

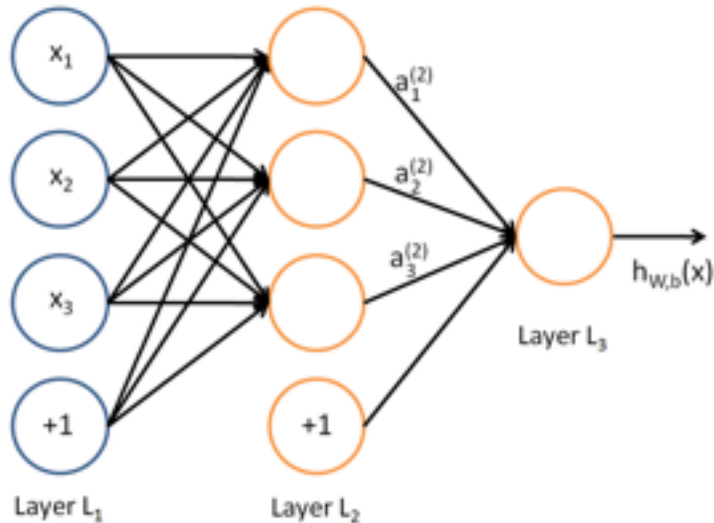
# A Brief Intro to Deep Learning

Tong He

Simon Fraser University

# What is Deep Learning?

- A family of machine learning models
  - Mainly neural networks
  - We only introduce a basic version
- Getting popular for the **extraordinary achievements**



# Extraordinary Achievements

- Image Understanding
- Self-Driving Car
- AlphaGo

# Image Understanding

Describes without errors	Describes with minor errors	Somewhat related to the image	Unrelated to the image
 <p>A person riding a motorcycle on a dirt road.</p>	 <p>Two dogs play in the grass.</p>	 <p>A skateboarder does a trick on a ramp.</p>	 <p>A dog is jumping to catch a frisbee.</p>
 <p>A group of young people playing a game of frisbee.</p>	 <p>Two hockey players are fighting over the puck.</p>	 <p>A little girl in a pink hat is blowing bubbles.</p>	 <p>A refrigerator filled with lots of food and drinks.</p>
 <p>A herd of elephants walking across a dry grass field.</p>	 <p>A close up of a cat laying on a couch.</p>	 <p>A red motorcycle parked on the side of the road.</p>	 <p>A yellow school bus parked in a parking lot.</p>

# Self-Driving Car

- Environment Recognition
  - Traffic Signals
  - Cars/Pedestrians
  - Cross road
- Real-time decision
- Car operation



