12. Feature Hierarchy

Neural Networks - From Brain to Business

Learning Goal

Understand how different layers learn increasingly abstract representations.

Key Concept (1/2)

Neural networks learn **hierarchical representations**. Each layer transforms the data into a more abstract form:
- **Input layer**: Raw features (price, volume, sentiment) - **Hidden layer 1**: Simple patterns (uptrend, high volume, positive sentiment) - **Hidden layer 2**: Combinations (momentum + volume spike = strong signal) - **Output layer**: Final decision (BUY/SELL with confidence)

Key Concept (2/2)

This hierarchy emerges automatically through training. The network learns what intermediate representations are useful for the final task. Early layers detect simple patterns; later layers combine these into complex concepts.
In image recognition, this is visible: early layers detect edges, middle layers detect shapes, late layers detect objects. For financial data, the learned features are less visually interpretable but follow the same principle.

Visualization

12_feature_hierarchy/feature_hierarchy.pdf

Key Formula

Layer-by-layer transformation:

$$h^{[1]} = f(W^{[1]}x + b^{[1]}) \quad \text{(simple patterns)}$$

$$h^{[2]} = f(W^{[2]}h^{[1]} + b^{[2]}) \quad \text{(complex patterns)}$$

$$\hat{y} = \sigma(W^{[3]}h^{[2]} + b^{[3]}) \quad \text{(decision)}$$

Each layer builds on the representations from the previous layer.

Intuitive Explanation

Think of a corporate decision-making process:
1. **Analysts (Input)**: Collect raw data - stock prices, trading volumes 2. **Junior managers (Hidden 1)**: Identify simple patterns - "prices rising," "unusual volume" 3. **Senior managers (Hidden 2)**: Combine patterns - "rising prices + high volume = momentum" 4. **Executive (Output)**: Make final call - "Confidence: 68% BUY"
Each level adds interpretation and abstraction. The executive doesn't need raw numbers - they need the synthesized judgment of their team.

Practice Problem 1

Problem 1
A network for fraud detection has 3 hidden layers. What might each layer learn?

Solution

Layer 1 - **Simple anomalies:** - Transaction amount unusually high - Time of transaction (late night) - Location different from usual - Transaction frequency spike
Layer 2 - **Pattern combinations:** - High amount + unusual time = suspicious - New location + high frequency = potential card theft - Normal amount + normal patterns = likely legitimate
Layer 3 - **Complex fraud signatures:** - Combination of multiple suspicious patterns - Sequential transaction patterns (testing then big purchase) - Network of related suspicious accounts
Output - **Fraud score:** - Probability of fraud (0-100%) - Decision threshold (e.g., flag if ¿ 80%)
Each layer abstracts further from raw data toward the final fraud determination.

Practice Problem 2

Problem 2
Why can't we interpret what middle layer neurons have learned as easily as input features?

Solution

Reasons for difficulty:
1. **Distributed representations**: Information is spread across many neurons, not localized in one
2. **Non-linear transformations**: Multiple activation functions make the relationship to inputs complex and non-obvious
3. **Abstract representations**: Middle layers don't correspond to concepts humans naturally think about - they optimize for the task, not human understanding
4. **Lack of semantic meaning**: Unlike "price" or "volume," a hidden neuron might activate for a combination that has no simple label
5. **Entangled features**: Multiple concepts are mixed together in the same neuron activations

Contrast with input features: - Input features have clear meaning: "closing price = \$150" - Hidden features are abstract: "neuron 7 in layer 2 = 0.83" - what does that mean?
This is why neural networks are often called "black boxes" - they work well but are hard to interpret.

Key Takeaways

- Networks learn hierarchical representations automatically
- Early layers: simple patterns; Later layers: complex combinations
- Deeper networks can represent more abstract concepts
- Hidden representations are often not human-interpretable
- More layers isn't always better - match complexity to problem