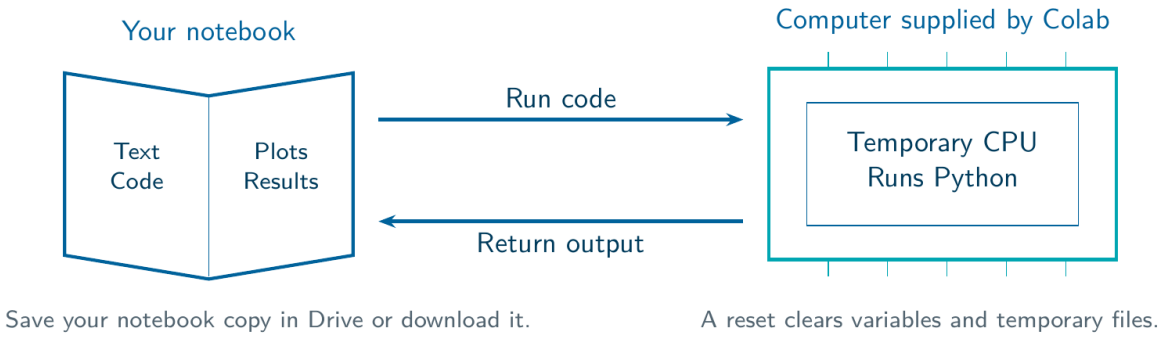
Start with one notebook



Presenters: Alaa Selim, MIT/Dartmouth; Mohamed Chehada, UT Austin.

Open **Classroom_Labs.ipynb**. It covers voltage, restoration, cyber and the saved Llama replays. The website's other notebooks are companion resources.

1. Keep your copy. Sign in and save a copy in Drive, or download your notebook when you finish.

2. Connect a CPU. Run the first setup cell. Wait for **Ready**, then run **L0** and **L1**. Keep both switches False.

3. Follow the slides. Run each named cell when prompted. You will change one number in each case. No coding experience or live LLM service is required.

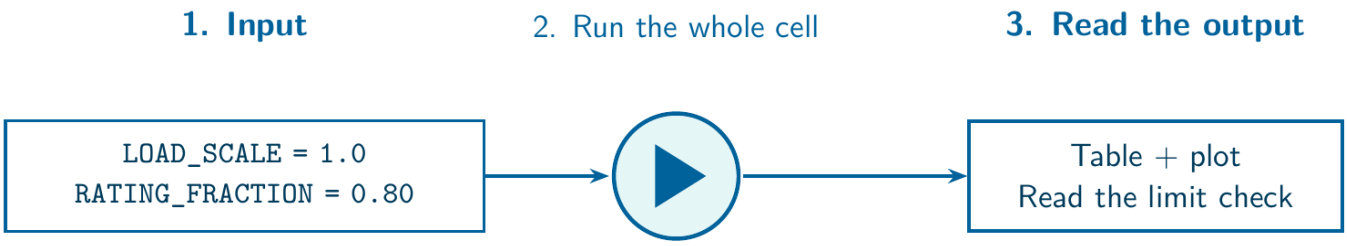
The runtime is a temporary computer. Saving the notebook does not keep its temporary files or variables. After a reset, rerun setup and earlier cells.

Before the first case, show your partner where the code, play button and output appear.

Open Classroom Labs in Google Colab
Scan the code or use the website's first Colab button.

Learn with one small experiment

Predict / run / read / change / explain



The play button executes Python. The saved text below a cell is its output.

Read the input. Find the L heading in the notebook. The slide shows selected lines; run the full cell.

Predict first. Tell your partner what you expect before revealing the result.

Run and wait. Use the play button beside that cell. Read the new table or plot underneath it.

Change one input. Rerun the cell, compare outputs and explain which check supports your next step.

Use the labels. Fresh means calculated now. Recorded replay means an actual old model reply with its original context. A saved controller can produce a new proposal from old weights.