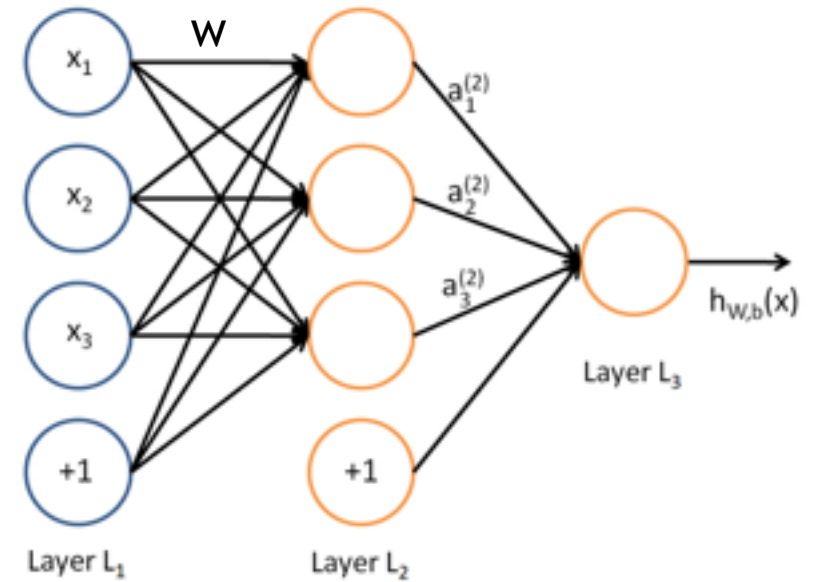
# AlphaGo

- The game of Go
- AlphaGo vs Lee Sedol



# A simple *shallow learning* model

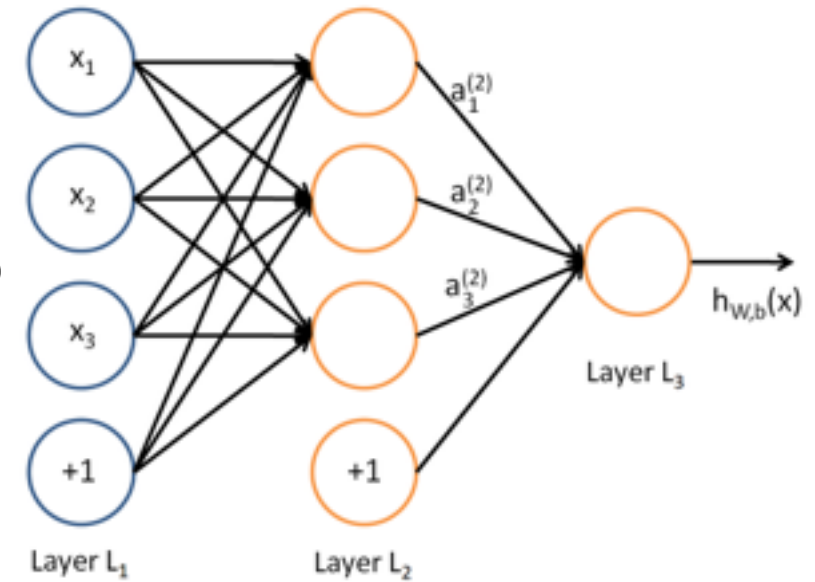
- A neural network
- One hidden layer
- Input  $\rightarrow$  hidden layer  $\rightarrow$  output
- Projection:  $a_j = f(\sum w_{ij}x_i)$ 
  - It holds between layers





# Intuition

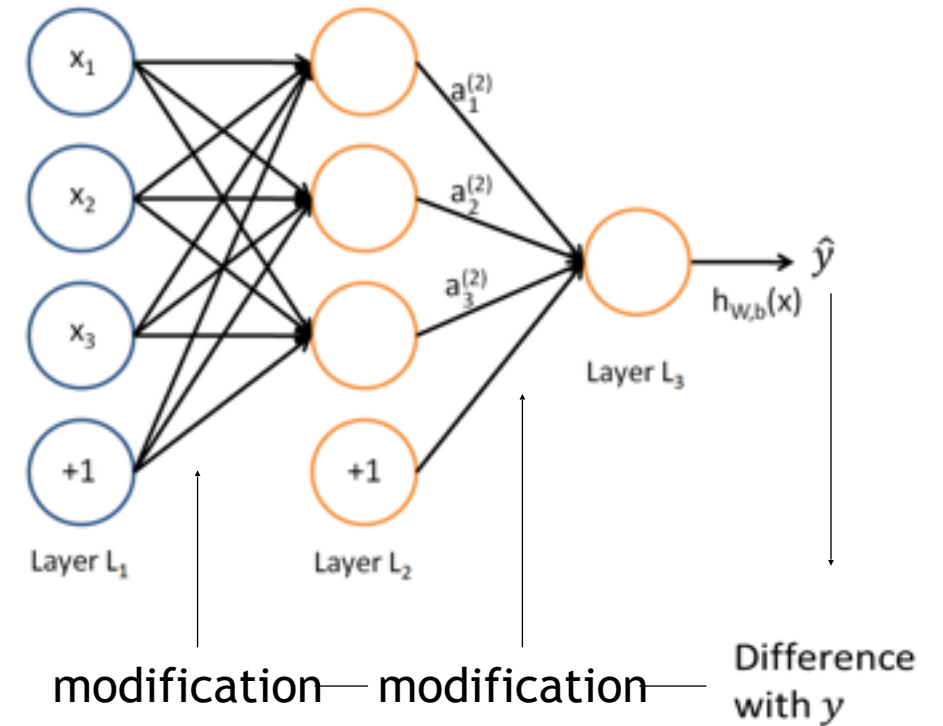
- Hidden layer: *an description*
  - Capture the non-linear patterns
- The output layer make better prediction
  - With a higher level of “understanding”





