ARTIFICIAL INTELLIGENCE AND NEURAL NETWORKS

Lecture 2b – Perceptrons, Logistic Units, and Linear Boundaries

Chizhi Chris ZHANG
zhangchizhi@ciomp.ac.cn
Advanced Computing and Digital Technology Research Center
University of Chinese Academy of Sciences

Spring 2026

Today's Question

Main question

When is one straight boundary enough, and when is the world asking for a richer internal picture?

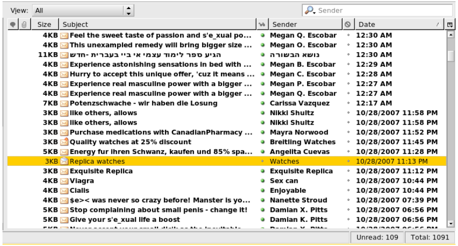
What changes from NN1

NN1 introduced neurons, layers, and internal representation. NN2 deliberately freezes the model at one layer so we can see what that simple mechanism can and cannot express.

Three words to keep watching

score, boundary, limit

One Inbox, Many Clues



A familiar decision

An email system looks at sender reputation, suspicious words, odd links, and unusual patterns in the message.

What a perceptron-like view says

Each clue nudges the message toward “probably spam” or “probably safe” until one threshold turns the score into a decision.

A One-Slide Recap

What to keep from NN1

A neuron forms a weighted sum, passes it through a rule, and contributes to a later representation.

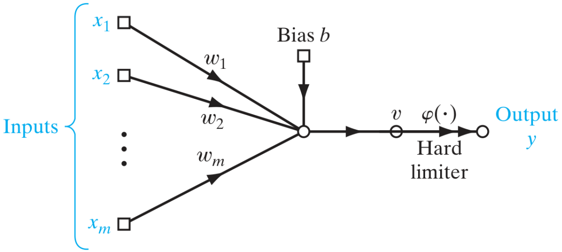
What we simplify today

We remove the depth and look at one layer, one score, and one decision boundary.

Why slow down here

If this baseline is not clear, later words such as logistic units, hidden layers, and backpropagation turn into vocabulary without mechanism.

Perceptron as a Small Mechanism



Core formula

$$z = w^\top x + b, \qquad \hat{y} = \mathbb{1}(z \geq 0)$$

A simple spam-filter picture

An unknown sender, many links, and suspicious wording can all push the score upward until the message crosses the spam threshold.

What the Score Means Before Decision

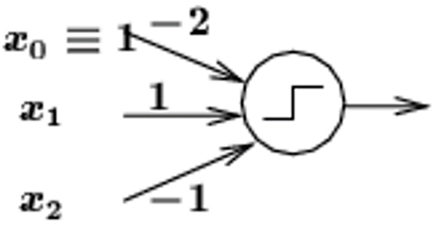
How to read z

- large positive score: the evidence points strongly toward the positive side
- score near zero: the example sits close to the boundary
- large negative score: the evidence points strongly toward the other side

A borderline case

A message from a familiar sender with one suspicious link may sit near the boundary: not clearly safe, not clearly spam.

Threshold Means Choice



What threshold does

It decides when a score becomes a positive prediction.

Key warning

A model score and a final decision are not the same thing. The threshold is already a policy choice.

One Boundary Shows Up in Daily Life

Ordinary examples

Spam filters, early warning systems, admission screening, fraud checks, and simple medical triage all often begin with one score crossing one cutoff.

Why this matters

The perceptron is not interesting only as history. It is a clear picture of how many real systems still turn several clues into one decision line.

What the user sees

People experience the final decision, but the important hidden choice is often where the threshold was placed and who decided it.

Why One Neuron Is Worth Studying

Why this small model is still worth the time

First, it is interpretable: we can see how inputs, weights, bias, and threshold affect the result.

Second, it is a good reference point: if a harder model cannot beat it honestly, the project likely has a deeper problem.

The real reason it stays in the course

The jump to multilayer networks only makes sense after students feel this limit directly.

How Linear Boundaries Look

Boundary equation

$$w_1 x_1 + w_2 x_2 + b = 0$$

Geometric meaning

In two dimensions, the model draws one line.
In higher dimensions, it draws one hyperplane.

Why this matters

Once students picture the boundary, they can immediately see which tasks fit the model and which ones do not.

How Weights and Bias Move the Boundary

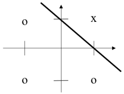
What the parameters control

Changing the weights rotates the boundary because the model changes which clues matter more. Changing the bias shifts the whole boundary without changing its general direction.

