Severn Edge AI — Classroom Guide

Duration: 3 hours **Audience:** 5th graders (10-11 years old) **Goal:** Students train and deploy their own gesture recognition AI

Materials Checklist

- □ Arduino Nano 33 BLE Sense (one per student)
- □ USB cables
- □ Laptops with Chrome browser
- □ Web app running (`npm run dev` or deployed URL)
- □ Firmware flashed to all boards

Hour 1: "What Does AI Actually See?" (60 min)

Hook (15 min)

Ask the class: **"Is AI magic?"**

Hold up the Arduino. "This tiny board is going to learn from YOU today."

Shake the board, open the **"See what the AI sees"** panel on screen. The 6 bars jump around as the board moves. Key insight: **AI sees numbers, not motion.** It finds patterns in those numbers to figure out what you're doing.

Live Demo (15 min)

Walk through the full flow end-to-end so students see the finish line before they start:

1. Connect to Arduino via Bluetooth
2. Choose gestures on the setup screen
3. Record a few samples (wave, shake)
4. Train the model (watch the accuracy climb)
5. Deploy to Arduino
6. Test it live — wave and see the prediction appear

Keep it fast. The goal is excitement, not explanation.

Connect (15 min)

- Each student plugs in their Arduino and opens the web app in Chrome
- Click Connect, find their board (e.g. "SevernEdgeAI-8")
- Celebrate each successful connection

Troubleshooting: If a board doesn't appear, check USB is plugged in and try refreshing the page.

Explore the Sensor (5 min)

Once connected, have students open the **"See what the AI sees"** panel (click the collapsible header on the Collect page). This shows a live terminal-style display with 6 values and color-coded bars:

- **Red bars** — accelerometer (ax, ay, az) — measures all forces on the board. When it's still, that force is gravity — which tells us which way is down. Shake it, and the readings change even if the tilt doesn't.
- **Blue bars** — gyroscope (gx, gy, gz) — measures how fast the board is rotating

Have students move the Arduino and watch the numbers react:

- "Wave it — what numbers change?"
- "Hold it perfectly still — what do you see?"
- "Flip it upside down — which number moved?"
- "Rotate it forward like a seesaw — which blue bar spikes?" (gyroscope!)

We sample these values about 25 times per second — the same rate the AI model was trained on. This builds intuition that the 6 numbers are all the AI has to work with. There's no camera, no microphone — just motion numbers.

Real devices combine accelerometer and gyroscope together so your phone, drone, or VR headset always knows its orientation.

Sensor Challenges (10 min)

Now that students can see the numbers, give them challenges that build intuition about *which* numbers matter:

Challenge 1 — "Same or Different?"

> *Do a slow wave, then a fast wave. Watch the panel. Can you see the difference?*

Kids notice: fast wave produces bigger gyro spikes. The AI *can* tell these apart, but the difference is subtle. This sets up the "make gestures distinct" rule naturally.

Challenge 2 — "Find the Axis"

> *Try to make ONLY the gz bar move. Nothing else.*

They'll discover spinning the board like a top isolates gz. This teaches that each axis responds to a specific type of motion — the AI has 6 independent signals, not just one blob of "movement."

Challenge 3 — "Trick Question"

> *Do a big circle. Now stir a pot. Watch the numbers. Are they the same to the sensor?*

Usually yes — two gestures that *look* different to a human can look *identical* to the sensor. This is the deepest lesson: **the AI doesn't see what you see. It sees what the sensor reports.**

Challenge 4 — "Design Your Gestures"

> *Before you pick your 3 gestures, test them on the sensor panel. Do they light up different bars? If two gestures make the same bars move the same way, the AI will struggle to tell them apart.*

This is the payoff — students now use the panel as a *design tool* to pick gestures that are sensor-distinct, not just visually distinct. Kids who do this will get higher accuracy and perform better in The Swap Challenge.

Hour 2: "Teach Your AI" (60 min)

What Makes Good Training Data? (5 min)

Show two examples side by side:

1. **Good data:** Someone waves consistently 10 times, same speed, same motion
2. **Bad data:** Someone waves differently every time, sometimes barely moves

Train both. Show the accuracy difference.

Core lesson: **Garbage in, garbage out.** The AI can only learn patterns if the patterns are consistent.

Choose Gestures (10 min)

Each student picks 3 gestures on the setup screen. Encourage creativity:

- "Cast a spell"
- "Stir a pot"
- "Swing a bat"
- "Draw a star"
- "Throw a ball"
- "Brush your teeth"

Rule: Gestures must be physically distinct. If two gestures feel similar (like "small wave" and "big wave"), the AI will struggle. This is a great teaching moment when it happens. Students who did the sensor challenges will already know to check the panel before committing.

