Machine Learning
Tasks, Taxonomy & Beyond

From Classification to Deep Learning

Nipun Batra | IIT Gandhinagar

ML is Everywhere in Your Daily Life

Time	What You Do	ML Behind It
Morning	Phone unlocks with your face	Face Recognition
Commute	Google Maps predicts traffic	Time Series Prediction
Email	Gmail filters spam, suggests replies	Text Classification + Generation
Music	Spotify recommends songs	Recommendation Systems
Shopping	Amazon shows "You might also like..."	Collaborative Filtering
Photos	Google Photos groups by faces	Clustering + Image Classification
Evening	Netflix suggests what to watch	Recommendation Systems
Chat	You ask ChatGPT a question	Language Models (Generative AI)

Each of these is a different ML task!

ML Notation: A Concrete Example

House Price Dataset (what we give to the model):

i	sqft (x_1)	beds (x_2)	age (x_3)	price (y)
1	1200	2	5	$250K
2	1800	3	10	$350K
3	2400	4	2	$500K
...	...	...	...	...
n	1500	2	15	$280K

- Each row i is one **sample**: $(\mathbf{x}_i, y_i)$
- $\mathbf{x}_i = [x_{i1}, x_{i2}, x_{i3}]$ = features for sample i
- y_i = target/label for sample i
- All samples together: $\mathbf{X}$ (matrix), $\mathbf{y}$ (vector)

ML Notation: The Symbols

Symbol	What it means	In our example
n	Number of samples	1000 houses
$\mathbf{x}_i$	Features of sample i	$[1200, 2, 5]$
y_i	Target of sample i	\$250K
$\mathbf{X}$	All features (matrix)	$n \times 3$ matrix
$\mathbf{y}$	All targets (vector)	$n \times 1$ vector
$f(\mathbf{x}; \theta)$	Model with parameters	Neural network
$\hat{y}_i$	Prediction for sample i	$f(\mathbf{x}_i) = 245K$

Train/Test Split: Use ~80% to train, ~20% to test (never peek at test during training!)

The Big Question

Every ML task boils down to **one question**:

"What are you trying to PREDICT?"

Predicting a Category?

Classification

"Is this email spam?"

Predicting a Number?

Regression

"What will be the price?"

Predicting a Sequence?

Seq2Seq

"How do you say this in French?"

Creating Something New?

Generative

"Draw me a cat in space"

> Once you know the "output type", you know which family the task belongs to!

The Three Learning Paradigms

Supervised

Learn from examples

Data: X + Y
Tasks: Classification
Regression, Detection

Unsupervised

Find patterns

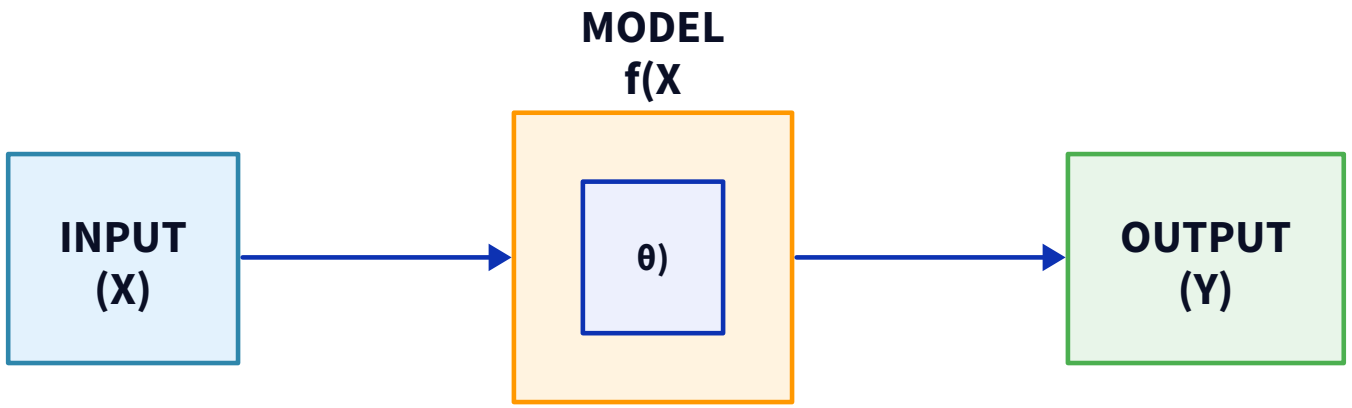
Data: X only
Tasks: Clustering
Dim Reduction

Self-Supervised

Create own labels

Data: X creates Y
Tasks: Next Token
Masked LM

The Universal ML Recipe



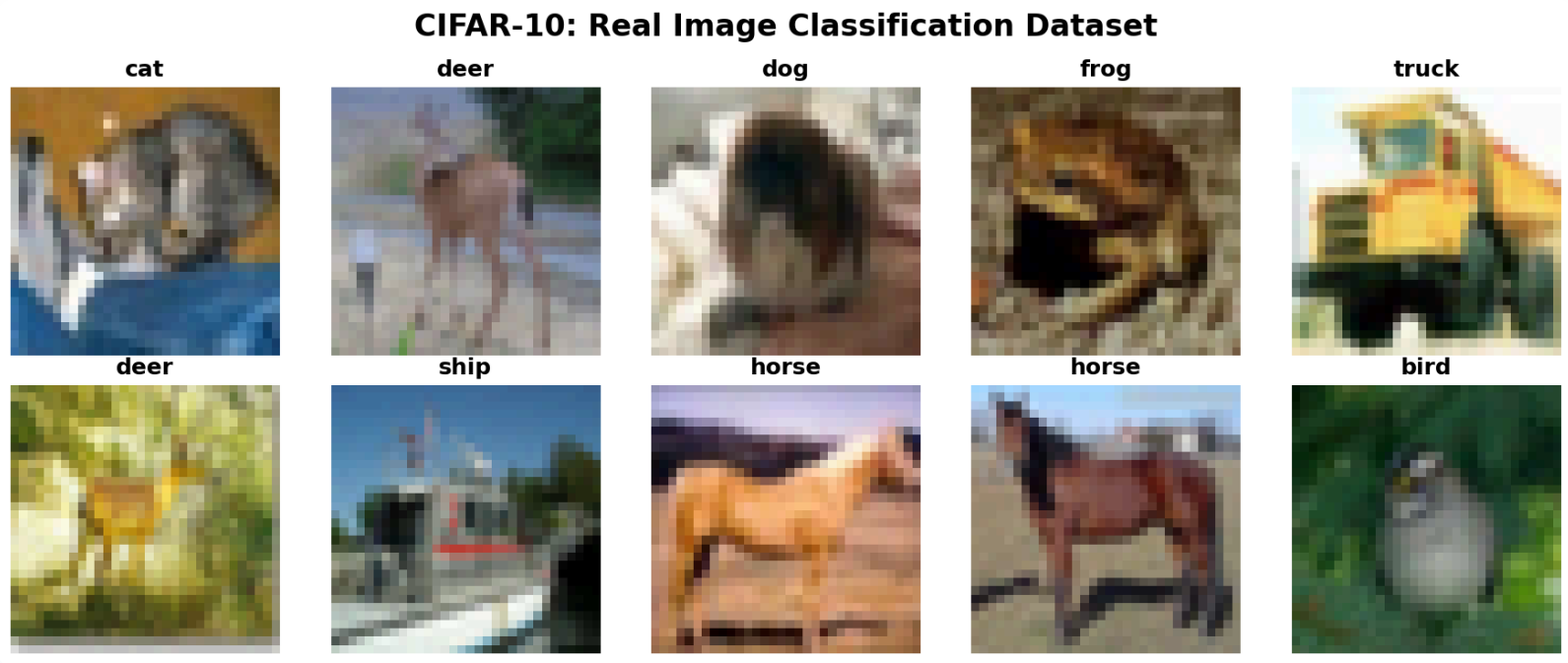
What changes between tasks:

- What X looks like (image, text, audio, numbers)

Part 1: Classification

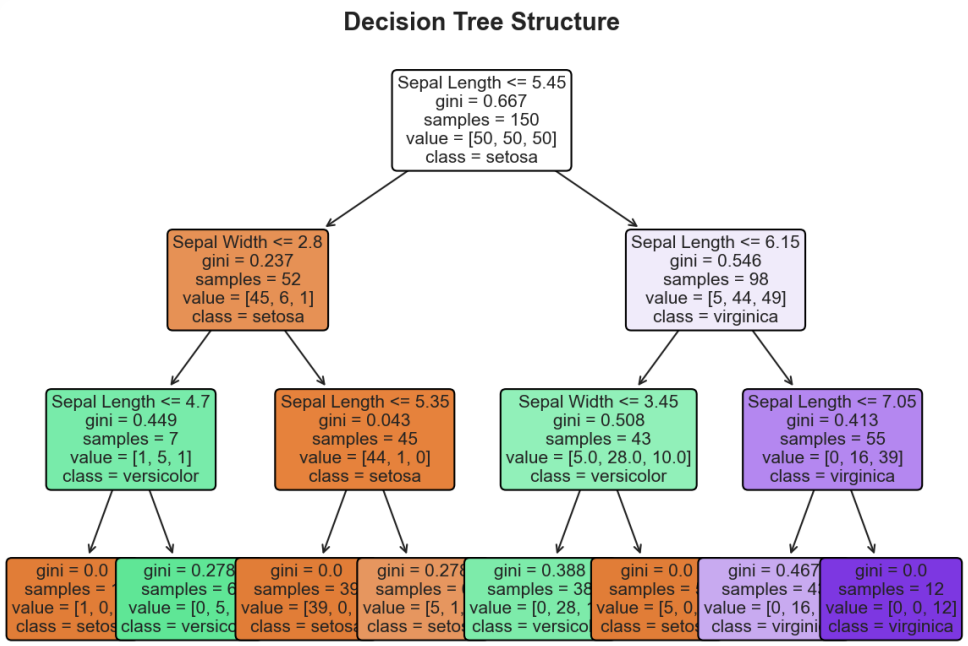
"Which Bucket Does This Belong To?"

