ANN Overview

(Artificial) neural networks

- A computing model inspired by the biological brain
- Flexible and very powerful
- Can be adjusted for different learning tasks (supervised, unsupervised, etc.)

Reasons to study neural networks

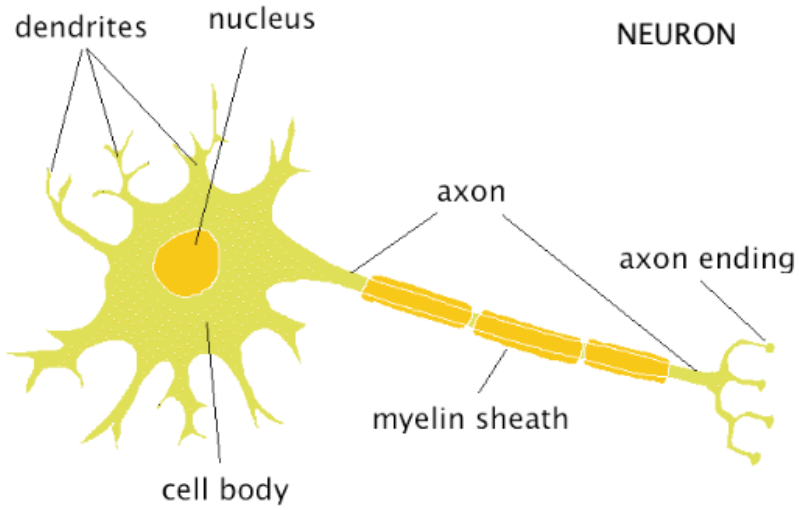
- We usually cannot directly study brain
  - We can with fMRI and EEG but not everyone can afford it
  - There are limitations (yet)
- To understand what we are likely good (bad) at
  - Supposed to be good at things that we are good at: e.g., vision
  - And bad at things that we are bad at: e.g., arithmetic
- A powerful paradigm to solve real-world problem (this course)

Our brain

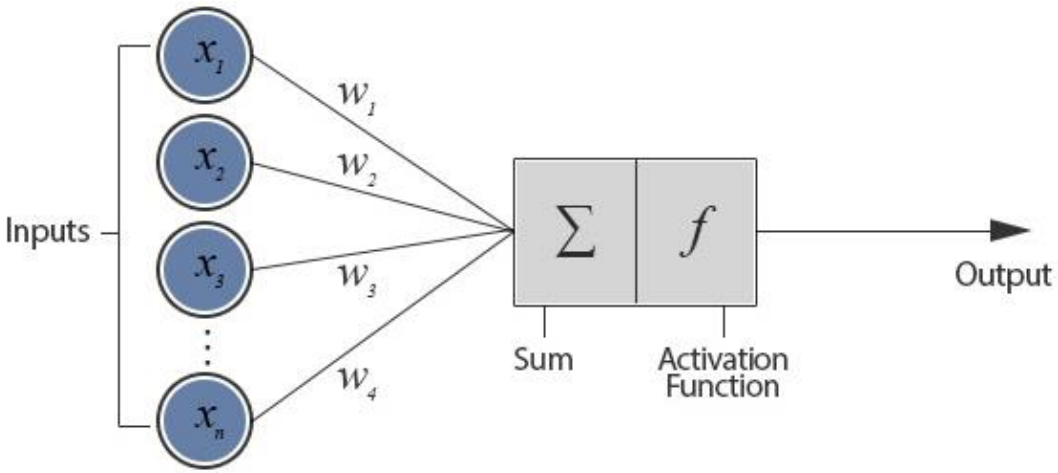
- Average weight 1.4 kg, about 2% of total body weight
- Responsible ~20% of our energy consumption!
  - ~12.6 Watts
  - It sounds a like but is extremely efficient. A typical GPU server requires ~1000 Watts
- Composed of neurons interconnected to each other

