

The Neural Bridge: From Haptic Cords to Python Code — A Multi-Abstraction Instructional Sequence for Teaching Neural Networks in K-12

Frauke Ritter

German International School Silicon Valley
Mountain View, California, USA
fritter@gissv.org

Ivo Betten

German International School Silicon Valley
Mountain View, California, USA
ibetten@gissv.org

Lukas Lehner

TU Wien
Vienna, Austria
lukas.lehner@tuwien.ac.at

Sebastian Berk

German International School Silicon Valley
Mountain View, California, USA
sberk@gissv.org

Abstract

Teaching neural networks at the secondary level is frequently impeded by the conceptual gap between everyday intuition and implementation-level code. This paper presents *The Neural Bridge*, a four-lesson instructional sequence for K-12 students that traverses three levels of abstraction: an embodied unplugged simulation using physical cords and carabiners (“7 Little Neurons”), a manual computation phase in Google Sheets, and a professional implementation in Google Colab. The sequence integrates Productive Failure [2] and the PRIMM framework [9] as its pedagogical scaffolding, and is grounded in Long and Magerko’s [8] seventeen AI literacy competencies. Delivered to 27 students across Grades 10–12 in December 2025, a pre-/post-study showed that correct identification of AI as a data-processing system increased from 63% to 82%. Notably, students’ self-assessed AI understanding *decreased* ($M=2.70$ to $M=2.00$), a positive outcome interpreted as metacognitive growth consistent with the Dunning-Kruger effect. Qualitative feedback identified the unplugged activity as the most impactful element.

Keywords

Neural Networks, K-12 Education, AI Literacy, Productive Failure, PRIMM, Unplugged Activities, Google Colab, Mixed Methods, Instructional Design

ACM Reference Format:

Frauke Ritter, Lukas Lehner, Ivo Betten, and Sebastian Berk. 2026. The Neural Bridge: From Haptic Cords to Python Code — A Multi-Abstraction Instructional Sequence for Teaching Neural Networks in K-12. In *Proceedings of the 31st ACM Conference on Innovation and Technology in Computer Science Education V. 2 (ITiCSE 2026)*, July 10–15, 2026, Madrid, Spain. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3803401.3811980>

1 Introduction and Motivation

In K-12 CS education, AI is often presented as an opaque “black box” where results emerge from hidden mathematics [8]. Students who

meet professional ML code without first building a mental model of what it represents frequently disengage or develop persistent anthropomorphic misconceptions, such as conflating iterative weight optimization with human-like reasoning. Our project, *The Neural Bridge*, counters this through a structured “Ladder of Abstraction” that fosters AI literacy across all five domains identified by Long and Magerko [8]: recognizing AI, understanding intelligence, inter-disciplinarity, using AI ethically, and applying AI.

The unit was designed for students in K-12 (grades 10–12) with basic Python knowledge (sequences, conditions, loops, functions) and delivered across four 90-minute lessons. The central contribution is the synthesis of two existing artifacts (“7 Little Neurons” unplugged [5, 7] and scaffolded Google Colab notebooks [6]) into a three-phase sequence structured by a deliberate abstraction ladder: each phase builds on the mental model constructed in the previous one, rather than treating unplugged and plugged activities as independent events. This structural logic, grounded in Productive Failure [2] and PRIMM [9], is described in detail in Section 2. All materials are available under CC BY 4.0 here: <https://doi.org/10.17605/OSF.IO/B7234>.

2 The Three-Phase Instructional Sequence

The three phases are not sequential modules but a coherent abstraction ladder. Phase 1 establishes an embodied mental model of weight-based computation through physical role play. This model serves as the anchor to which Phase 2 attaches formal mathematical representation (Google Sheets formulas), making the transition from body to symbol meaningful rather than arbitrary. Phase 3 then reuses both the mental model and the mathematical understanding to make professional Google Colab code readable before it is modifiable. This deliberate sequencing ensures continuity across phases; without explicit bridging, unplugged and plugged activities risk being perceived as independent events [3].

2.1 Phase 1: Embodied Simulation and Productive Failure (Periods 1–2)

Students physically enact a minimal feedforward network. Each student represents a neuron, receiving inputs from neighbors, applying a threshold rule printed on a role card, and passing output forward.



This work is licensed under a Creative Commons Attribution 4.0 International License. *ITiCSE 2026, Madrid, Spain*

© 2026 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-2633-0/2026/07

<https://doi.org/10.1145/3803401.3811980>

The class begins by assigning *random* weights via dice rolls—a deliberate choice. The network inevitably misclassifies, producing what we call the “Houston, we have a problem!” moment. Students must then invent their own update rules to correct the behavior before receiving canonical instruction on gradient descent.

This is the core mechanism of Productive Failure [2]: students construct and test prior knowledge, making subsequent explanations significantly more meaningful. The role cards specify both the activation threshold and the update direction, but deliberately omit the magnitude, forcing students to discover that learning rate is a design choice, not a given. The activity maps directly to Long and Magerko’s competencies of *Recognizing AI* and *Algorithmic Awareness*, students come to understand that AI behavior results from iterative optimization, not human-like reasoning.

Following the simulation, students transfer their update rules to Google Sheets formulas. This bridges embodied experience to formal mathematical representation and directly addresses the *Understanding Representations* competency. A second “Green Cards” round exposes students to unseen test data, surfacing the concept of training-data dependence and introducing the idea of bias.