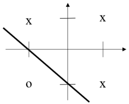
A simple picture

If test score and attendance both matter, larger weight on attendance tilts the rule toward attendance, while the bias makes the cutoff stricter or looser.

Some Tasks Are Easy for One Boundary



AND works



OR works too

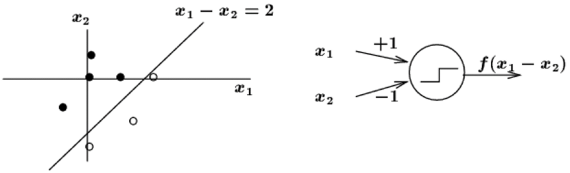
What this page is really saying

Not every task is hard. Some problems really do fit one clear boundary, and it is a mistake to reach for a deeper model automatically.

What to remember

The point is not to dismiss simple models. The point is to know their honest range.

A Worked Example of Classification



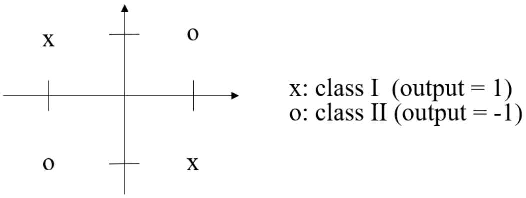
How to read it

Take the input values, apply the weights, add the bias, and compare the score with the threshold.

What the example gives you

The formula stops looking abstract once one specific input actually walks through the mechanism from numbers to decision.

XOR Shows the Real Limit



What fails

No single line can separate the positive and negative examples of XOR.

Why this matters historically

This was the moment the field had to admit that better training alone would not fix everything. The representation itself was too weak.

Why a Loss Is Needed

Why training needs feedback

A model cannot improve unless we tell it what "wrong" means.

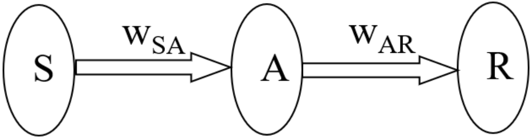
What loss does

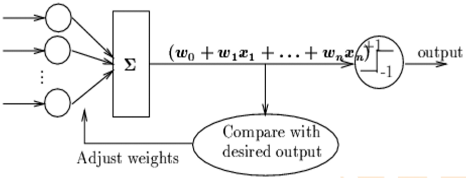
It turns mismatch between prediction and target into a quantity we can optimize.

The part people rush past too fast

The loss function is not just a formula. It is the definition of what kind of mistake the system will care about most.

Perceptron and Logistic Units Differ





Perceptron unit

hard yes-or-no output, good for learning the basic classification mechanism

Why the second version matters

Continuous outputs make optimization and confidence-style interpretation easier when ranking or threshold tuning matters.

Gradient Updates and Learning Rate

Update rule

$$\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}$$

If η is too small

training drifts slowly and may never reach a useful point in time

If η is too large

training jumps around, overshoots, or diverges

A practical habit

Before changing many things at once, get the learning rate into a sensible range.

The Training Loop in Plain Language

The loop is simpler than the notation suggests

First, make a score from the current weights.

Second, measure how far that score is from the desired answer.

Third, move the weights slightly so similar mistakes become less likely next time.

Why this page matters

Students often think learning is a mysterious jump. It is really repeated correction.

Why This Still Matters in Modern AI

What deeper systems change

Deep models can build far richer internal features before the final decision is made.

What often remains at the end

After all that representation building, a final layer still often turns features into scores that are ranked, thresholded, or classified.

A modern picture

A large model may build the representation, but a later scoring head still decides whether a message is spam, a post is risky, or an image matches a label.

Why NN2 still matters

One-layer logic is not the whole modern system, but it still survives inside modern systems.

From Score to Probability

The sigmoid view

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

What it gives you

A raw linear score can be mapped into a value that we read as a probability-like confidence.

A familiar use

Instead of saying only "spam" or "not spam", the system can rank messages by how risky they look and let a later threshold decide what to block.

Why people like this move

This is the bridge from "a line in feature space" to "a score that people can actually use".

Score, Probability, and Decision Are Not the Same

Three different objects

- score: raw weighted evidence
- probability-like output: a confidence-style signal that helps ranking
- final decision: the yes-or-no action after considering costs

Do not collapse these into one thing

Confusing these three layers is one reason students think a model output is already a final judgment.