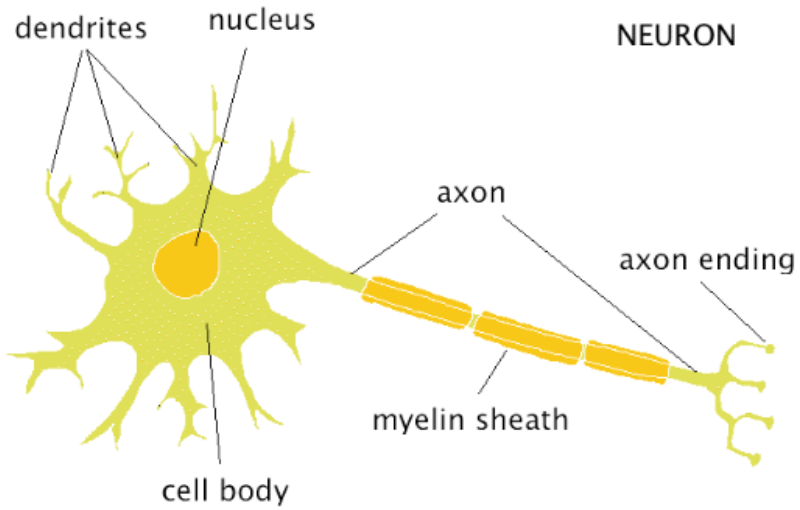
Biological neurons

- Dendrites collect chemicals
- Neuron may “fire” based on the chemical input
- Axon ending will generate chemicals those will in turn be consumed by other neurons