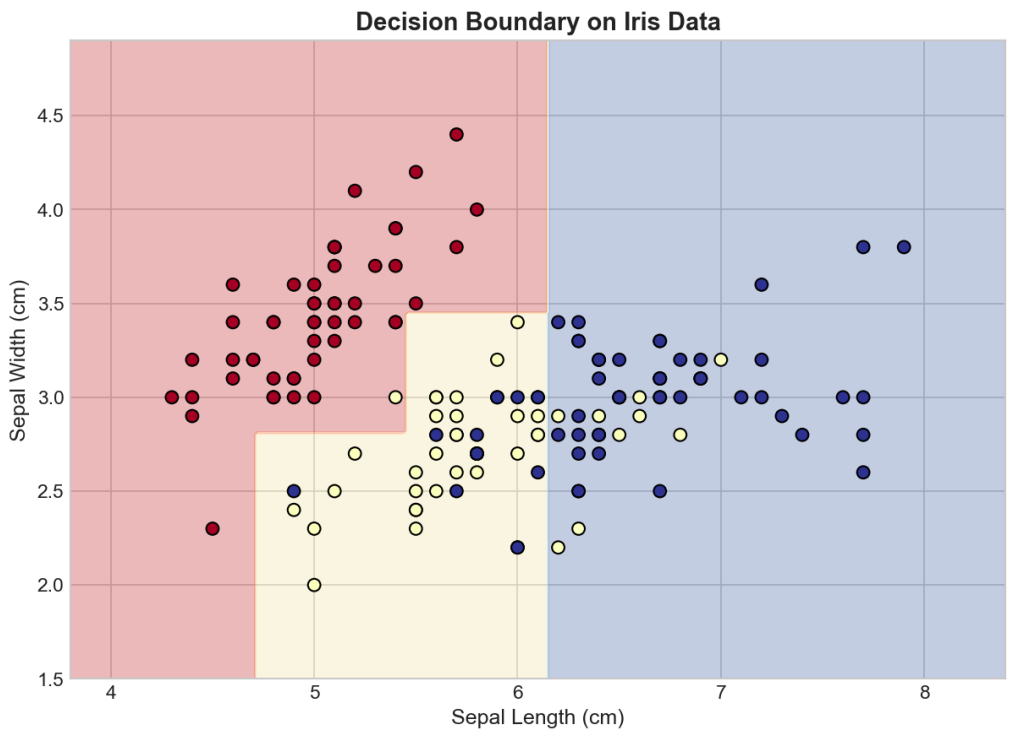
Classification: Real Examples from CIFAR-10



You look at the input and pick **one category** from a fixed set. That's classification!

Classification: How a Decision Tree Learns

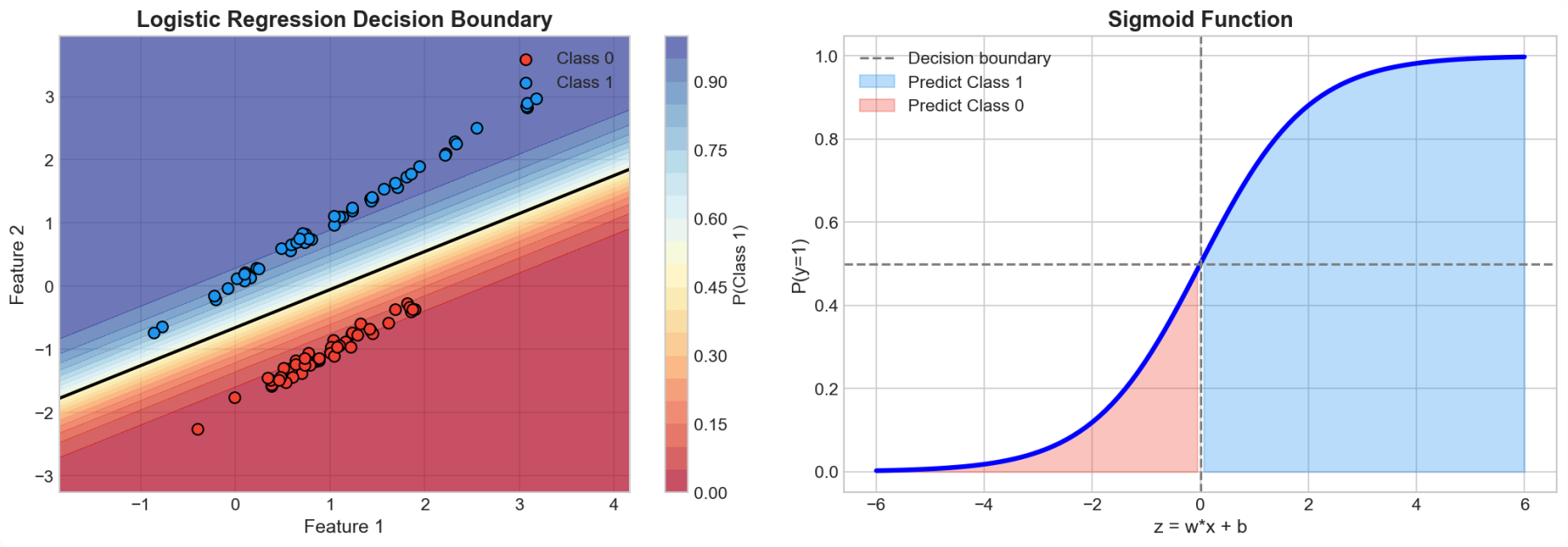




A decision tree learns **if-then rules** from data:

"If sepal length > 5.5 AND sepal width < 3.0, then iris-versicolor"

Classification: Logistic Regression



Logistic regression learns a **decision boundary** that separates classes.

Binary vs Multi-Class Classification

Binary Classification

Two possible outcomes

- Spam / Not Spam
- Fraud / Legitimate
- Pass / Fail
- Tumor: Benign / Malignant

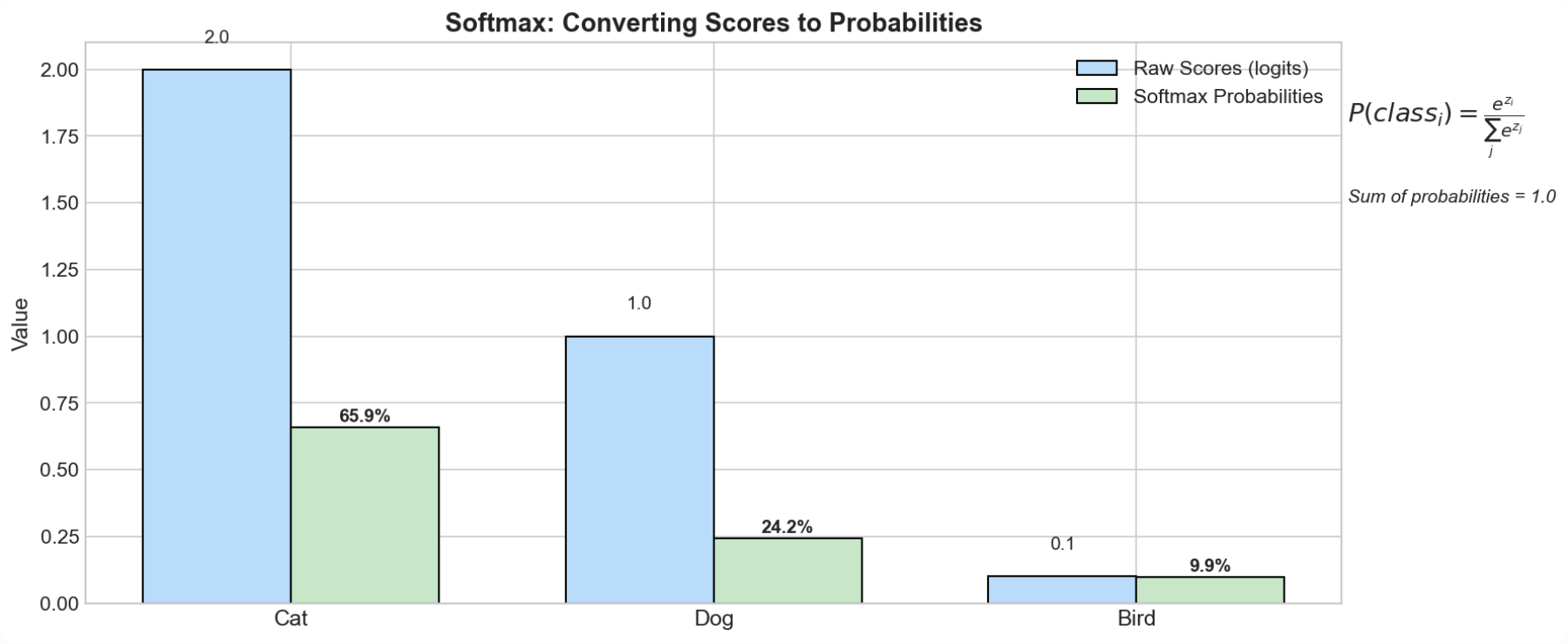
Multi-Class Classification

Many possible outcomes

- Digit recognition (0-9)
- ImageNet (1000 classes)
- Emotion detection (6+ emotions)
- Animal species identification

> Same algorithm, just different number of outputs!