Students can rename the defaults, remove them, or add new ones (up to 8).

Collect Training Data (30 min)

Students record 10+ samples per gesture:

1. Select a gesture card
2. Click "Record Sample"
3. Perform the gesture for 4 seconds — the **"See what the AI sees"** panel shows live numbers as they record
4. Repeat until all gestures show "DONE"

Tip: Encourage students to keep the sensor panel open while recording. Watching the numbers helps them understand why consistent motion matters — if the bars look different every time, the AI will struggle.

Walk around the room helping. Common issues:

- **"It keeps rejecting my sample"** — Move bigger! The quality check needs to see real motion.
- **"My accuracy is bad"** — Are your gestures different enough? Collect more samples.

Deploy + First Test (15 min)

1. Click "Next: Train Model"
2. Watch the training progress (accuracy should climb above 80%)
3. Click "Deploy to Arduino"
4. Switch to Test mode
5. Perform gestures and watch predictions appear

Celebrate working models! Help debug failures.

Hour 3: "The Gesture Games" (60 min)

Contest: The Swap Challenge (40 min)

This is the main event. It tests whether students' models actually learned real motion patterns.

Setup (5 min):

- Each student writes down their 3 gesture names on a piece of paper (face down)
- Students pair up with a neighbor

Round 1 — Guess the Gestures (15 min):

- Partners swap Arduinos
- You must figure out the other student's 3 gestures by trying different motions and reading the predictions
- You get the prediction label (Gesture 0, 1, 2) but NOT the name
- First student to correctly name all 3 gestures wins the round

Round 2 — The Stranger Test (15 min):

- Each student demonstrates their gestures live
- But their partner performs them on the Arduino
- Score: how many does the model get right out of 10 attempts?
- Highest score wins

Why this works:

- Proves the model learned real patterns, not just memorized one person's style
- Physical, loud, competitive — kids love it
- Teaches generalization: did your model learn "the gesture" or just "your hand"?

Scoring:

Round	Points
Swap Challenge: Guess all 3 gestures	3 points per correct guess
Stranger Test: Correct predictions	1 point per correct prediction

Reflection (15 min)

Bring the class back together. Ask:

- **"What made your model better?"** (More data, distinct gestures, consistent recording)
- **"What made it worse?"** (Similar gestures, sloppy recording, not enough samples)
- **"What surprised you?"** (How fast it trained, how it sometimes gets confused, how it works without seeing you)
- **"Could the AI learn any gesture?"** (What are the limits?)

Draw out the core lessons:

1. **Data quality matters** — consistent, clean data makes better models
2. **Distinct patterns help** — the AI needs to tell things apart
3. **More data helps** — but only if it's good data
4. **AI isn't magic** — it's pattern matching on numbers

Wrap-Up (5 min)

"You just trained and deployed a real neural network. That's what AI engineers do — they collect data, train models, and put them on devices. The only difference is scale. You used 30 samples and 6 sensors. A self-driving car uses millions of samples and hundreds of sensors. But the idea is exactly the same."

Alternative/Bonus Contest Ideas

The Fourth Gesture Challenge

After the main contest, students add a 4th gesture and retrain:

- Can you find a gesture that doesn't hurt accuracy?
- Students compete for highest 4-class accuracy
- Teaches: more classes = harder classification, class boundaries matter

Fool the AI

Students try to find motions that trick another student's model into wrong predictions:

- Most successful "fooling" wins
- Teaches adversarial thinking: what confuses ML?
- Example: a gesture that's halfway between two trained gestures

---

Teacher Notes

Before Class

- Flash firmware to all Arduinos and verify each connects via BLE
- Note each board's number (e.g. SevernEdgeAI-8) and label the physical boards to match
- Test the web app on the classroom laptops (Chrome only — BLE doesn't work in Firefox/Safari)
- Have a backup laptop in case of Bluetooth issues

Common Failure Modes

Problem	Fix
Board not appearing in Bluetooth dialog	Unplug and replug USB, refresh page
Training accuracy stays low	Collect more samples, make gestures more distinct
Model upload fails	Check Bluetooth connection, try again
Different predictions every time	Train longer, collect more data, check gesture consistency

Key Vocabulary for Students

- **Training data** — Examples you show the AI so it can learn
- **Model** — The AI's "brain" after training
- **Inference** — When the AI makes a prediction on new data
- **Accuracy** — How often the AI gets the right answer
- **Sensor** — The chip that measures motion (accelerometer + gyroscope). Students can see live sensor values in the "See what the AI sees" panel
- **Deploy** — Sending the trained model to the Arduino