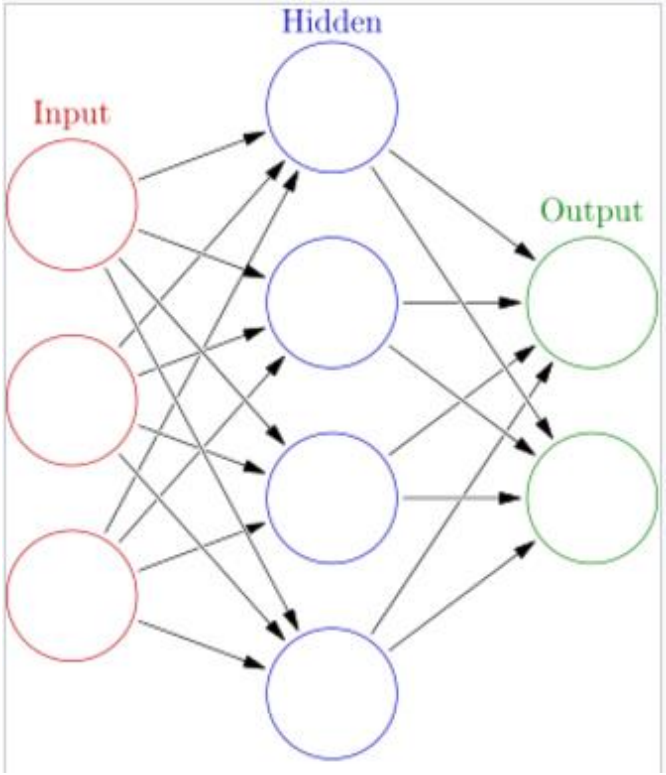
Biological neurons





Neural networks

Single layer network