The Math: Softmax Turns Scores into Probabilities



Softmax converts raw scores (logits) to probabilities that sum to 1.

The model isn't just saying "Cat" - it's saying "85% sure it's a cat!"

Part 2: Regression

"How Much? How Many?"

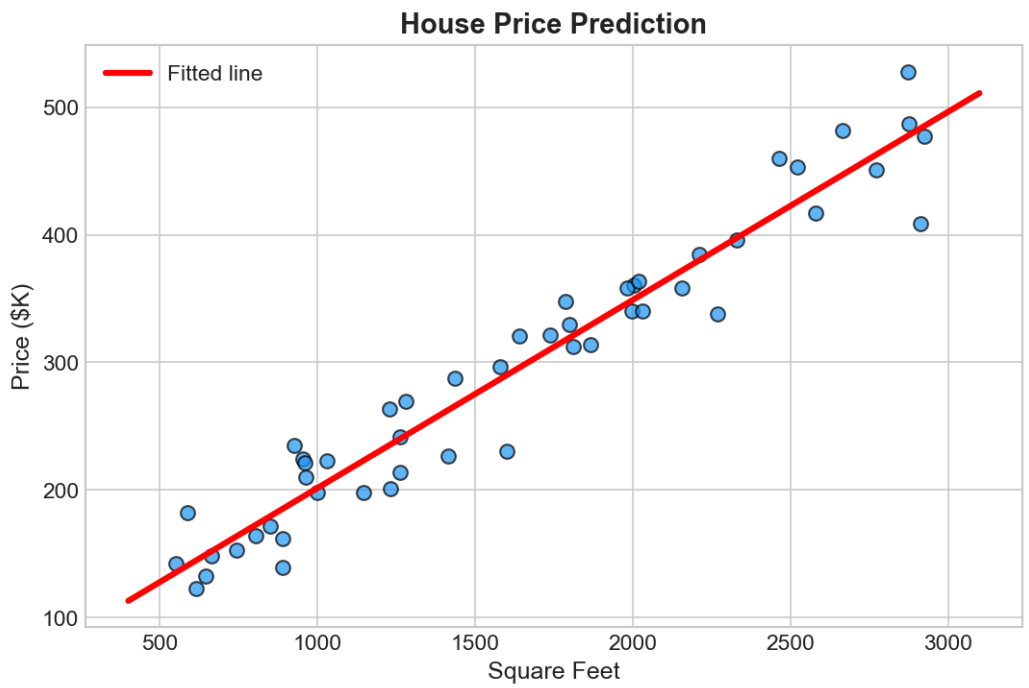
Regression: When the Answer is a Number

Classification: *"Which category?"* - Discrete answer
Regression: *"How much?"* - Continuous number

Question	Answer
"How old is this person?"	27.3 years
"What's this house worth?"	$425,000
"How many units will sell?"	1,247 units
"What temperature tomorrow?"	28.5 C
"How long until the bus arrives?"	7.2 minutes

The output is **any number** on a continuous scale!

Regression in Action: Linear Regression



Linear Regression Formula:

$$\hat{y} = w_0 + w_1 \cdot x$$

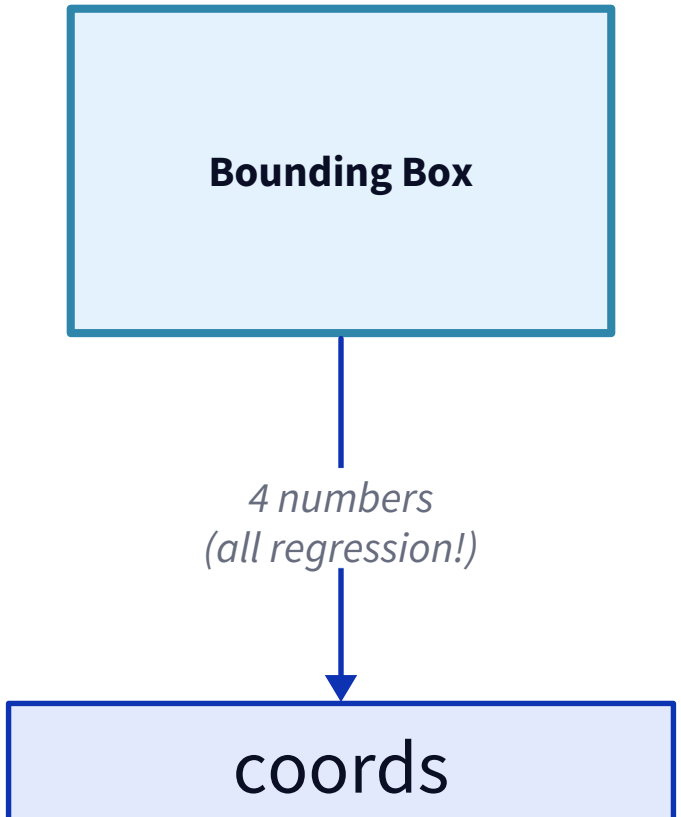
Fitted: price = 54, 241 + 147 * sqft

Each extra sqft adds ~$150 to price!

The model learns: **Price = $50,000 + $150 * (square feet)**

Regression is Hidden Everywhere!

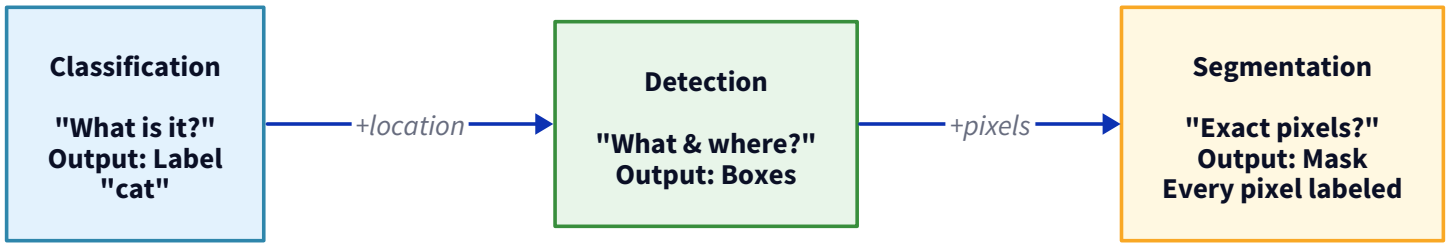
Bounding box detection is actually **regression**:



Part 3: Computer Vision Hierarchy

From Labels to Pixels

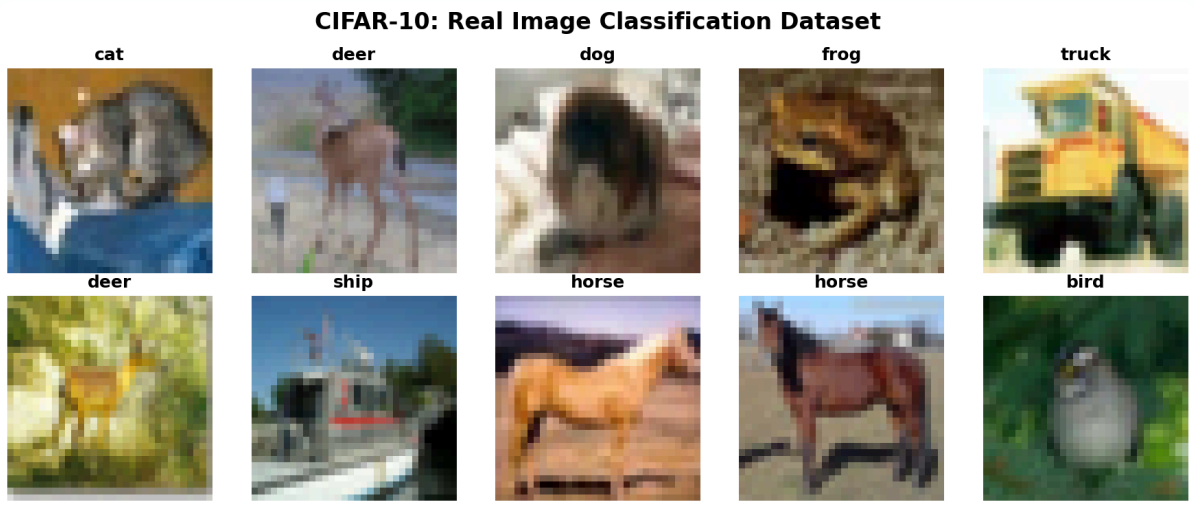
The Vision Task Ladder



Each level gives you more information but requires more data and compute!

