Neural Computation Projects

Overview

These projects are designed as learning-oriented exercises in neural computation, emphasizing understanding over perfect implementation. Each project should be completable in approximately 2-3 days of focused work. You are encouraged to work in groups, though individual reports (4-6 pages, or 8-10 pages if including Jupyter notebooks) must be submitted by Monday, 9 AM of finals week. The oral exam (15 minutes) will consist of a 5-minute presentation of your key findings using one visual aid, followed by questions about both your project and the course material. If you encounter difficulties reproducing published results—which is common in cutting-edge research—document your process, propose hypotheses for the challenges, and reflect on what you learned. The goal is to demonstrate independent work, understanding of neural computation concepts, and ability to scope problems appropriately. Additional project proposals may be submitted by Thursday if the listed topics don't align with your interests.

1. Online Perceptron/Max Margin

Implement online classification using the perceptron algorithm and compare it with the passive-aggressive algorithm (specifically the simplest version that behaves like a perceptron). Begin by recapping the theoretical foundations of the perceptron algorithms, including convergence guarantees and geometric interpretations. Demonstrate how the passive-aggressive algorithm addresses the perceptron's weaknesses, particularly in handling overlapping, noisy data—a crucial consideration for neural computation. For scoping, use a simple (possibly synthetic) 2-D Dataset that you can use to visualize the difference between the two approaches.

Starting References:

- Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., & Singer, Y. (2006). Online passive-aggressive algorithms. Journal of Machine Learning Research, 7(Mar), 551-585.
- Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. Psychological Review, 65(6), 386.

2. Hodgkin-Huxley Dynamics

Explore computational implications of different neural firing patterns by implementing models with various ion channel combinations. Select at least two phenomena such as spike-frequency adaptation, variable firing thresholds, bursting, or subthreshold oscillations. Analyze how these different dynamics could support distinct computational functions in neural circuits, such as enabling NMDA spikes through bursts or supporting oscillatory phase coding. Alternative approach: implement the Izhikevich model, which more reliably generates diverse spiking behaviors. For scoping, focus on finding published parameter sets rather than extensive parameter exploration, as achieving specific spiking behaviors can be challenging.

Starting References:

- Pospischil, M., Toledo-Rodriguez, M., Monier, C., Piwkowska, Z., Bal, T., Frégnac, Y., ... & Destexhe, A. (2008). Minimal Hodgkin–Huxley type models for different classes of cortical and thalamic neurons. Biological Cybernetics, 99(4), 427-441.
- Gerstner, W., Kistler, W.M., Naud, R. and Paninski, L., 2014. Neuronal dynamics: From single neurons to networks and models of cognition. Cambridge University Press.
- Izhikevich, E. M. (2003). Simple model of spiking neurons. IEEE Transactions on Neural Networks, 14(6), 1569-1572.

3. Inhibitory Synapse Types

Investigate the computational roles of different inhibitory synapse types through simulation. Compare shunting (divisive/gain-modulating) inhibition with subtractive (hyperpolarizing) inhibition with respect to neural computation, or examining how their placement on the dendritic tree affects neural computation. Implement a small model showing how each type modulates excitatory inputs differently, discussing implications for, for example, gain control, normalization, and selective gating of information. You may consider how distally placed inhibitory synapses can effectively shut off entire dendritic branches. For scoping, vary inhibition parameters and types in a simple model with controlled inputs.

Starting References:

- Koch, C. (2004). Biophysics of computation: information processing in single neurons. Oxford University Press. (Chapters 4-5)
- Doiron, B., Longtin, A., Berman, N., & Maler, L. (2001). Subtractive and divisive inhibition: effect of voltage-dependent inhibitory conductances and noise. Neural Computation, 13(1), 227-248.

4. NMDA Receptor Dynamics

Explore active synaptic computation through NMDA receptor dynamics. Simulate the voltage-dependent magnesium block and coincidence detection properties of NMDA receptors at a single synapse level, showing how they require both presynaptic glutamate release and postsynaptic depolarization. Create experiments demonstrating when NMDA channels open versus remain blocked. Discuss how these properties enable dendritic computation and synaptic plasticity, potentially implementing a simple STDP mechanism where NMDA activation is triggered by postsynaptic spikes or show non-linear integration. For scoping, focus on modeling a few synapses with clear input regimes rather than network-level dynamics to demonstrate the impact of NMDARr channels.

Starting References:

- Koch, C. (2004). Biophysics of computation: information processing in single neurons. Oxford University Press. (Chapter 4)
- Jahr, C. E., & Stevens, C. F. (1990). Voltage dependence of NMDA-activated macroscopic conductances predicted by single-channel kinetics. Journal of Neuroscience, 10(9), 3178-3182.

