Data Mining:

Concepts and Techniques

(3rd ed.)

Classification: Advanced Methods

Jiawei Han, Micheline Kamber, and Jian Pei

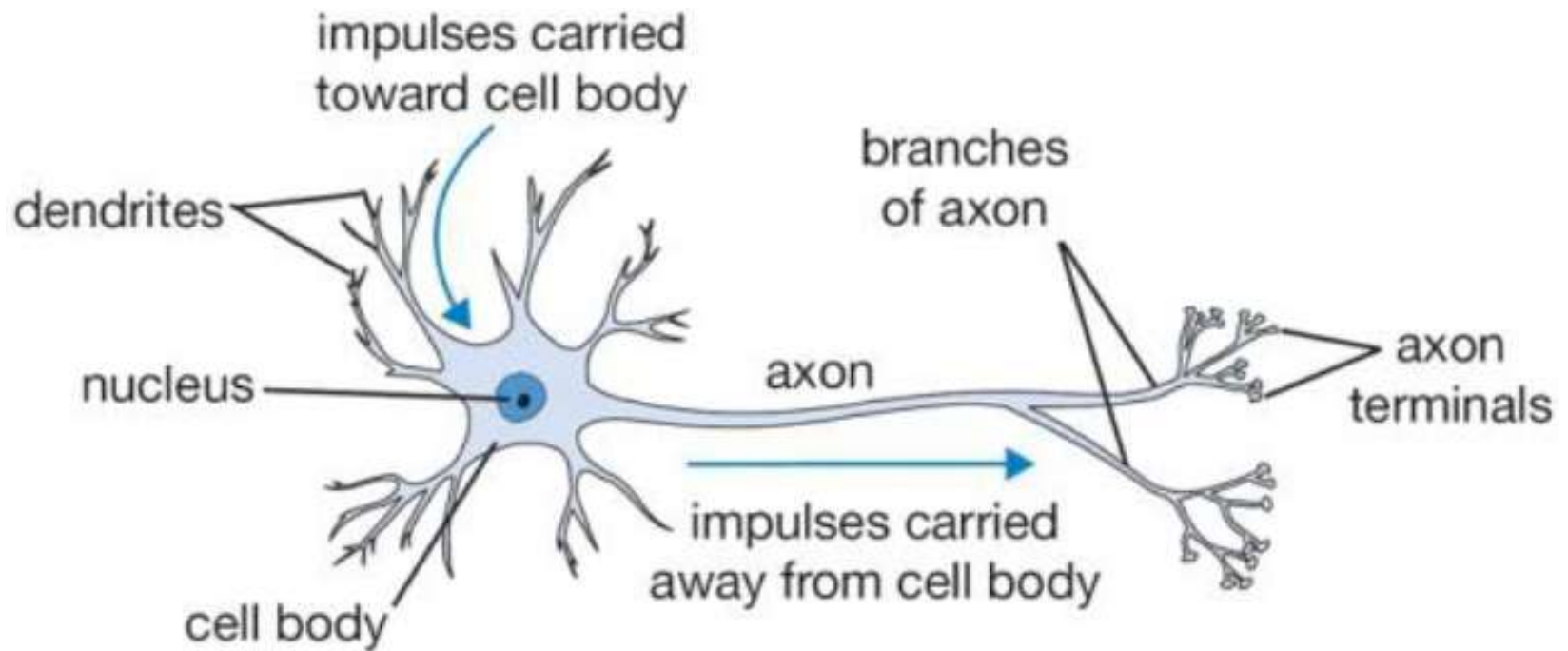University of Illinois at Urbana-Champaign &