Level 1: Image Classification

What: Assign one label to an image.



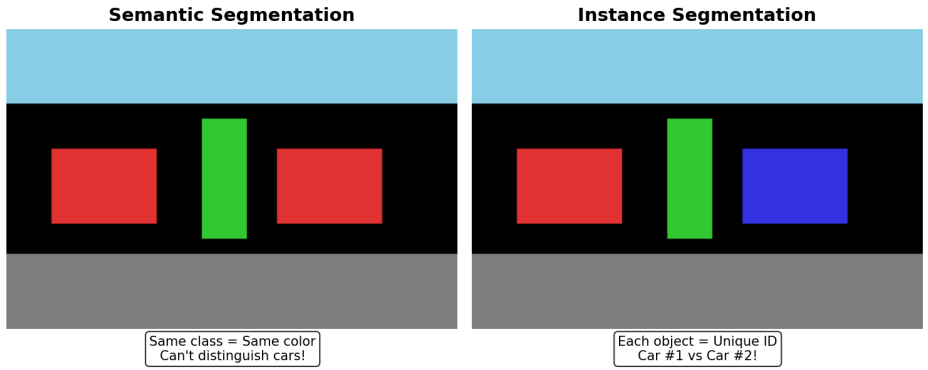
Use Cases:

- Google Photos: "Show me all photos with dogs"
- Medical: "Is this X-ray normal or abnormal?"
- Quality Control: "Is this product defective?"

Level 2: Object Detection

Detection = Classification (what) + Regression (where)

Level 3 & 4: Segmentation



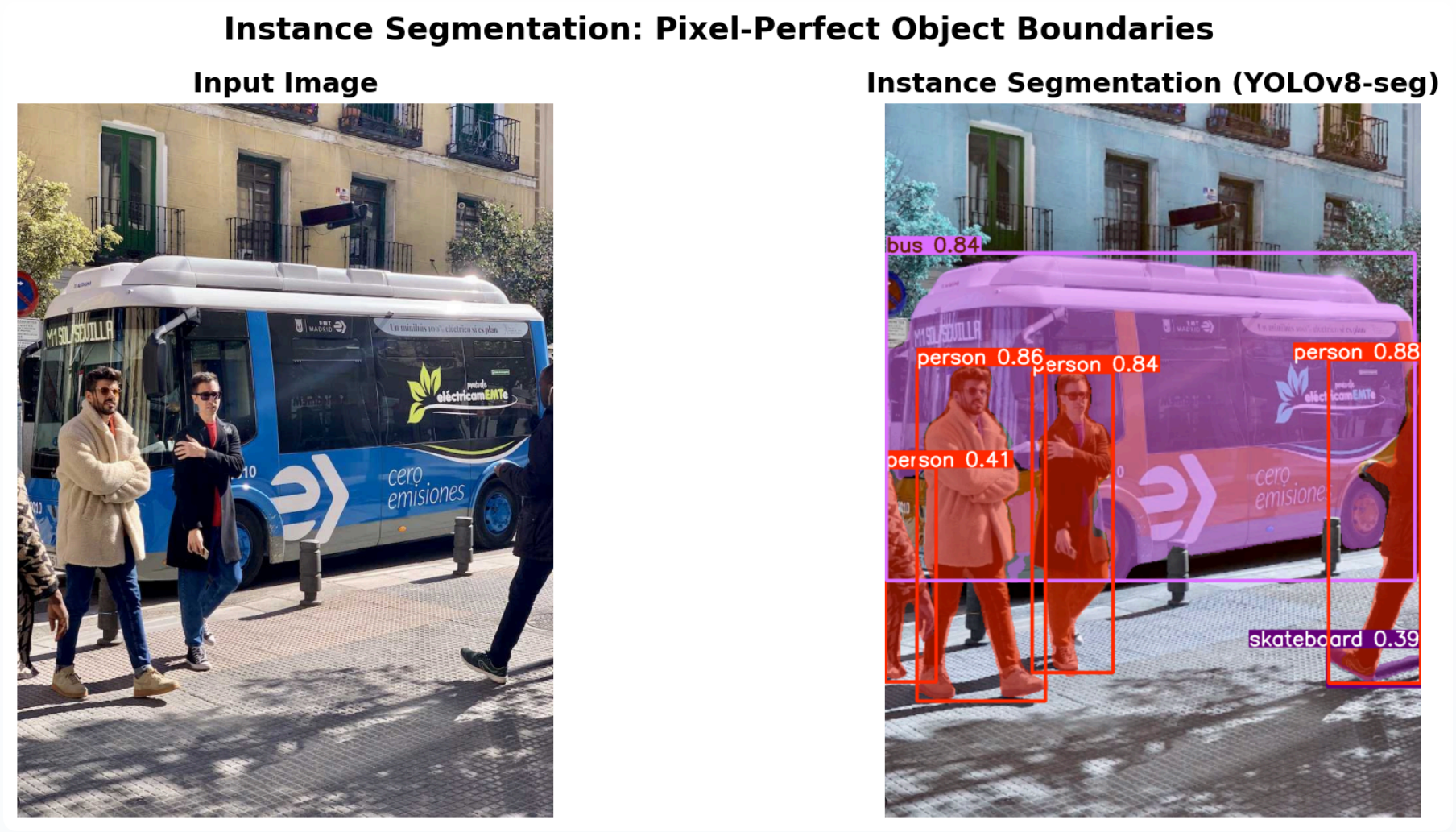
Key Difference

Semantic: pixel -> class

Instance: pixel -> class + object ID

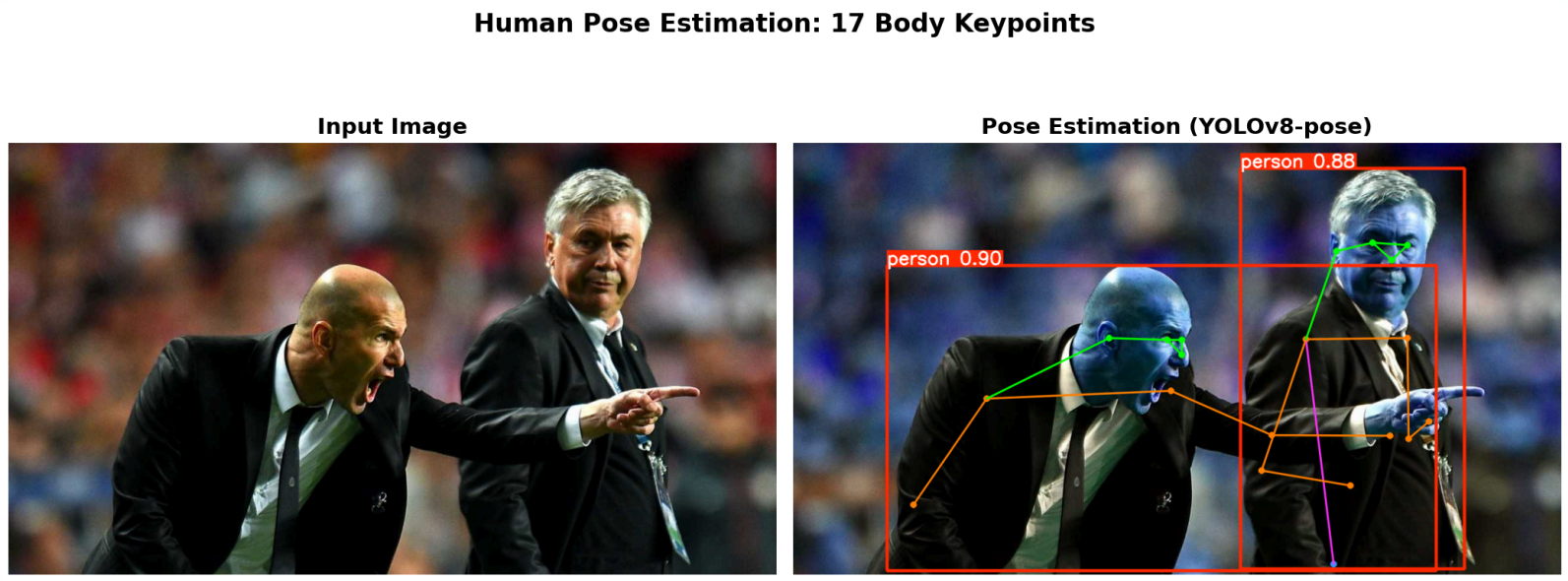
Self-driving cars need Instance to track individual vehicles!

Instance Segmentation in Action



Pose Estimation: Finding Body Keypoints

What: Find skeleton keypoints of humans or animals.



Applications: Fitness apps, motion capture, sign language, fall detection

Part 4: Natural Language Processing

Teaching Machines to Read & Write

The NLP Task Landscape

Task Type	What It Does	Example
Sentiment Analysis	Classify emotion	"Great movie!" → Positive
Named Entity Recognition	Find names, places, dates	"Sundar Pichai visited NYC"
Question Answering	Find answers in text	"When was Einstein born?"
Translation	Convert between languages	English → Hindi
Summarization	Shorten long text	1000 words → 50 words
Text Generation	Create new text	ChatGPT, Claude

> Modern LLMs (GPT-4, Claude) can do ALL of these with a single model!