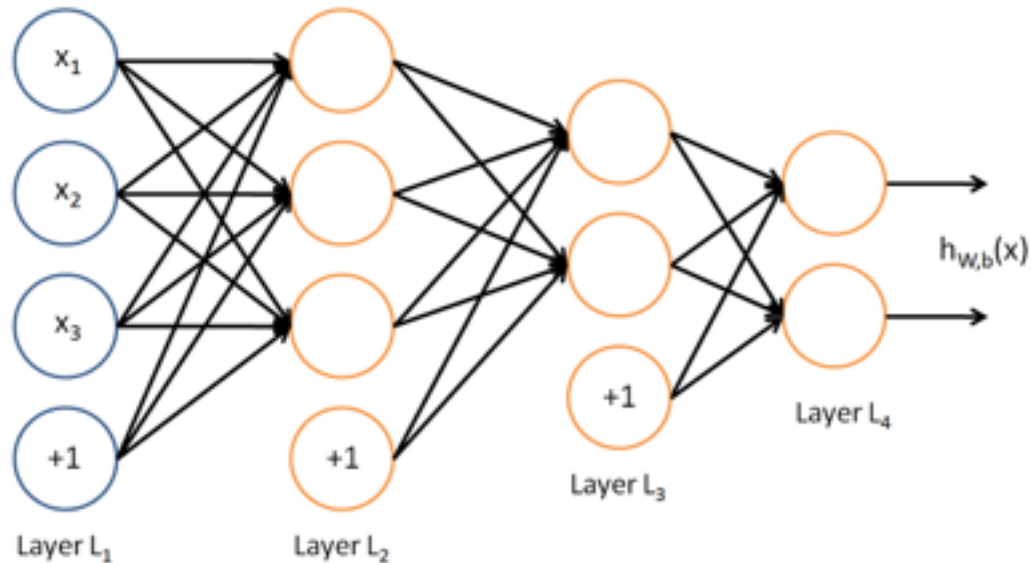
# Training: Backpropagation

- Prediction Error
  - How far are we from being perfect
- Modify hidden  $\rightarrow$  output
  - Take a step towards perfection
- Modify input  $\rightarrow$  hidden
  - Take another step towards perfection



# *A deep learning model*

- A neural network
- add *multiple* hidden layers?
- Input  $\rightarrow$  hidden layer  $\rightarrow \dots \rightarrow$  hidden layer  $\rightarrow$  output



