learning & Behavior*, 22:395–408, 1994.
- T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.
- J. A. Harris, M. L. Jones, G. K. Bailey, and R. F. Westbrook. Contextual control over conditioned responding in an extinction paradigm. *J Exp Psychol Anim Behav Process*, 26(2):174–185, Apr 2000.
- D. Hassabis, D. Kumaran, C. Summerfield, and M. Botvinick. Neuroscience-inspired artificial intelligence. *Neuron*, 95(2):245–258, 2017.
- K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- P. Holland. Differential effects of reinforcement of an inhibitory feature after serial and simultaneous feature negative discrimination training. *Journal of experimental psychology. Animal behavior processes*, 10 4: 461–75, 1984.
- P. C. Holland and G. D. Fox. Effects of hippocampal lesions in overshadowing and blocking procedures. *Behav Neurosci*, 117(3):650–656, Jun 2003.
- E. Jonas and K. P. Kording. Could a neuroscientist understand a microprocessor? *PLoS computational biology*, 13(1):e1005268, 2017.
- P. D. S. Kakade. Explaining away in weight space. *Advances in neural information processing systems*, 13:451, 2001.
- L. Kamin. Trace conditioning of the conditioned emotional response. *Journal of comparative and physiological psychology*, 54:149–53, 1961.
- T. Kluyver, B. Ragan-Kelley, F. Pérez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla, and C. Willing. Jupyter notebooks – a publishing format for reproducible computational workflows. In F. Loizides and B. Schmidt, editors, *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87 – 90. IOS Press, 2016.
- N. Kriegeskorte and P. K. Douglas. Cognitive computational neuroscience. *Nature neuroscience*, 21(9): 1148–1160, 2018.
- M. G. Kutlu and N. Schmajuk. Deactivation and reactivation of the inhibitory power of a conditioned inhibitor: Testing the predictions of an attentional-associative model. *Learning & Behavior*, 40:83–97, 2012a.

- M. G. Kutlu and N. A. Schmajuk. Solving pavlov's puzzle: Attentional, associative, and flexible configural mechanisms in classical conditioning. *Learning & Behavior*, 40(3):269–291, 2012b.
- B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017.
- Y. LeCun, Y. Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.
- D. Levenstein, V. A. Alvarez, A. Amarasingham, H. Azab, R. C. Gerkin, A. Hasenstaub, R. Iyer, R. B. Jolivet, S. Marzen, J. D. Monaco, et al. On the role of theory and modeling in neuroscience. *arXiv preprint arXiv:2003.13825*, 2020.
- S. W. Linderman and S. J. Gershman. Using computational theory to constrain statistical models of neural data. *Current opinion in neurobiology*, 46:14–24, 2017.
- R. Lubow, B. Rifkin, and M. Alek. The context effect: The relationship between stimulus preexposure and environmental preexposure determines subsequent learning. *Journal of Experimental Psychology: Animal Behavior Processes*, 2:38–47, 1976.
- W. J. Ma and B. Peters. A neural network walks into a lab: towards using deep nets as models for human behavior. *arXiv preprint arXiv:2005.02181*, 2020.
- L. Matzel, T. Schachtman, and R. Miller. Recovery of an overshadowed association achieved by extinction of the overshadowing stimulus. *Learning and Motivation*, 16:398–412, 1985.
- R. Miller and H. Matute. Biological significance in forward and backward blocking: resolution of a discrepancy between animal conditioning and human causal judgment. *Journal of experimental psychology. General*, 125 4:370–86, 1996.
- V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540): 529–533, 2015.
- P. R. Montague, P. Dayan, and T. J. Sejnowski. A framework for mesencephalic dopamine systems based on predictive hebbian learning. *Journal of neuroscience*, 16(5):1936–1947, 1996.
- Y. Niv. Reinforcement learning in the brain. *Journal of Mathematical Psychology*, 53(3):139–154, 2009.
- K. Oberauer, S. Lewandowsky, E. Awh, G. D. A. Brown, A. Conway, N. Cowan, C. Donkin, S. Farrell, G. J. Hitch, M. J. Hurlstone, W. J. Ma, C. C. Morey, D. E. Nee, J. Schwenke, E. Vergauwe, and G. Ward. Benchmarks for models of short-term and working memory. *Psychol Bull*, 144(9):885–958, 09 2018.