5. Tempotron

Implement the Tempotron model for classifying spatiotemporal spike patterns. Design experiments measuring classification performance at various pattern loads (α), exploring how membrane time constant (τ_m), synaptic time constant (τ_s), and pattern duration (T) affect learning capacity. Additionally, test the algorithm on rate-coded Poisson spike patterns where neurons are randomly assigned "high" (e.g., 20 Hz) or "low" (e.g., 10 Hz) firing rates for 500ms, evaluating whether the Tempotron can learn this rate-based classification despite its timing-focused design. Note that reproducing exact published results can be challenging; focus on understanding the model's behavior and limitations.

Starting References:

- Gütig, R., & Sompolinsky, H. (2006). The tempotron: a neuron that learns spike timing–based decisions. Nature Neuroscience, 9(3), 420-428.
- Gütig, R. (2016). Spiking neurons can discover predictive features by aggregate-label learning. Science, 351(6277), aab4113.

6. BCM Learning Rule

Implement the Bienenstock-Cooper-Munro (BCM) learning rule featuring a sliding modification threshold. Demonstrate through simulations how the threshold θ_M adapts based on postsynaptic activity history, preventing runaway potentiation or depression. Show how BCM naturally develops orientation selectivity in visual cortex models using either natural image patches or synthetic oriented gratings as input. Compare its stability properties with standard Hebbian learning and Oja's rule, discussing how BCM performs covariance-based rather than correlation-based learning.

Starting References:

- Cooper, L. N., Intrator, N., Blais, B. S., & Shouval, H. Z. (2004). Theory of cortical plasticity. World Scientific.
- Bienenstock, E. L., Cooper, L. N., & Munro, P. W. (1982). Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. Journal of Neuroscience, 2(1), 32-48.

7. STDP Variants

Investigate diverse forms of spike-timing-dependent plasticity beyond classical asymmetric STDP. Implement an alternative STDP curve or mechanism including symmetric, inverted, or multiplicative STDP and compare it to the classical implementation. For scoping, design controlled small-scale experiments that demonstrate specific computational properties—avoid attempting to solve complex problems. You can also consider models with weight decay or self-regulation mechanisms. Optionally explore fast synaptic plasticity models like Tsodyks' model for rhythm generation, but scope carefully to avoid complexity rabbit holes.

Starting References:

- Caporale, N., & Dan, Y. (2008). Spike timing–dependent plasticity: a Hebbian learning rule. Annual Review of Neuroscience, 31, 25-46.
- Pfister, J. P., & Gerstner, W. (2006). Triplets of spikes in a model of spike timing-dependent plasticity. Journal of Neuroscience, 26(38), 9673-9682.
- Roelfsema, P.R. and Holtmaat, A., 2018. Control of synaptic plasticity in deep cortical networks. Nature Reviews Neuroscience, 19(3), pp.166-180.

8. Information Bottleneck

Implement the information bottleneck method to understand neural compression principles. Track mutual information I(X;T) between input and neural representation, and I(T;Y) between representation and target during learning. Create visualizations showing compression dynamics. For scoping, use a simple classification task like detecting zeros in MNIST or construct synthetic datasets with clear coding. Avoid complex multi-neuron networks, implement at least one single neuron example. Discuss connections to deep learning debates (2019-2021) about whether this framework applies universally to artificial neural networks, and contrast it with discussions around real neural networks and theoretical neuroscience (e.g. predictive coding)

Starting References:

- Tishby, N., Pereira, F. C., & Bialek, W. (2000). The information bottleneck method. arXiv preprint physics/0004057.

- Shwartz-Ziv, R., & Tishby, N. (2017). Opening the black box of deep neural networks via information. arXiv preprint arXiv:1703.00810.

9. Dendritic Computation

Explore dendritic trees as computational units beyond passive integration. Choose either: (a) Implement sparse "dendrite networks" where hidden units represent dendrites connected to specific spatial patches or random subsets of inputs, testing how this structural prior affects learning (based on Poirazi group's recent work), or (b) Implement dendritic delay-based sound localization using cable properties, where synapses at different dendritic locations introduce time delays enabling coincidence detection for directional hearing. For the temporal version, consider using Brian simulator for cable modeling. Scope carefully—aim for proof-of-concept demonstrations rather than large experiments.

Starting References:

- London, M., & Häusser, M. (2005). Dendritic computation. Annual Review of Neuroscience, 28, 503-532.
- Chavlis, S. and Poirazi, P., 2025. Dendrites endow artificial neural networks with accurate, robust and parameter-efficient learning. Nature Communications, 16(1), p.943.
- Agmon-Snir, H., Carr, C. E., & Rinzel, J. (1998). The role of dendrites in auditory coincidence detection. Nature, 393(6682), 268-272.