Multiple layer (deep) network

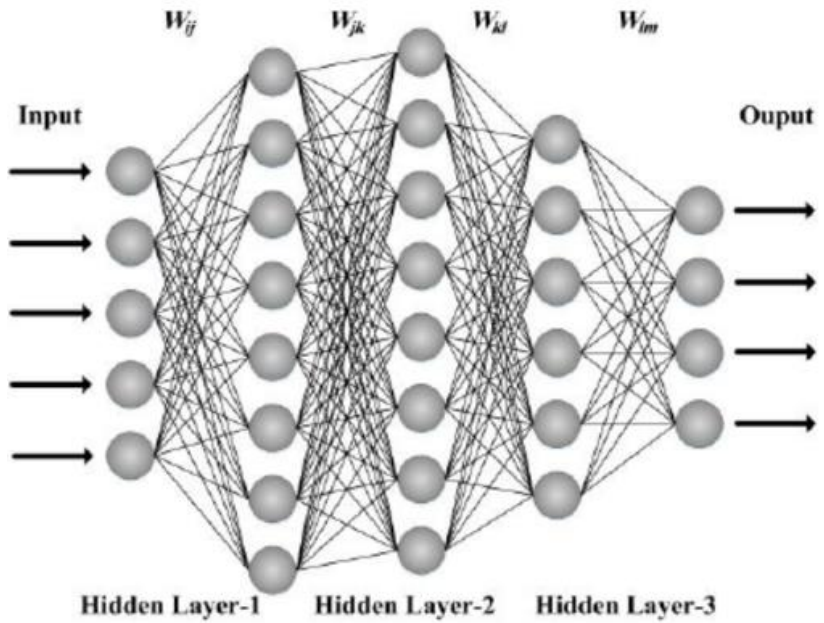
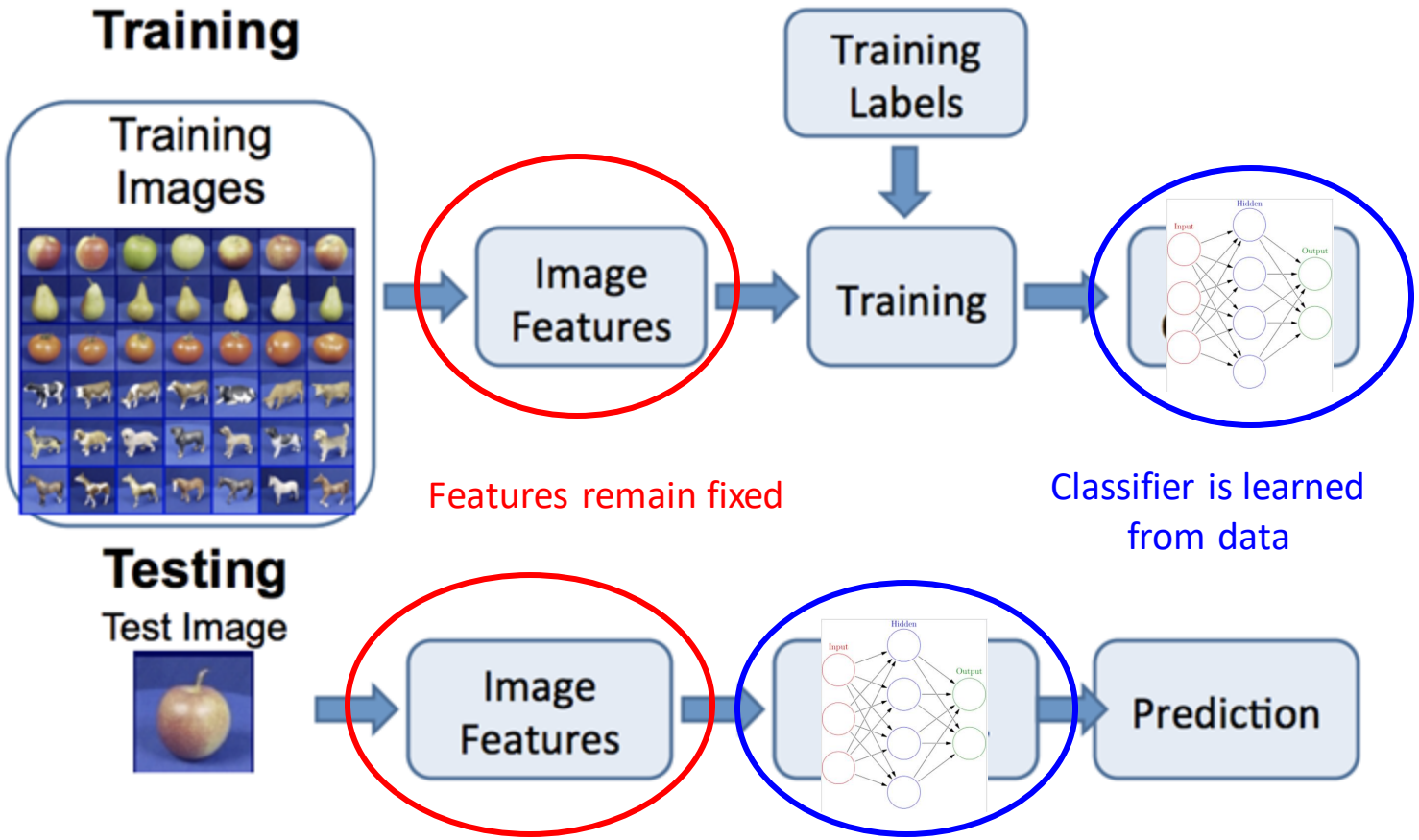