Simon Fraser University

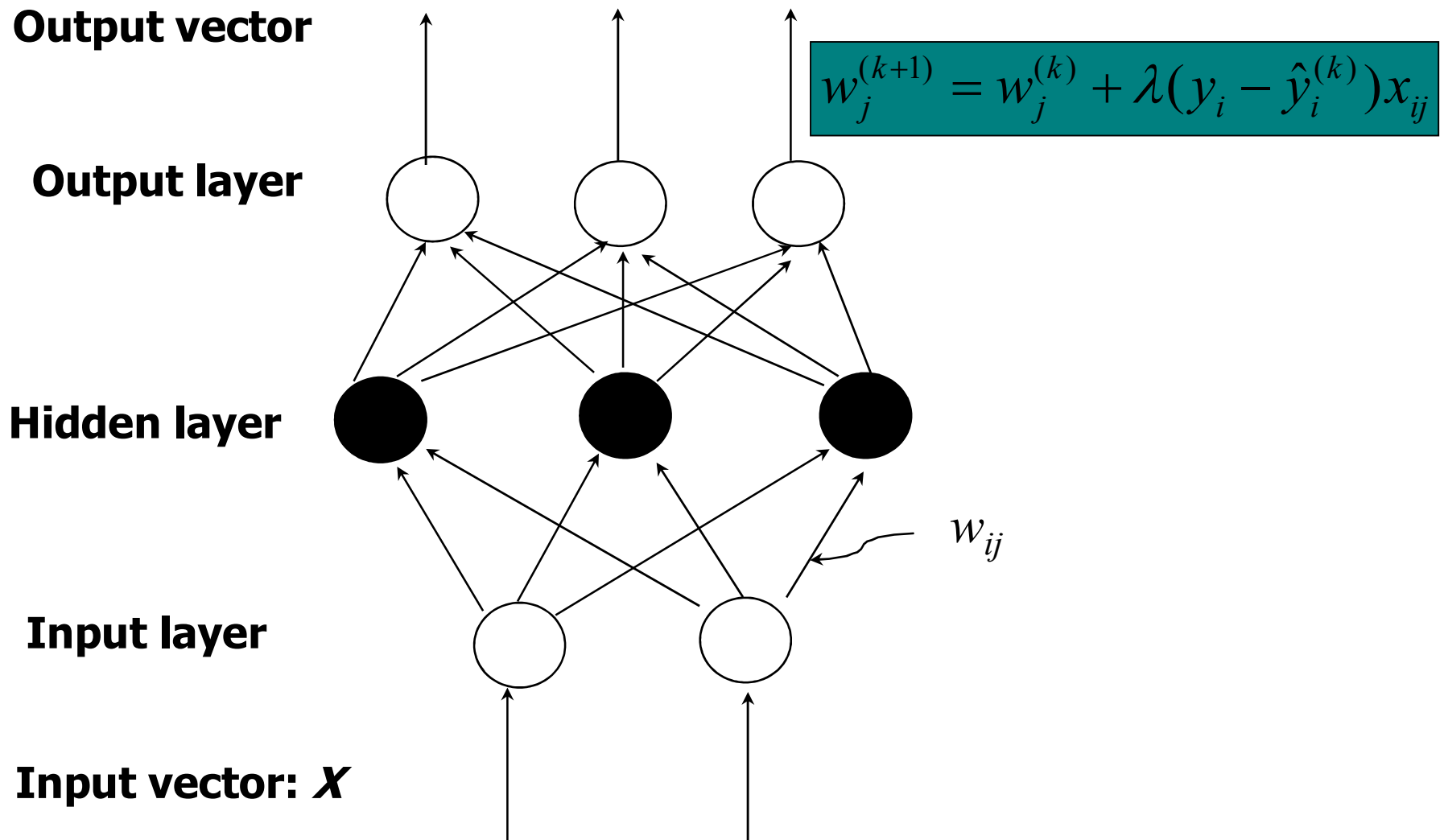# Neuron

Data Mining: Concepts and Techniques

Classification by Backpropagation

- Backpropagation: A **neural network** learning algorithm
- Started by psychologists and neurobiologists to develop and test computational analogues of neurons
- A neural network: A set of connected input/output units where each connection has a **weight** associated with it
- During the learning phase, the **network learns by adjusting the weights** so as to be able to predict the correct class label of the input tuples
- Also referred to as **connectionist learning** due to the connections between units