Named Entity Recognition (NER)

Classify each word in the sequence:

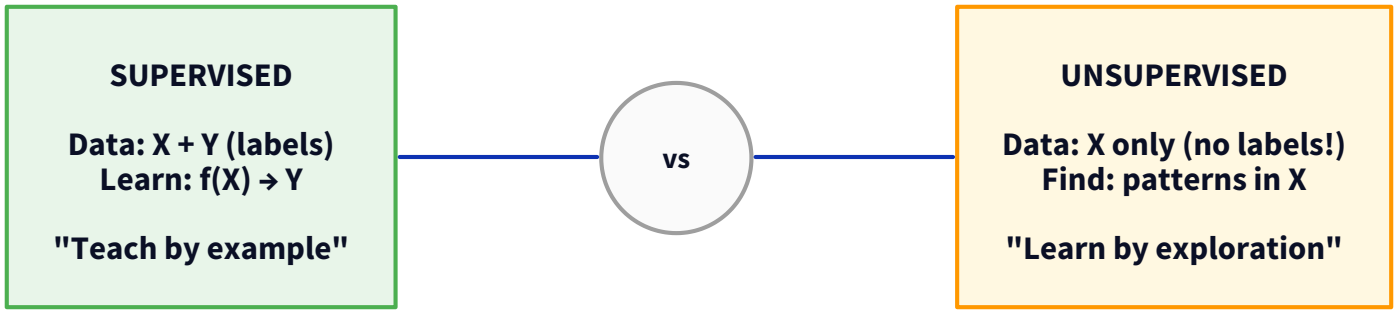
```
Input:    "Sundar  Pichai   visited  New    York   yesterday"
           |       |        |        |      |      |
           ▼       ▼        ▼        ▼      ▼      ▼
Output:   PER      PER       O       LOC    LOC     O

PER = Person Name
LOC = Location
O   = Other (not an entity)
```

> Think of it as "semantic segmentation for text" - every word gets a label!

Part 5: Unsupervised Learning

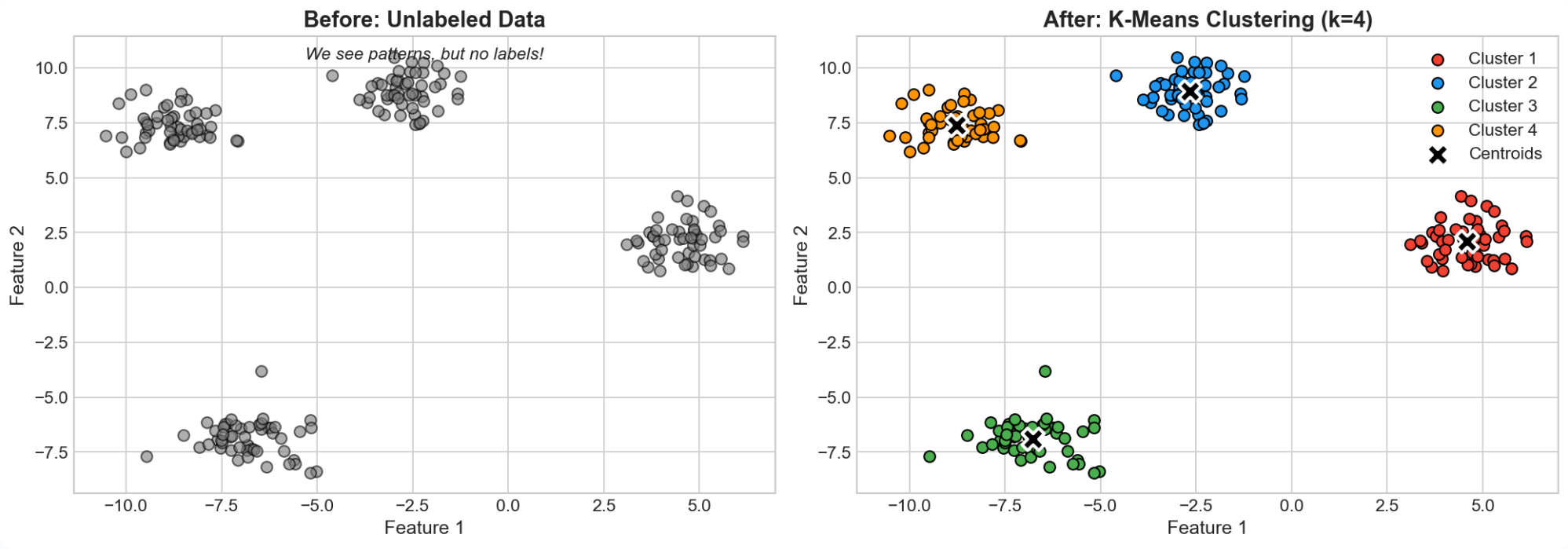
Finding Patterns Without Labels

Supervised vs Unsupervised



No one tells the model what to look for - it discovers structure on its own!

Clustering: K-Means in Action

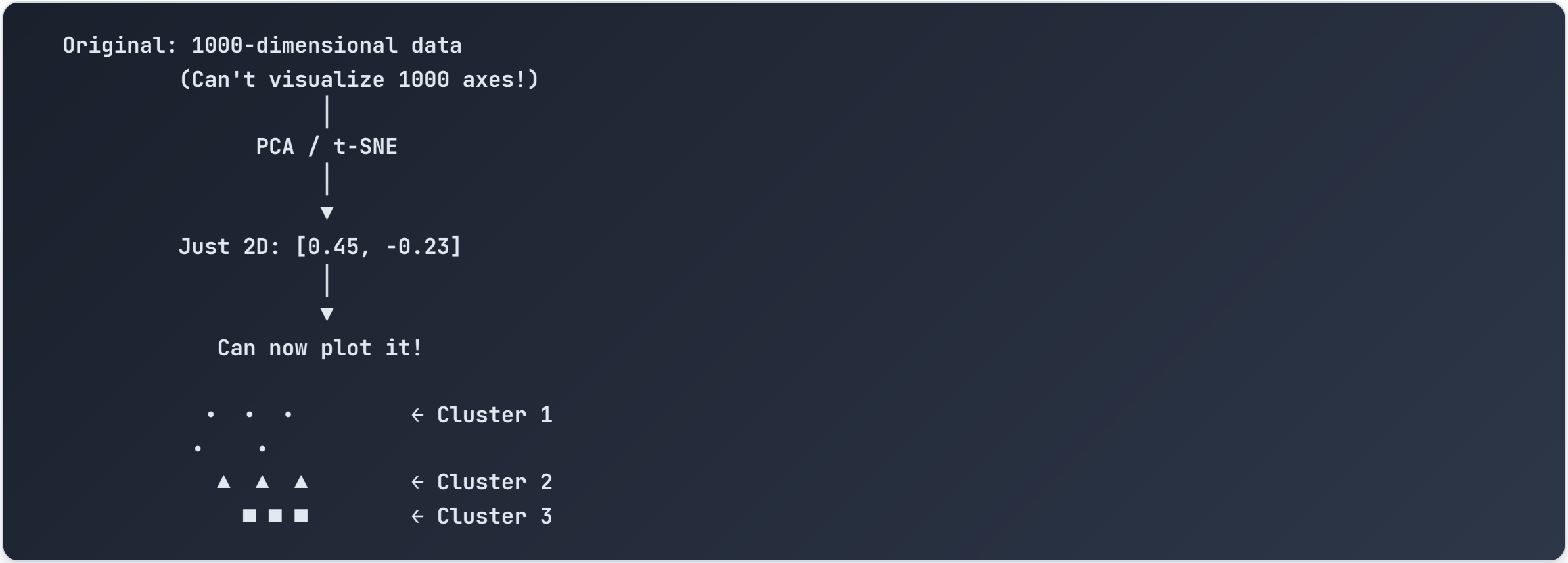


K-Means: No labels needed! The algorithm discovers natural groupings.