Image Classification before DL



Problem:
How do we know which features to use? We may need different features for each problem!

Solution:
Learn the features jointly with the classifier!

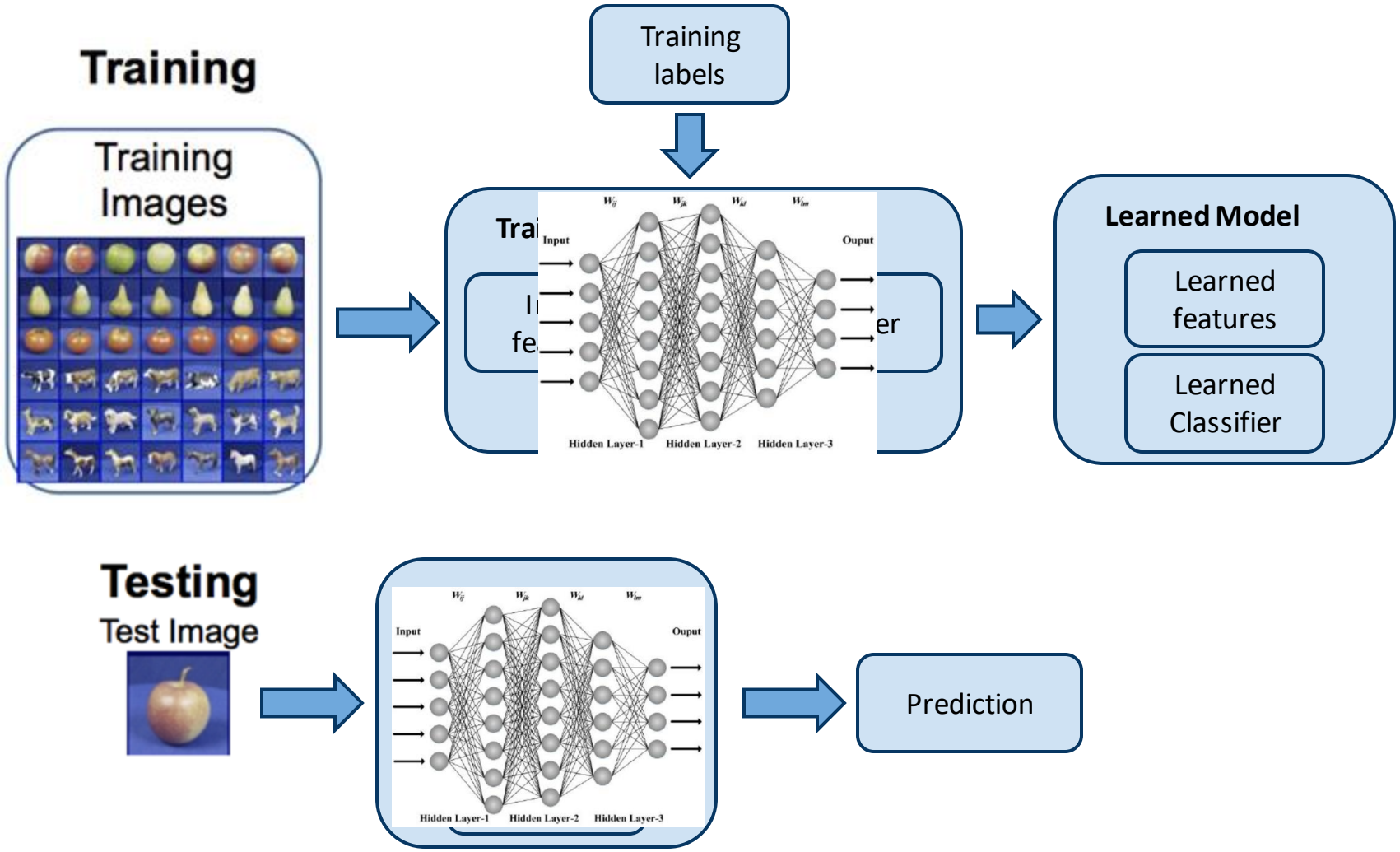
Figure credit: D. Hoiem, L. Lazebnik

Linear regression/Logistic regression/SVM are ANNs

Table 1: Traditional ML techniques that can be interpreted as a single-neuron "network"

Problem	Technique	Activation	Loss	Regularization
Regression	Linear regression	Identity/none	Square error	None
	Ridge regression			l_2-norm
	Lasso			l_1-norm
Binary classification	Logistic regression	Logistic function	cross-entropy	None
	Hard-margin SVM	Identity/none	"0-∞" loss	l_2-norm
	Soft-margin SVM		hinge loss	
Multi-class classification	Softmax regression classifier	Softmax	cross-entropy	None
	Multi-class (soft-margin) SVM	Identity/none	multi-class hinge loss	l_2-norm

Image Classification after DL



A very brief history of ANN

- Warren McCulloch and Walter Pitts introduced Threshold Logic Unit as a computational model of neuron in 1943
- Donald Hebb created a hypothesis of learning (Hebbian theory) based on neural plasticity (increase synaptic efficacy by repeated and persistent stimulation) in late 1940's
- Frank Rosenblatt created perceptron in 1958 but no concrete learning method appeared to be described
- Bernard Widrow and Marcian Hoff of Stanford developed "ADALINE" and "MADALINE" (Mulitple ADAptive LiINear Elements) in 1959. First neural networks to tackle real world problem. It is still used in air traffic control systems (really?)

A very brief history of ANN (con't)

- Perceptron got high expectation in early year but neural science does not catch up. And the traditional von Neumann and Turing architecture took over the computing scene. Ironically, von Neumann himself suggested to imitate neuron function with telegraph relays or vacuum tubes
- Neural network research was hit hard by the introduction of the book Perceptrons by Marvin Minsky and Seymour Papert. It proved the limitation of perceptrons but combined with the initial hype, people lost trust of the potential of neural networks and we entered the "first dark age" of neural networks

A very brief history of ANN (con't)

- The interest to neural network slowly revived with the invention of the backpropagation algorithm by Paul Webos in 1974. The algorithm was reinvented by many other such as Parker (1985) and LeCun (1985)
- Hopfield popularizes a form of bi-directional networks (Hopfield networks) in the 1980's and neural network research was blooming in that decade
- But as the support vector machine (SVM) by Vladmir Vapnik was popularized in late 1980's and early 1990's. Neural network slowly lost favors. SVM took hold instead of neural networks because it has more elegant math and worked better at that time
  - Less parameters to tune
  - Computers were too slow then and labeled datasets were too small

A very brief history of ANN (con't)

- Neural networks got a come-back for the last decade as recurrent neural networks and deep feedforward neural networks won numerous competitions in both pattern recognition and machine learning domains
- Fast GPUs were a key for the come-back. Relatively cheap and powerful computing resources are now widely available. And the wide spread of large public labeled datasets helped a whole lot too

Conclusions

- Machine learning allows us to automatically generate programs to solve various tasks by learning from data
- Different machine learning types:
  - Supervised learning (e.g., object recognition)
    - Classification: logistic regression, SVMs
    - Regression: linear regression
  - Unsupervised learning (e.g., clustering)
  - Reinforcement learning (e.g., autonomous driving)
- Deep learning is “just” neural networks with many layers (hence deep)