- P. I. Pavlov. Conditioned reflexes: an investigation of the physiological activity of the cerebral cortex. 1927.
- J. M. Pearce and G. Hall. A model for pavlovian learning: variations in the effectiveness of conditioned but not of unconditioned stimuli. *Psychological review*, 87(6):532, 1980.
- E. Redhead and J. Pearce. Similarity and discrimination learning. *Quarterly Journal of Experimental Psychology*, 48:46 – 66, 1995.
- E. Redhead and J. Pearce. Some factors that determine the influence of a stimulus that is irrelevant to a discrimination. *Journal of Experimental Psychology: Animal Behavior Processes*, 24:123–135, 1998.
- R. Rescorla. Reduction in the effectiveness of reinforcement after prior excitatory conditioning. *Learning and Motivation*, 1:372–381, 1970.
- R. Rescorla. Variation in the effectiveness of reinforcement and nonreinforcement following prior inhibitory conditioning. *Learning and Motivation*, 2:113–123, 1971.
- R. A. Rescorla. A theory of pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Current research and theory*, pages 64–99, 1972.
- R. A. Rescorla. Spontaneous recovery. *Learn Mem*, 11(5):501–509, 2004.
- R. A. Rescorla and C. D. Heth. Reinstatement of fear to an extinguished conditioned stimulus. *J Exp Psychol Anim Behav Process*, 1(1):88–96, Jan 1975.
- S. Ricker and M. Bouton. Reacquisition following extinction in appetitive conditioning. *Animal Learning & Behavior*, 24:423–436, 1996.

- R. Ross and P. Holland. Conditioning of simultaneous and serial feature-positive discriminations. *Animal Learning & Behavior*, 9:293–303, 1981.
- M. A. Saavedra. Pavlovian compound conditioning in the rabbit. *Learning and Motivation*, 6:314–326, 1975.
- A. Saxe, S. Nelli, and C. Summerfield. If deep learning is the answer, what is the question? *Nature Reviews Neuroscience*, pages 1–13, 2020.
- J. Schrittwieser, I. Antonoglou, T. Hubert, K. Simonyan, L. Sifre, S. Schmitt, A. Guez, E. Lockhart, D. Hassabis, T. Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839): 604–609, 2020.
- W. Schultz, P. Dayan, and P. R. Montague. A neural substrate of prediction and reward. *Science*, 275(5306): 1593–1599, 1997.
- D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.
- B. F. Skinner. Two types of conditioned reflex and a pseudo type. *The Journal of General Psychology*, 12(1): 66–77, 1935.
- K. L. Stachenfeld, M. M. Botvinick, and S. J. Gershman. The hippocampus as a predictive map. *Nature neuroscience*, 20(11):1643, 2017.
- C. K. Starkweather, B. M. Babayan, N. Uchida, and S. J. Gershman. Dopamine reward prediction errors reflect hidden-state inference across time. *Nature neuroscience*, 20(4):581–589, 2017.
- R. S. Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44, 1988.
- R. S. Sutton and A. G. Barto. Time-derivative models of pavlovian reinforcement. 1990.
- R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- E. L. Thorndike. Animal intelligence: an experimental study of the associative processes in animals. *The Psychological Review: Monograph Supplements*, 2(4):i, 1898.
- M. Toromanoff, E. Wirbel, and F. Moutarde. End-to-end model-free reinforcement learning for urban driving using implicit affordances. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7153–7162, 2020.
- A. Wagner, L. Siegel, and G. Fein. Extinction of conditioned fear as a function of percentage of reinforcement. *Journal of comparative and physiological psychology*, 63:160–4, 03 1967. doi: 10.1037/h0024172.
- A. R. Wagner, F. Logan, K. Haberlandt, and T. Price. Stimulus selection in animal discrimination learning. *Journal of experimental psychology*, 76 2:171–80, 1968.
- D. L. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the national academy of sciences*, 111(23):8619–8624, 2014.
- H. Yin, R. C. Barnett, and R. R. Miller. Second-order conditioning and Pavlovian conditioned inhibition: operational similarities and differences. *J Exp Psychol Anim Behav Process*, 20(4):419–428, Oct 1994.
- C. L. Zimmer-Hart and R. Rescorla. Extinction of pavlovian conditioned inhibition. *Journal of comparative and physiological psychology*, 86 5:837–45, 1974.