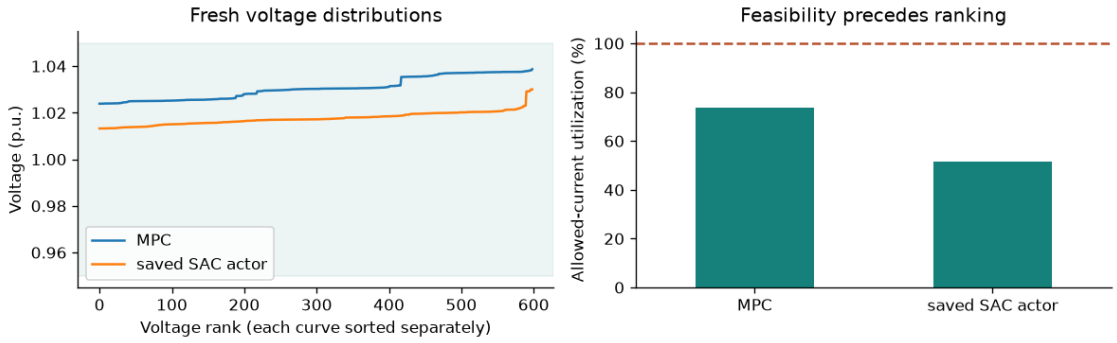
Common messages: HOLD means a required condition is missing or unmet. REVIEW means a person still needs to assess it. L11 SKIPPED is expected when the live-model switch is False.

If a variable is missing: rerun the required earlier cells. Editing a number does not update an old output until you rerun.

Work in pairs: one person runs, one checks. Swap roles for the next case.

Case 1: voltage support

L2–L5 / exercise on slide 16



Problem: which solar settings keep voltage and line current within their limits?

How an agent helps: request studies, read their checks and return acceptable options for operator review.

Run L2 at the defaults: LOAD_SCALE = 1.0 and RATING_FRACTION = 0.80. Read feasible, current_step_J and max_allowed_current_pct.

Read the plots: the left plot shows sorted modeled voltages. The right plot compares current use with the 100% allowed-current line.

Four-minute task: change only RATING_FRACTION to **0.75**. Predict, rerun all of L2 and compare both methods. Check limits before comparing the score J.

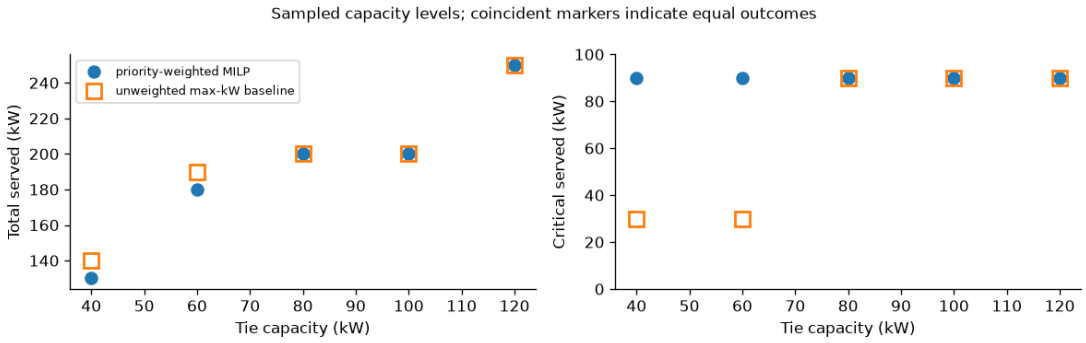
Reset: restore 0.80 and rerun L2. Then follow L3–L5. Those cells use the separate original test_01 archive; your edited L2 does not rewrite it.

Model coverage: the default check covers the MV model. It does not prove that every LV device or dynamic limit passes.

Record: the changed allowance, each current percentage, pass/fail, and what the model does not cover.

Case 2: restoring service

L6–L7 / exercise on slide 28



Problem: a fault removes a supply path. Which loads should receive the limited available power?

How an agent helps: keep the operator's priority in each study request and ask for new checks when limits change.

Run L6: compare served_kw and critical_kw for the priority-weighted plan and the maximum-kW plan. Both use the same six-node graph.

Read the plots together: total service is on the left; critical service is on the right. Explain the tradeoff at 40 kW.

Four-minute task: append **180.** to CAPACITIES_KW. Keep **40.** first. Predict, rerun L6 and compare the 40 kW and 180 kW outcomes.

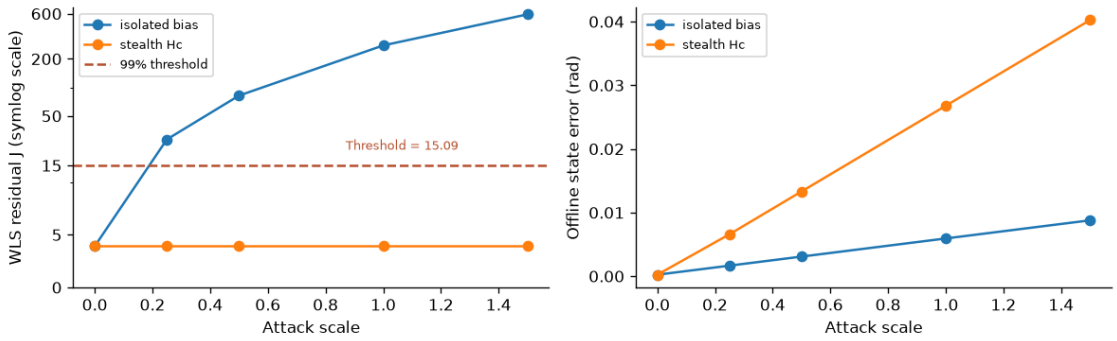
Reset before L7: restore the original capacity list and rerun L6. L7 uses the first record, then tests a changed request with the old evidence.