Neural Network as a Classifier

- Weakness
  - Long training time
  - Require a number of parameters typically best determined empirically, e.g., the network topology or "structure."
  - Poor interpretability: Difficult to interpret the symbolic meaning behind the learned weights and of "hidden units" in the network
- Strength
  - High tolerance to noisy data
  - Ability to classify untrained patterns
  - Well-suited for continuous-valued inputs *and outputs*
  - Successful on an array of real-world data, e.g., hand-written letters
  - Algorithms are inherently parallel
  - Techniques have recently been developed for the extraction of rules from trained neural networks

A Multi-Layer Feed-Forward Neural Network

Output vector

$$w_j^{(k+1)} = w_j^{(k)} + \lambda(y_i - \hat{y}_i^{(k)})x_{ij}$$

Output layer

Hidden layer

w_{ij}

Input layer

Input vector: X

How A Multi-Layer Neural Network Works

- The **inputs** to the network correspond to the attributes measured for each training tuple

- Inputs are fed simultaneously into the units making up the **input layer**

- They are then weighted and fed simultaneously to a **hidden layer**

- The number of hidden layers is arbitrary, although usually only one

- The weighted outputs of the last hidden layer are input to units making up the **output layer**, which emits the network's prediction

- The network is **feed-forward**: None of the weights cycles back to an input unit or to an output unit of a previous layer

- From a statistical point of view, networks perform **nonlinear regression**: Given enough hidden units and enough training samples, they can closely approximate any function
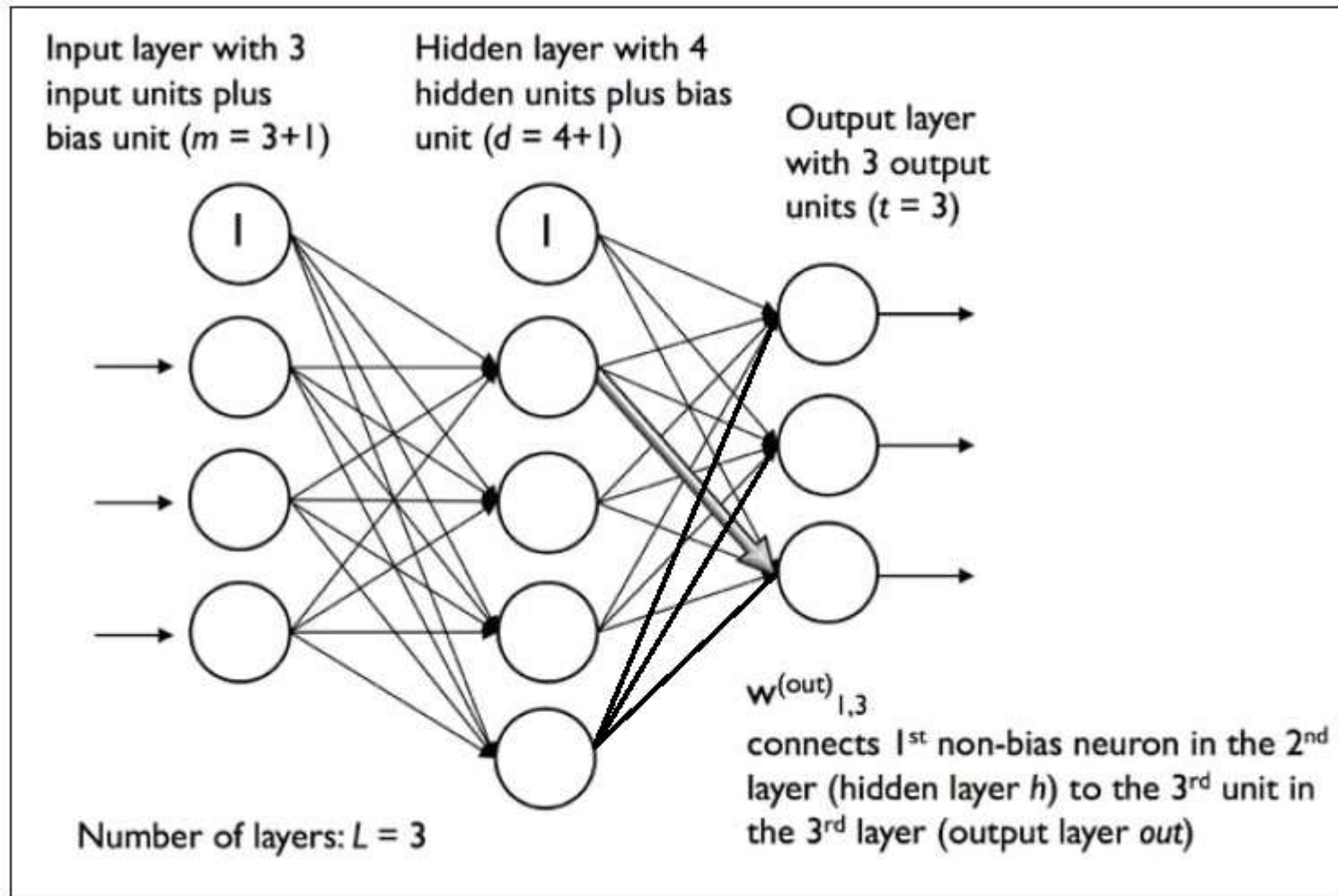
Defining a Network Topology