> **Applications:** Customer segmentation, gene expression analysis, document clustering

Dimensionality Reduction

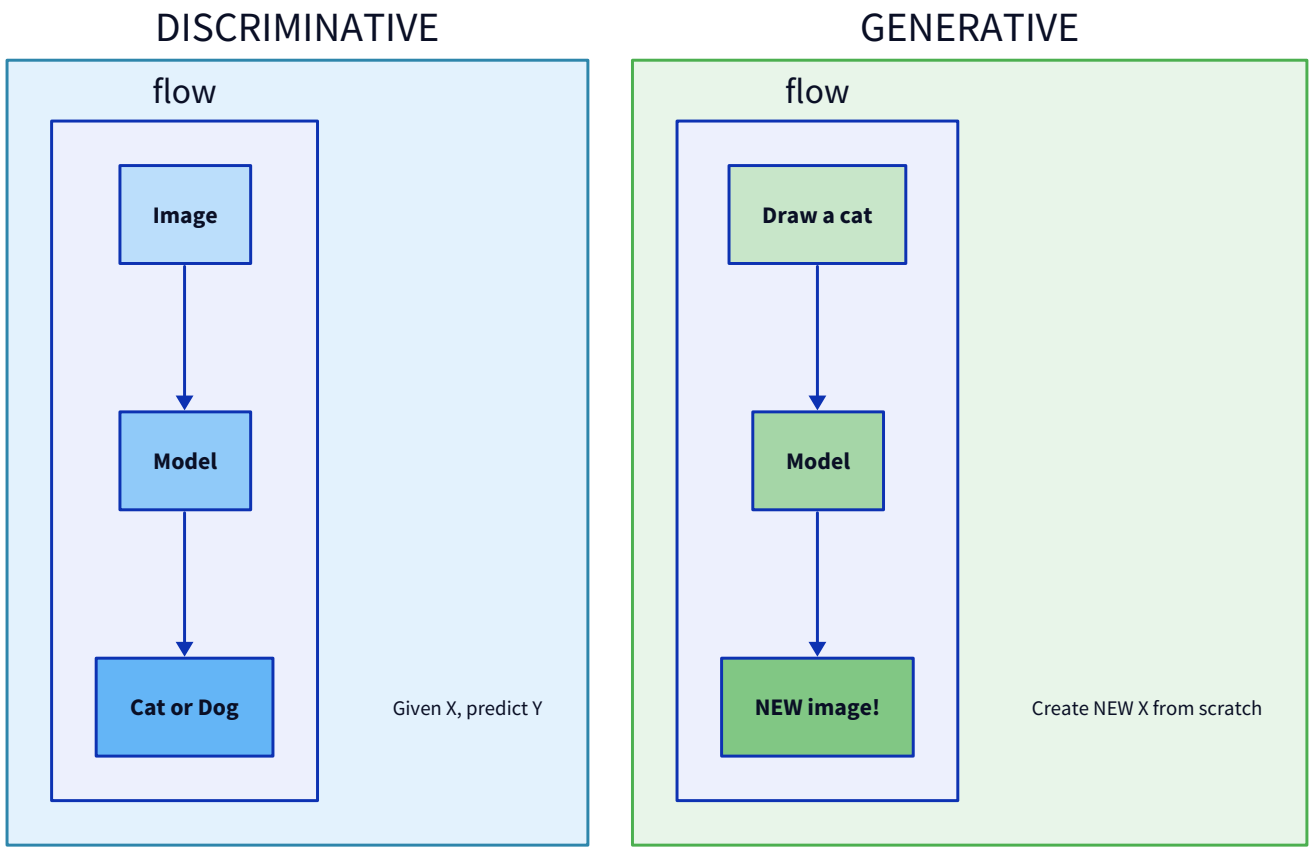
Problem: High-dimensional data is hard to visualize.



Part 6: Generative Models

Creating New Data

Generative vs Discriminative



The Generative AI Revolution

Domain	Tool	What It Does
Text	ChatGPT, Claude	Write essays, code, poems
Images	DALL-E, Midjourney, Stable Diffusion	Generate any image from text
Music	Suno, Udio	Create full songs with lyrics
Video	Sora, Runway	Generate realistic video clips
Code	GitHub Copilot, Claude	Write and debug code
Voice	ElevenLabs	Clone and synthesize voices

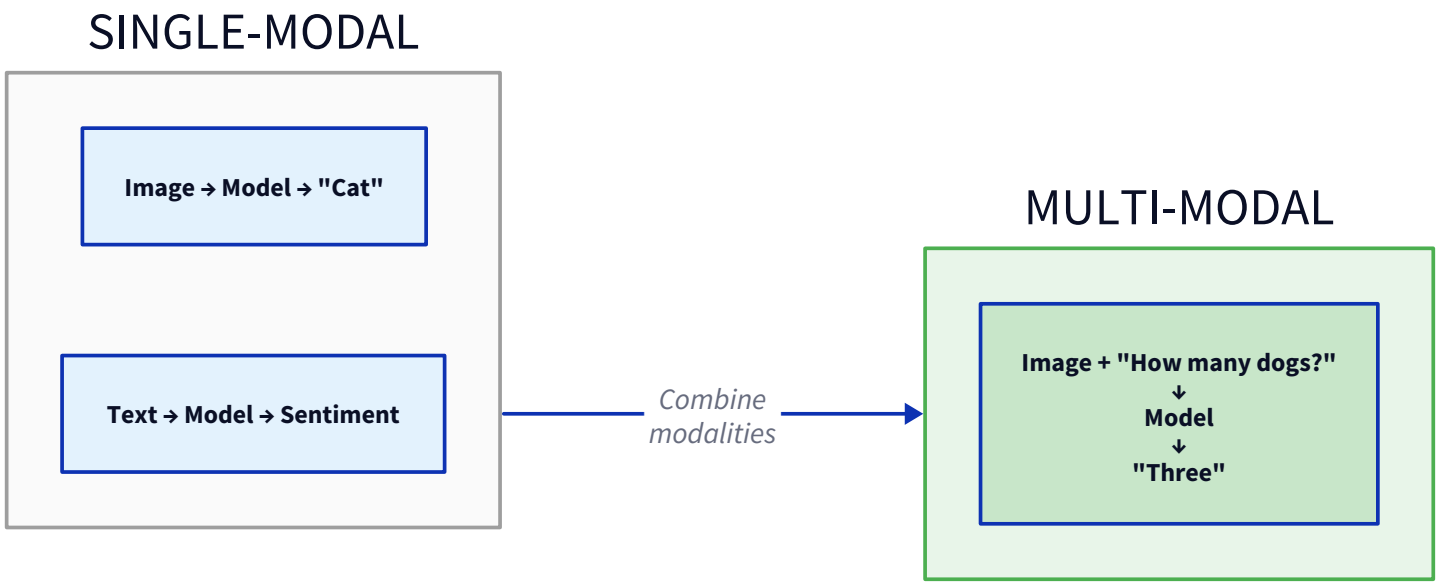
All of these generate NEW content that never existed before!

Part 7: Multimodal AI

Combining Everything

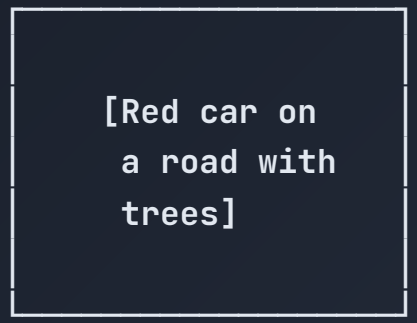
Multimodal = Multiple Modalities

Modalities: Text, Image, Audio, Video, etc.



Visual Question Answering (VQA)

Image:



Questions & Answers:

Q: "What color is the car?"
A: "Red"

Q: "Is it daytime or night?"
A: "Daytime"

Q: "How many trees are visible?"
A: "Four trees"

Requires BOTH:
- Understanding image
- Understanding language
- Reasoning about both!

Part 8: Reinforcement Learning

Learning Through Interaction

RL: A Different Paradigm

Goal: Maximize total reward over time through trial and error.

RL Examples

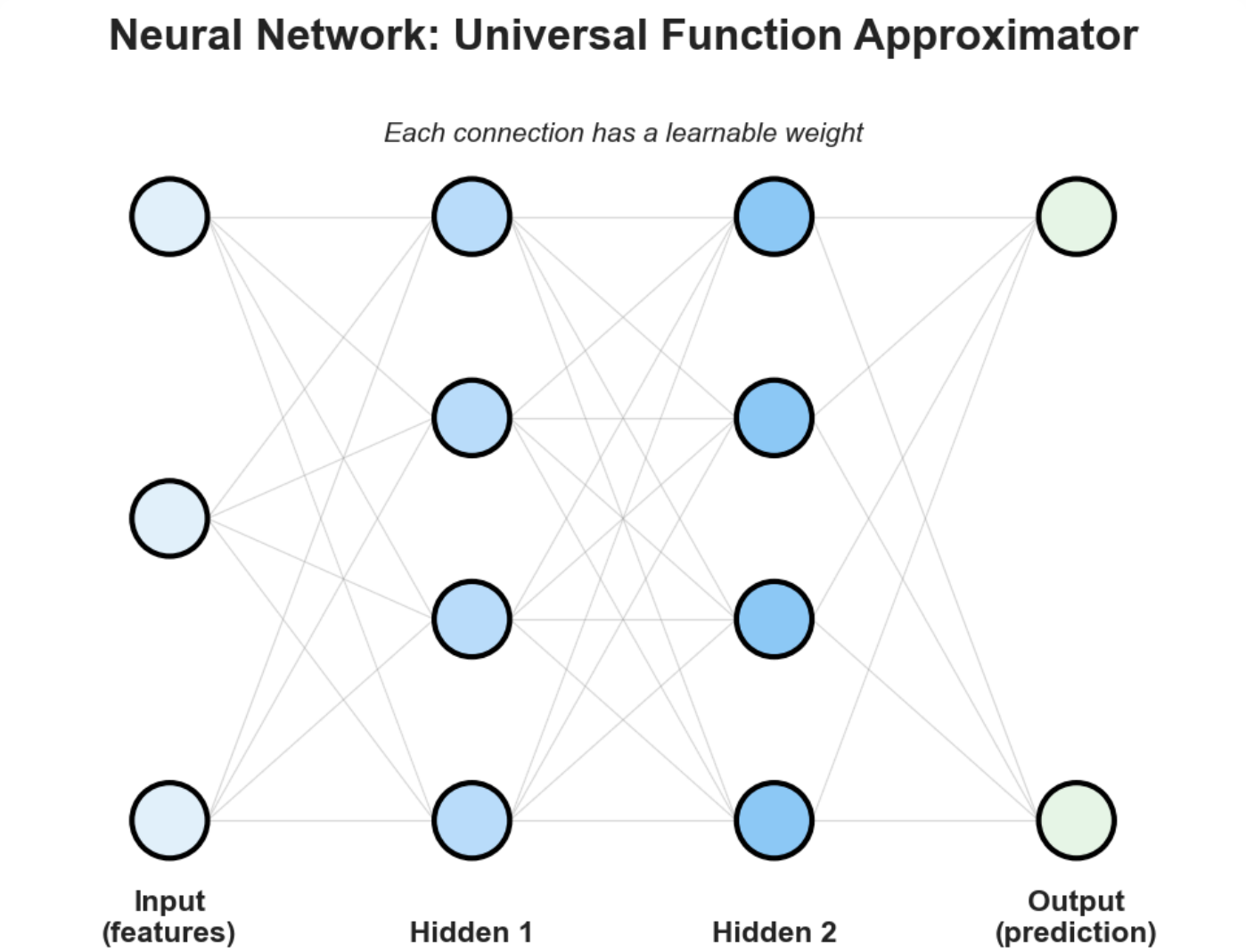
Domain	Example	What It Learned
Games	AlphaGo	Beat world champion at Go
Games	AlphaStar	Grandmaster at StarCraft II
Robotics	Boston Dynamics	Walk, run, dance
Infrastructure	Google Data Centers	40% energy reduction
AI Alignment	RLHF for ChatGPT	Be helpful and safe