2.2 Phase 2: Structured Code Reading with PRIMM (Periods 3–4)

Since research indicates that unplugged activities are most effective when explicitly connected to subsequent programming [3], Phase 2 moves into Google Colab using the PRIMM framework [9]:

Predict & Run: Students predict training outcomes for MNIST handwriting recognition before executing cells, then compare predictions to results.

Investigate: Students use a specialized LLM-based scaffold (GEM Tutor) to map Colab code lines back to a Machine Learning Workflow Flowchart developed in the plenary.

Modify: Students adapt Cross-Validation to address unstable single-split evaluation—directly applying the bias-awareness developed in Phase 1.

Make: Students implement and export their own trained model, creating a separate program that loads without retraining.

The unit culminates in a return to the opening discussion: facial recognition at their own school. Students weigh competing values—convenience, security, privacy, autonomy—developing Long and Magerko’s *Ethical Use of AI* and *Societal Impact* competencies in a personally relevant context.

3 Evaluation Results

The instructional unit was delivered in person over three weeks in December 2025. A pre-test was administered. Due to missing individual identifiers, a conservative between-subjects analysis was employed, likely underestimating true learning effects [1].

Conceptual understanding. Correct identification of AI as a system that “only processes data” increased from 63% (17/27) to 82% (14/17), indicating movement away from anthropomorphic misconceptions toward recognition of AI as mathematical computation.

Metacognitive growth. Students’ self-assessed AI understanding *decreased* ($M=2.70$, $SD=0.91 \rightarrow M=2.00$, $SD=0.50$ on a 5-point

scale). The narrowing standard deviation suggests shared, homogeneous experience. This counterintuitive result reflects the Dunning-Kruger effect [4]: students gained enough knowledge to recognize the limits of their understanding, a metacognitive skill essential for responsible AI engagement and a direct indicator of the *Critical AI Thinking* competency.

Practical confidence. Despite decreased self-assessed understanding, students reported increased confidence working with AI tools ($M=2.71$, $SD=0.77$), indicating development of the *Applying AI* competency.

Qualitative findings. The “Seven Little Neurons” unplugged activity was the most frequently cited helpful element. Representative responses included: “*I really like the example with red and blue class, where we students pretended to be neurons and step by step built our own neural network*” and “*At the beginning when we went outside [...] I feel like I picked up the stuff way better.*”

4 Discussion

The Neural Bridge confirms that embodied cognition matters profoundly for demystifying AI. When students physically embody neurons, they build mental models resistant to anthropomorphization—a persistent barrier to AI literacy that Long and Magerko identify as requiring explicit pedagogical intervention [8]. The explicit unplugged-to-plugged transition, informed by Kerr and Kopcha [3], was essential: students’ Colab engagement was notably stronger because they had already internalized what `model.fit()` represents kinesthetically. Limitations include the small post-test sample ($n=17$, 63% retention), the absence of individual identifiers preventing within-subjects analysis, and the time investment required (four 90-minute lessons). Future work will incorporate individual tracking, animated backpropagation visualizations requested by students, and streamlined Google Sheets scaffolding to reduce cognitive load on formula syntax without sacrificing the *Algorithmic Awareness* benefit of manual computation.

References

- [1] Andy Field. 2024. *Discovering statistics using IBM SPSS statistics*. Sage publications limited.
- [2] Manu Kapur. 2008. Productive Failure. *Cognition and Instruction* 26, 3 (2008), 379–424. doi:10.1080/07370000802212669
- [3] Jessica Kerr and Theodore Kopcha. 2025. Unplugged to plugged: An introduction to coding for elementary school children. *Journal of Technology-Integrated Lessons and Teaching* 4, 1 (2025), 9–19. doi:10.13001/jtlt.v4i1.9293
- [4] Justin Kruger and David Dunning. 1999. Unskilled and unaware of it: How difficulties in recognizing one’s own incompetence lead to inflated self-assessments. *Journal of Personality and Social Psychology* 77, 6 (1999), 1121–1134. doi:10.1037/0022-3514.77.6.1121
- [5] Lukas Lehner. 2023. *Machine learning unplugged - Training decision trees and artificial neural networks with children*. Thesis. Technische Universität Wien. doi:10.34726/hss.2023.107391
- [6] Lukas Lehner. 2026. Teaching Material: Seven Little Neurons - Plugged Extensions. doi:10.17605/OSF.IO/EHVPU
- [7] Lukas Lehner, Selina Reinhard, Martina Landman, and René Röpke. 2026. Turning Students into Neurons — Evaluating an Embodied CS Unplugged Learning Activity on Neural Networks. In *Proceedings of the 20th WiPSCE Conference on Primary and Secondary Computing Education Research*. doi:10.1145/3801749.3801755
- [8] Duri Long and Brian Magerko. 2020. What is AI Literacy? Competencies and Design Considerations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, Honolulu HI USA, 1–16. doi:10.1145/3313831.3376727
- [9] Sue Sentance, Jane Waite, and Maria Kallia. 2019. Teaching computer programming with PRIMM: a sociocultural perspective. *Computer Science Education* 29, 2-3 (July 2019), 136–176. doi:10.1080/08993408.2019.1608781