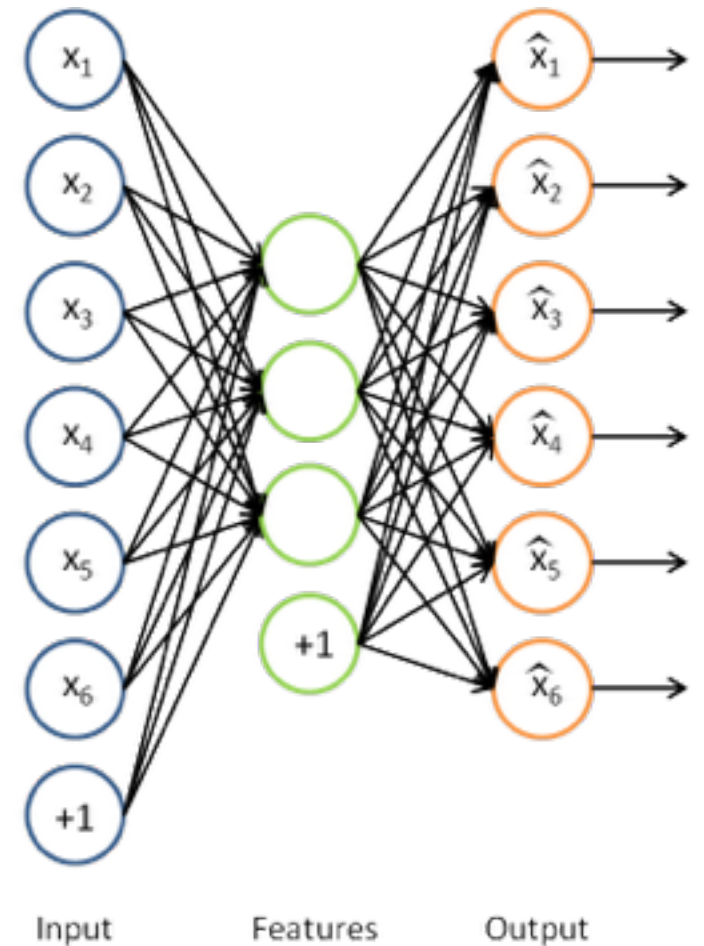
*“Houston, we have a problem”*

# Difficulties

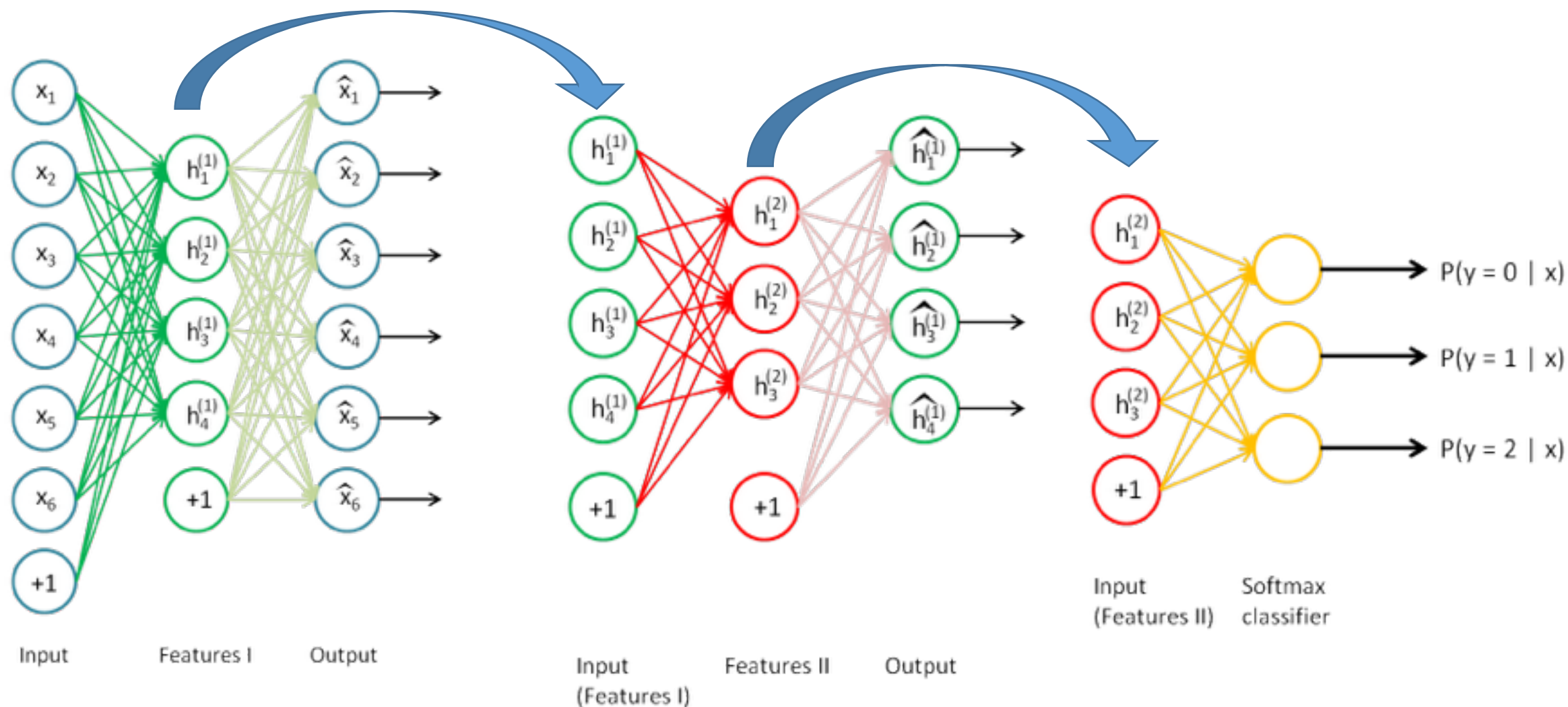
- Not enough data
- Local optima
- Diffusion of gradients