RLHF (Reinforcement Learning from Human Feedback) is how ChatGPT learns to be helpful!

Part 9: The Common Thread

Neural Networks & Deep Learning

Neural Networks: The Universal Tool



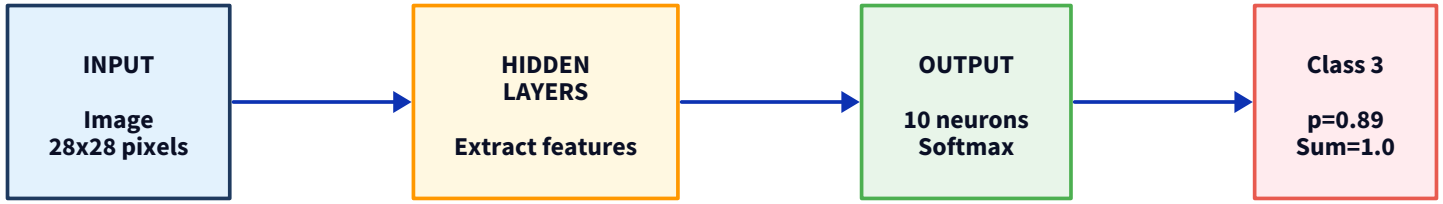
NN Output: Binary Classification

Output: 1 neuron with **Sigmoid** activation → probability $p \in [0, 1]$

Loss: Binary Cross-Entropy = $-[y \cdot \log(p) + (1-y) \cdot \log(1-p)]$

> Example: Disease prediction, spam detection, fraud detection

NN Output: Multi-class Classification

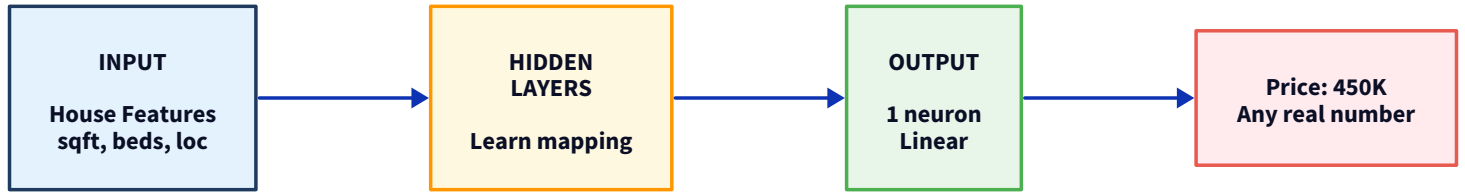


Output: C neurons with **Softmax** → probabilities sum to 1.0

Loss: Categorical Cross-Entropy = $-\Sigma\, y_i \cdot \log(p_i)$

> Example: Digit recognition (10 classes), ImageNet (1000 classes)

NN Output: Regression

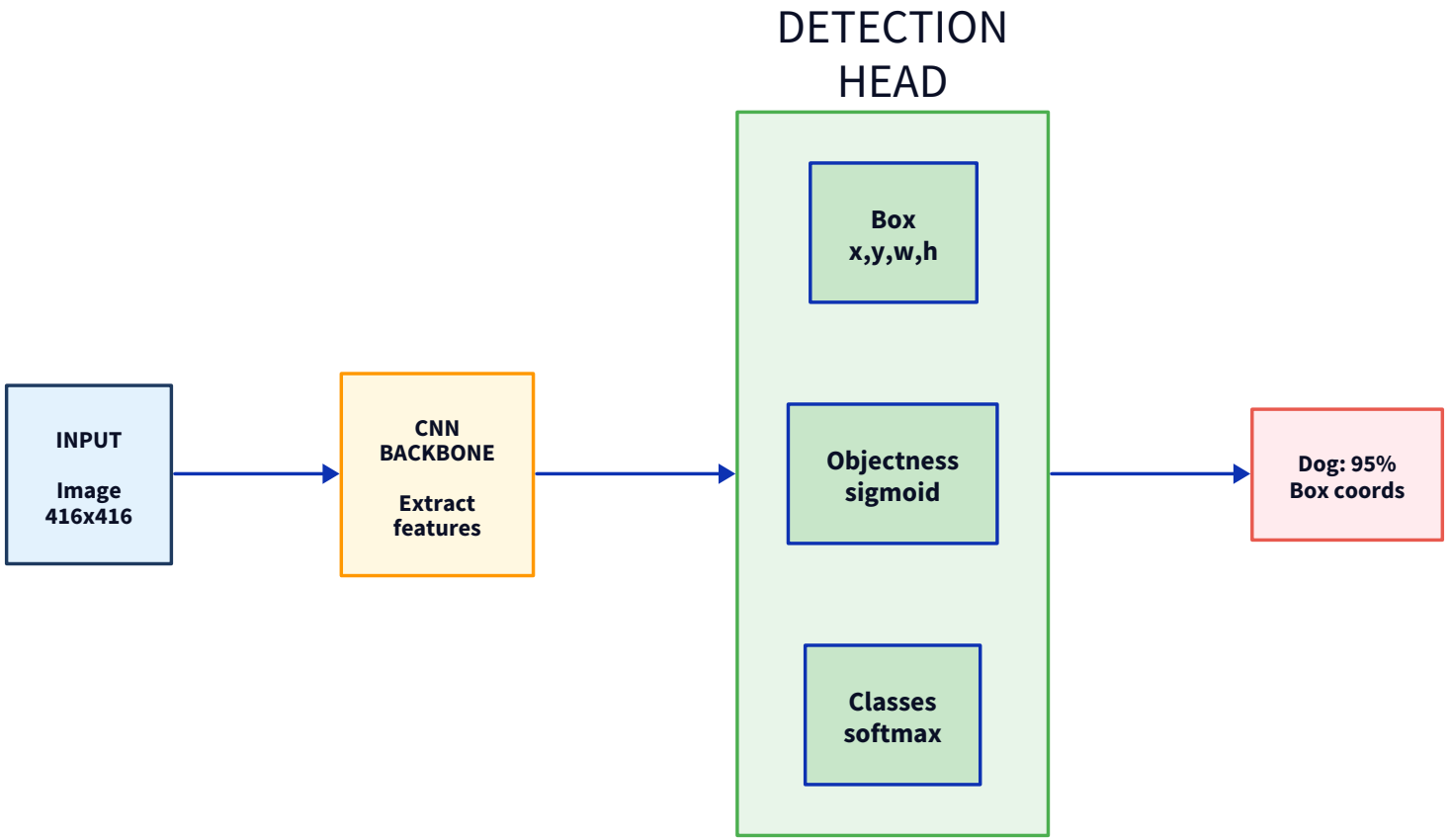


Output: 1 neuron with **No activation** (linear) → any real number

Loss: Mean Squared Error (MSE) = $(1/n)\Sigma(y_i - \hat{y}_i)^2$

> Example: House prices, stock prediction, age estimation

NN Output: Object Detection (Multi-task)