- Decide the **network topology:** Specify # of units in the *input layer*, # of *hidden layers* (if > 1), # of units in *each hidden layer*, and # of units in the *output layer*

- Normalize the input values for each attribute measured in the training tuples to [0.0—1.0]

- One **input** unit per domain value, each initialized to 0

- **Output**, if for classification and more than two classes, one output unit per class is used

- Once a network has been trained and its accuracy is **unacceptable**, repeat the training process with a *different network topology* or a *different set of initial weights*
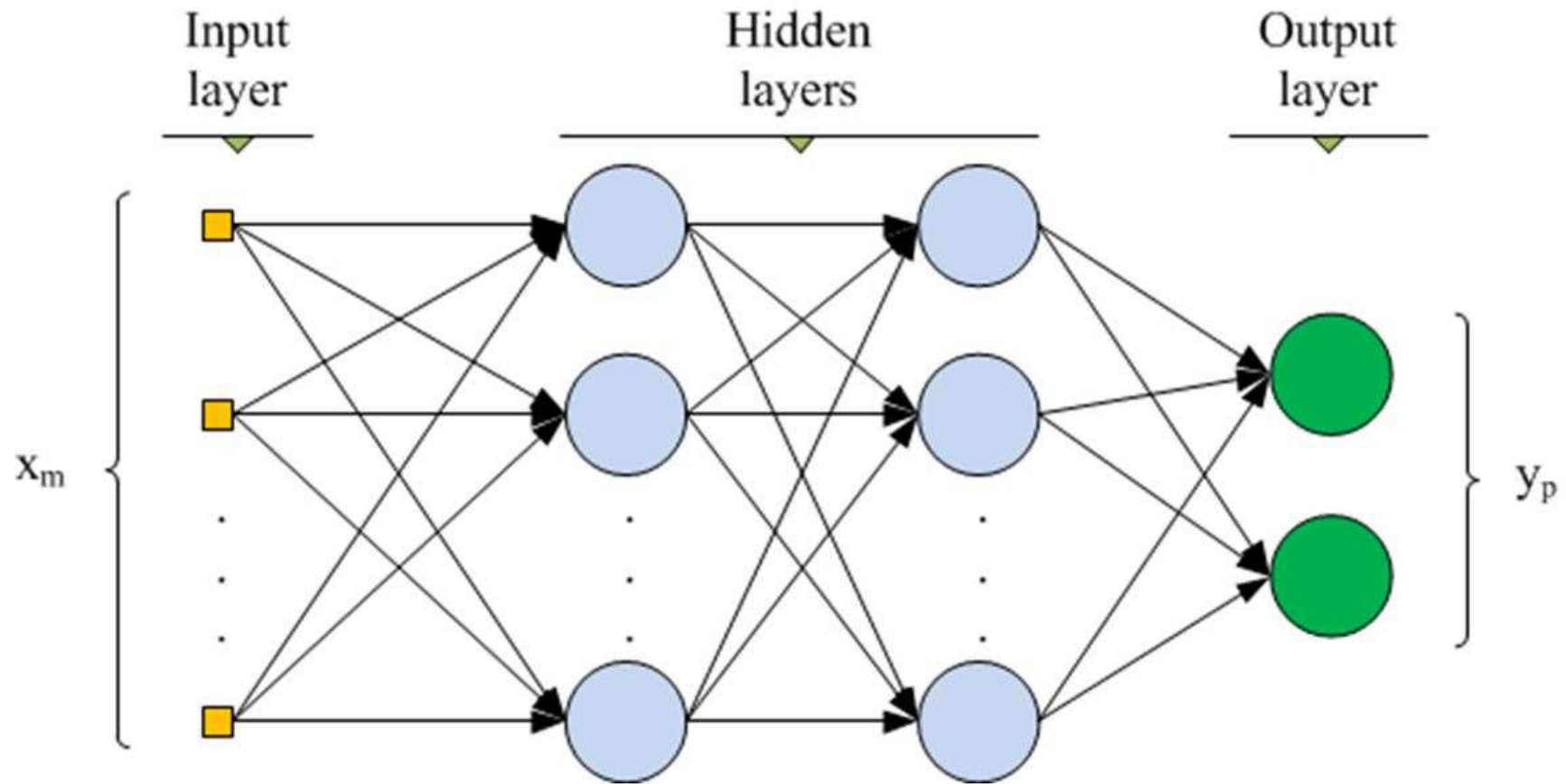
Backpropagation

- Iteratively process a set of training tuples & compare the network's prediction with the actual known target value

- For each training tuple, the weights are modified to **minimize the mean squared error** between the network's prediction and the actual target value

- Modifications are made in the "**backwards**" direction: from the output layer, through each hidden layer down to the first hidden layer, hence "**backpropagation**"

- Steps
    - Initialize weights to small random numbers, associated with biases
    - Propagate the inputs forward (by applying activation function)
    - Backpropagate the error (by updating weights and biases)
    - Terminating condition (when error is very small, etc.)

Backpropagation

Input layer with 3 input units plus bias unit ($m = 3+1$)

Hidden layer with 4 hidden units plus bias unit ($d = 4+1$)

Output layer with 3 output units ($t = 3$)

$w^{(out)}_{1,3}$ connects 1st non-bias neuron in the 2nd layer (hidden layer h) to the 3rd unit in the 3rd layer (output layer out)

Number of layers: $L = 3$

Backpropagation

Data Mining: Concepts and Techniques

Self Learned Cat

We then ran experiments that asked, informally: If we think of our neural network as simulating a very small-scale "newborn brain," and show it YouTube video for a week, what will it learn? Our hypothesis was that it would learn to recognize common objects in those videos. Indeed, to our amusement, one of our artificial neurons learned to respond strongly to pictures of... cats. Remember that this network had never been told what a cat was, nor was it given even a single image labeled as a cat. Instead, it "discovered" what a cat looked like by itself from only unlabeled YouTube stills. That's what we mean by self-taught learning.

One of the neurons in the artificial neural network, trained from still frames from unlabeled YouTube

Self Learned Cat

Data Mining: Concepts and Techniques