Meaning of HOLD: old evidence no longer matches. That does not prove a newly calculated plan would fail. The model also omits AC voltage and physical switching order.

Record: total and critical kW at both capacities. Explain which service goal each plan follows.

Case 3: checking suspicious readings

L8–L9 / exercise on slide 41



Problem: wrong readings can still fit a grid model. A quiet alarm alone does not establish data trust.

How an agent helps: request the checks, keep missing evidence visible and return HOLD when a required condition is absent.

Run L8: compare isolated bias with coordinated Hc shifts. The residual alarm measures mismatch. State error uses known simulation truth for teaching.

Four-minute task: add **.10** to ATTACK_SCALES. Predict, rerun L8 and compare the alarm with the state error at that scale.

Reset before L9: restore the original scale list and rerun L8. L9 is a separate fixed recovery case; its correction does not follow your changed sweep.

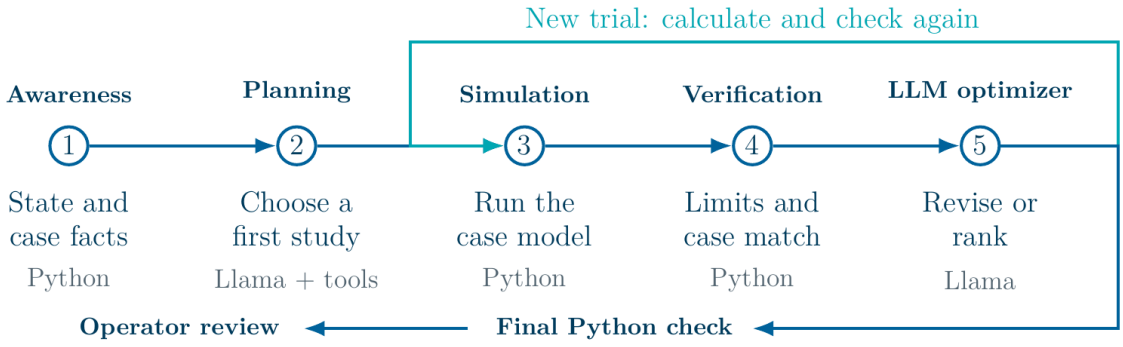
Run L9: no trusted replacement data gives HOLD. Assuming the additional readings are trusted allows a PROPOSE result. The calculation does not authenticate those readings.

Discuss: a small residual or clear model explanation alone cannot establish sensor trust.

Record: residual J, alarm flag, state error and the condition needed before a recovery proposal.

The framework and operator review

L10 / L10B / L11 / L12



Five roles: awareness reads the case, planning chooses a study, simulation calculates, verification checks and the optimizer refines. A changed trial is calculated and checked again.

What runs: one Llama model fills the planning and optimizer roles. Python runs the other stages. L10 displays their saved trace, not a new multi-agent run.

Run L10, then L10B: set SCENARIO = "voltage", then "restoration" and "cyber", rerunning each time. These use separate recorded studies.

Follow the agentic loop: goal, Llama tool choice, returned evidence, next choice. The replay makes no new model call.

Restoration challenge: find the wrong Llama ranking. Explain why the numerical guard keeps the lower-cost passed candidate.

L11: keep the live call disabled. **L12:** inspect the full-scope HOLD, the wrong-ID rejection and the simulated readback. equipment_dispatch stays False.

Exit ticket: explain one result using input, tool, result, check and operator decision. Save the changed inputs and outputs.

Choose one case. Which decision did Llama make, which fact came from a tool, and what must the operator still review?

Monica Lam. *CS224V Agentic AI, Lecture 1: Introduction.* Stanford University, Fall 2026, pp.2, 19, 40 and 52–53. Full lecture PDF. Original source figures and clearly labeled adaptations connect the lecture to our grid examples. The lab uses custom Python.
Google Colaboratory. *Frequently Asked Questions.* research.google.com/colaboratory/faq.html.
Full paper citations, figure credits and exact notebook source map: Teaching and Source Guide.