10. Random Feedback Alignment

Implement random feedback alignment (RFA) as a biologically plausible alternative to backpropagation. Demonstrate how fixed random feedback weights can support learning in multi-layer networks (2-3 hidden layers, 100-500 units) on a reduced MNIST task (take 20,000 examples or so). Extend with input whitening (decorrelating inputs) to improve gradient alignment. Optionally implement gradient normalization or other techniques that modify the gradient signal to improve feedback alignment. Discuss biological plausibility compared to backpropagation, particularly connections to dopamine's diffuse reward signaling.

Starting References:

- Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J. (2016). Random synaptic feedback weights support error backpropagation for deep learning. Nature Communications, 7(1), 1-10.
- Nøkland, A. (2016). Direct feedback alignment provides learning in deep neural networks. Advances in Neural Information Processing Systems, 29.

Complete Reference List

1. Agmon-Snir, H., Carr, C. E., & Rinzel, J. (1998). The role of dendrites in auditory coincidence detection. Nature, 393(6682), 268-272.
2. Bienenstock, E. L., Cooper, L. N., & Munro, P. W. (1982). Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. Journal of Neuroscience, 2(1), 32-48.
3. Caporale, N., & Dan, Y. (2008). Spike timing–dependent plasticity: a Hebbian learning rule. Annual Review of Neuroscience, 31, 25-46.
4. Chavlis, S. and Poirazi, P., 2025. Dendrites endow artificial neural networks with accurate, robust and parameter-efficient learning. Nature Communications, 16(1), p.943.
5. Cooper, L. N., Intrator, N., Blais, B. S., & Shouval, H. Z. (2004). Theory of cortical plasticity. World Scientific.
6. Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., & Singer, Y. (2006). Online passive-aggressive algorithms. Journal of Machine Learning Research, 7(Mar), 551-585.
7. Doiron, B., Longtin, A., Berman, N., & Maler, L. (2001). Subtractive and divisive inhibition: effect of voltage-dependent inhibitory conductances and noise. Neural Computation, 13(1), 227-248.
8. Gerstner, W., Kistler, W.M., Naud, R. and Paninski, L., 2014. Neuronal dynamics: From single neurons to networks and models of cognition. Cambridge University Press.
9. Gütig, R. (2016). Spiking neurons can discover predictive features by aggregate-label learning. Science, 351(6277), aab4113.
10. Gütig, R., & Sompolinsky, H. (2006). The tempotron: a neuron that learns spike timing–based decisions. Nature Neuroscience, 9(3), 420-428.
11. Izhikevich, E. M. (2003). Simple model of spiking neurons. IEEE Transactions on Neural Networks, 14(6), 1569-1572.
12. Jahr, C. E., & Stevens, C. F. (1990). Voltage dependence of NMDA-activated macroscopic conductances predicted by single-channel kinetics. Journal of Neuroscience, 10(9), 3178-3182.
13. Koch, C. (2004). Biophysics of computation: information processing in single neurons. Oxford University Press.
14. Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J. (2016). Random synaptic feedback weights support error backpropagation for deep learning. Nature Communications, 7(1), 1-10.
15. London, M., & Häusser, M. (2005). Dendritic computation. Annual Review of Neuroscience, 28, 503-532.
16. Nøkland, A. (2016). Direct feedback alignment provides learning in deep neural networks. Advances in Neural Information Processing Systems, 29.
17. Pfister, J. P., & Gerstner, W. (2006). Triplets of spikes in a model of spike timing-dependent plasticity. Journal of Neuroscience, 26(38), 9673-9682.

18. Pospischil, M., Toledo-Rodriguez, M., Monier, C., Piwkowska, Z., Bal, T., Frégnac, Y., ... & Destexhe, A. (2008). Minimal Hodgkin–Huxley type models for different classes of cortical and thalamic neurons. Biological Cybernetics, 99(4), 427-441.
19. Roelfsema, P.R. and Holtmaat, A., 2018. Control of synaptic plasticity in deep cortical networks. Nature Reviews Neuroscience, 19(3), pp.166-180.
20. Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. Psychological Review, 65(6), 386.
21. Shwartz-Ziv, R., & Tishby, N. (2017). Opening the black box of deep neural networks via information. arXiv preprint arXiv:1703.00810.
22. Tishby, N., Pereira, F. C., & Bialek, W. (2000). The information bottleneck method. arXiv preprint physics/0004057.