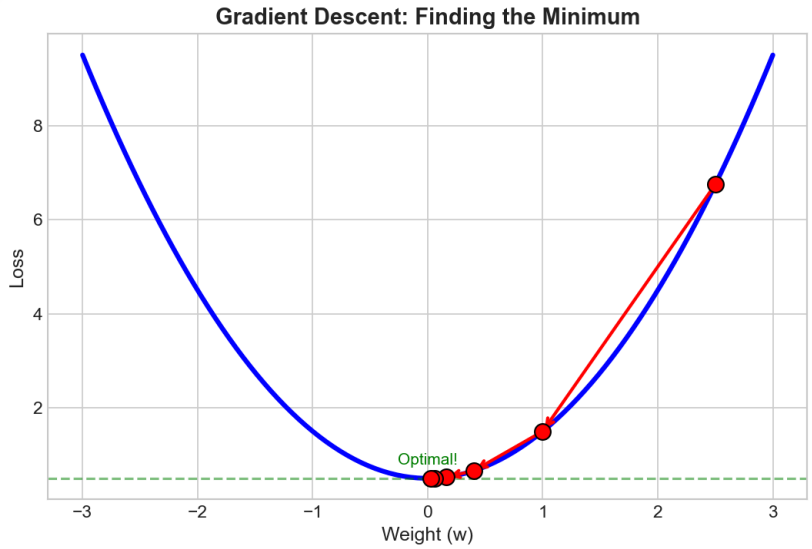


NN Output: Summary

Task	Output Neurons	Activation	Loss Function
Binary Classification	1	Sigmoid	Binary Cross-Entropy
Multi-class (C classes)	C	Softmax	Categorical Cross-Entropy
Multi-label	C	Sigmoid (each)	Binary CE (per label)
Regression	1 (or k)	None/Linear	MSE or MAE
Detection	B × (5 + C)	Mixed	Multi-part loss

> The output layer design tells you everything about the task type!

How Neural Networks Learn: Gradient Descent



Gradient Descent Update Rule:

$$w_{new} = w_{old} - \eta \cdot \nabla L$$

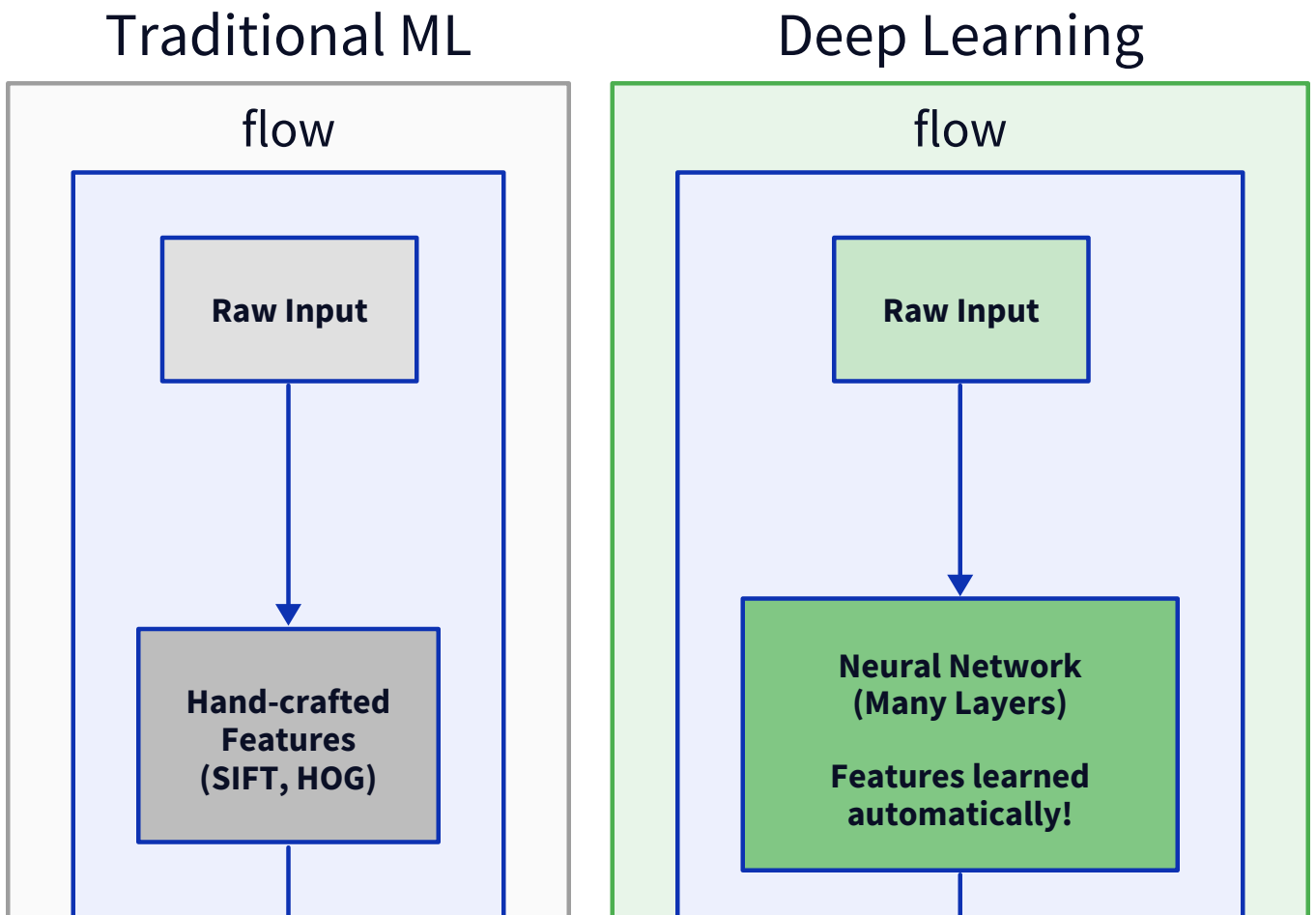
η = learning rate (step size)

∇L = gradient (direction of steepest ascent)

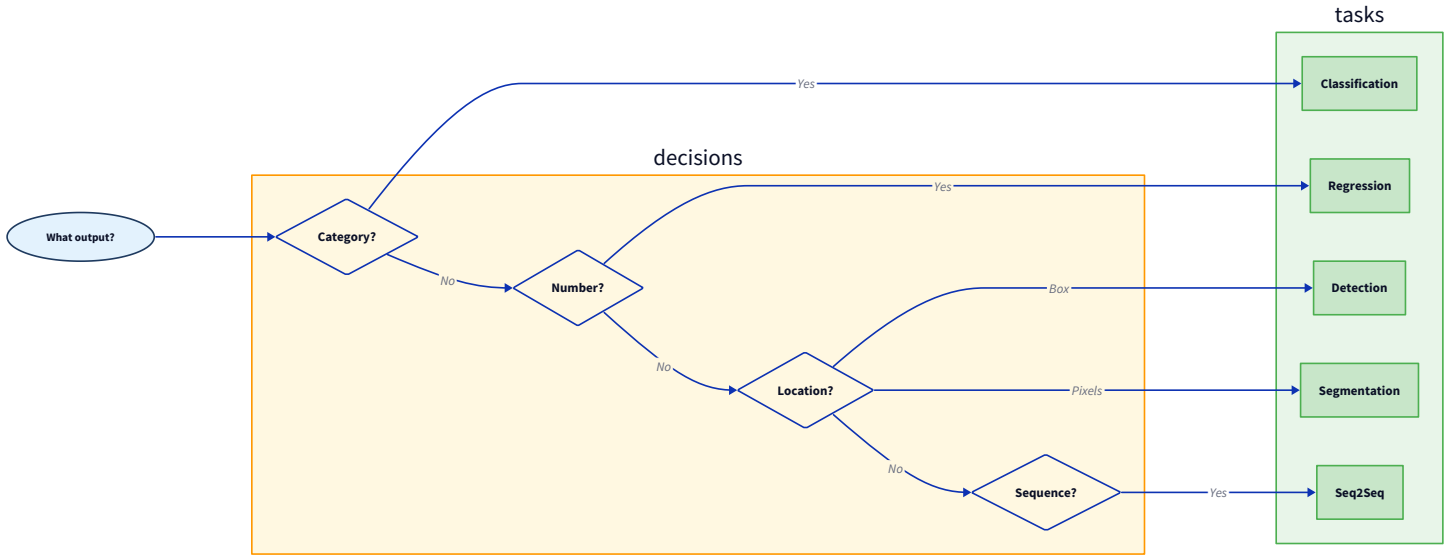
We go OPPOSITE to gradient to minimize loss!

Training = Finding the weights that minimize the loss function

The Deep Learning Revolution



The Decision Flowchart



Key Takeaways

1. **Classification** - Predict a category (discrete)
2. **Regression** - Predict a number (continuous)
3. **Detection** - Classification + Box Regression
4. **Segmentation** - Classification for every pixel
5. **Seq2Seq** - Sequence in, sequence out
6. **Unsupervised** - Find patterns without labels
7. **Generative** - Create new data
8. **Multimodal** - Combine text, images, audio
9. **RL** - Learn from rewards through interaction

> Understanding the output type tells you which family of techniques to use!

Coming Up Next

Lecture 3: Language Models

- Next Token Prediction
- Pre-training, SFT, RLHF
- From GPT to ChatGPT

Lecture 4: Object Detection

- YOLO and beyond
- Real-time detection

Thank You!

"All models are wrong, but some are useful."

— George Box

Key Takeaway

Match the **output type** to the right **task formulation**

Questions?