# Autoencoder

- Care more about the abstraction first
- A neural network to predict the input
  - Recover input from the hidden layer
  - High quality features

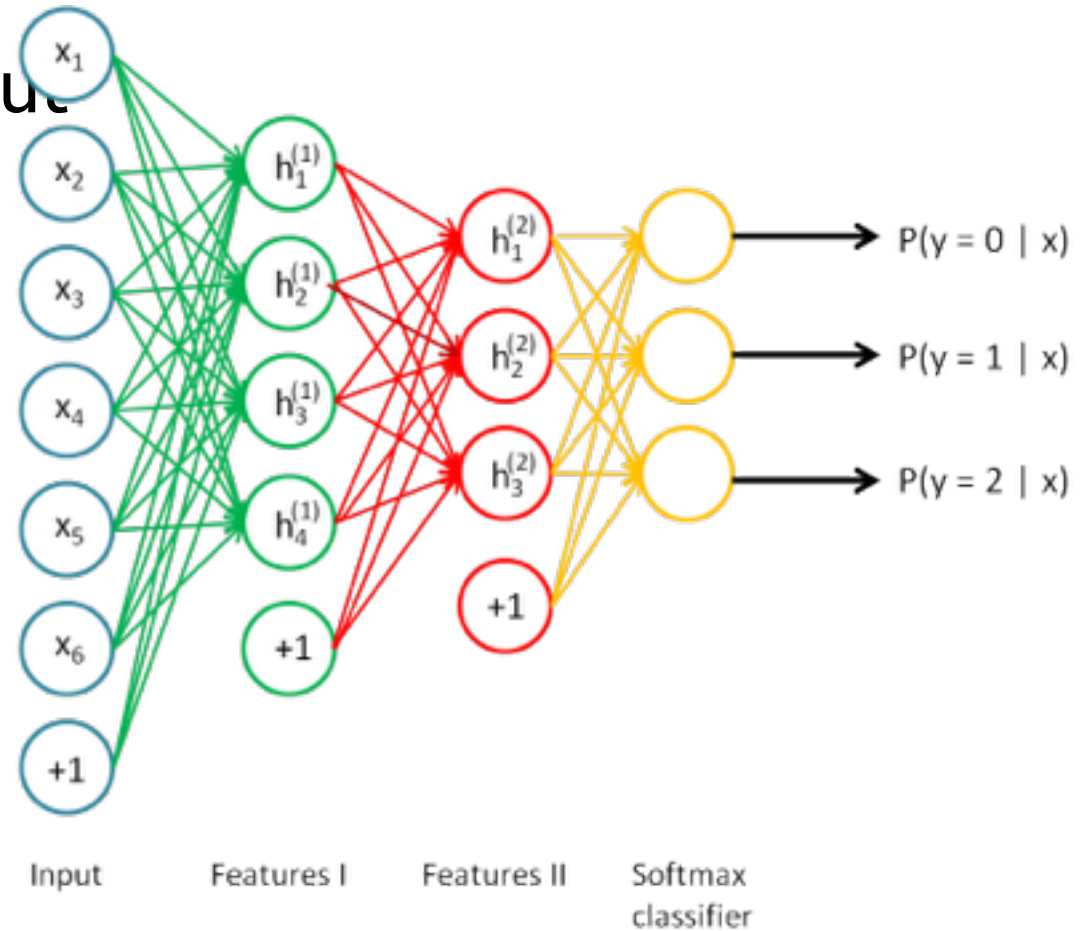


# Stacked Autoencoder



# Stacked Autoencoder

- Use the last hidden layer as input
- Fine tune at the last step



# Stacked Autoencoder

- Use the last hidden layer as input
  - Get higher abstraction
  - No need to have labeled data
- Avoid training entirely at first
  - Avoid diffusion of gradients
- Pre-train → Fine-tune
  - Find better local optima



# Other families

- CNN
- RNN
- LSTM

# R packages

- MXNet: <https://github.com/dmlc/mxnet>
- H2O: <http://www.h2o.ai/>

# Resources

- UFLDL: Unsupervised Feature Learning and Deep Learning
- Deeplearning.net
- Research papers

Q&A