The Same Score Can Lead to Different Decisions

Why context matters

The same model score can be used differently depending on what kind of mistake is costly.

What changes

Different thresholds trade false alarms against missed detections. There is no single threshold that is universally “correct” outside context.

A screening example

If missing a true positive is very expensive, we may accept more false alarms in exchange for higher recall.

The main lesson

Model output is not a decision by itself.

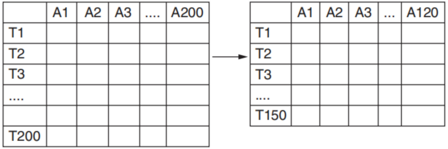
Why Scaling and Geometry Still Matter

Why scale changes learning

If one feature ranges from 0 to 1 and another ranges from 0 to 100,000, the larger numbers can dominate distance and gradient updates before the model ever learns what matters.

A concrete case

Imagine predicting risk from **late library returns** and **annual spending**. Without rescaling, the model may react more to the unit size than to the real signal.



Looking at the geometry helps. A quick plot often reveals whether the problem is the model, the units, or the way we described the task.

When a Linear Model Is Already Enough

Good reasons to start simple

- the classes are roughly separable by one simple boundary
- we need an interpretable first comparison before trying richer models
- speed or transparency matters as much as raw accuracy

Why this still matters

The right lesson is not “simple is outdated”. It is “simple is valuable inside its honest range”.

Where Single-Layer Models Fail

Three signs one layer is too weak

- the real rule depends on combinations, not isolated features
- XOR-like geometry shows that one straight boundary cannot carve the space correctly
- manual feature crosses can help for a while, but they become brittle as tasks grow

Why this becomes the bridge to NN3

The jump to deeper models is not "more power because bigger". It is "more power because representation becomes richer".

Feature Engineering Helps, But Only So Far

What feature design really does

It rewrites the problem so a simple model can notice a pattern that was hidden in the raw fields.

A realistic example

“Missed payments” and “days since last login” may each be weak alone. Their interaction can become much more informative than either column by itself.

Why it stops scaling

As useful interactions become numerous, subtle, and unstable over time, hand-building all the right combinations turns into guesswork.

Why this becomes NN3

Hidden layers learn many of these intermediate combinations automatically instead of waiting for us to invent them one by one.

What a Better Representation Changes

Before and during

At first the raw input space looks tangled and one boundary is not enough. Intermediate transformations reshape the data into a more useful internal form.

After

A later layer can draw a simpler decision boundary on top of that new representation.

Common Misreadings of This Lecture

A common misreading

"Linear models are old, so they do not matter anymore."

Another common misreading

"If a task is hard, a single-layer model just needs more tuning."

A better reading

They remain useful references, interpretation tools, and teaching models for boundary geometry.

A better reading

When the representation is wrong, tuning cannot repair the model family itself.

The Limit Is Structural

One sentence worth keeping

Sometimes the important failure is not optimization failure. It is structural failure: the model family cannot express the right pattern.

Why this line matters later

It explains why later lectures need hidden layers, backpropagation, and richer architectures.

Why NN2 and NN3 Are Paired

What NN2 has done

Single-layer models are clear, fast, and useful whenever one boundary is enough.

What NN3 adds

We now ask how hidden layers and backpropagation let a network build richer internal representations and learn them efficiently.

What Later Neural Lectures Will Add

What this lecture gave you

Weighted sums, thresholds, linear boundaries, loss, updates, and the structural limit of one layer.

What later lectures specialize

Hidden layers for richer representation, backpropagation for efficient learning, and later model families for images, sequences, and modern generative systems.

Why the pairing works

NN2 gives the clear baseline. NN3 explains how later neural models escape that baseline without becoming magic.

Summary

- Perceptrons and logistic units turn weighted evidence into scores, and scores into decisions through boundaries and thresholds.
- XOR makes the limit visible: when one boundary is the wrong geometry, harder tuning will not rescue the model family.
- When you meet a neural model, ask what boundary it can draw, what score it produces, and whether the structure is strong enough for the task.

Next: Hidden Layers and Backpropagation

Where we go next

The next neural lecture asks how a network can build better internal features instead of relying on one final boundary drawn on raw inputs.

Keep this question in mind

If one neuron is a weighted vote, what kind of internal transformation would make a hard pattern easier to separate?

Why this matters. This is the exact step from single-layer clarity to multilayer learning.

Thank You