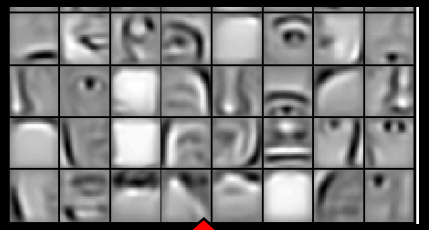
Some slides were adated/taken from various sources, including Andrew Ng's Coursera Lectures, CS231n: Convolutional Neural Networks for Visual Recognition lectures, Stanford University CS Waterloo Canada lectures, Aykut Erdem, et.al. tutorial on Deep Learning in Computer Vision, Ismini Lourentzou's lecture slide on "Introduction to Deep Learning", Ramprasaath's lecture slides, and many more. We thankfully acknowledge them. Students are requested to use this material for their study only and NOT to distribute it.

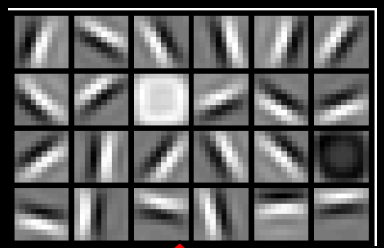
Learning feature hierarchies/Deep learning

Why feature hierarchies

object models

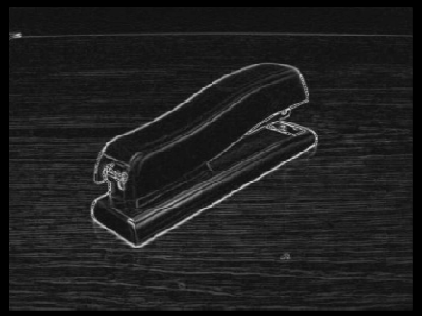


object parts
(combination
of edges)



edges

pixels



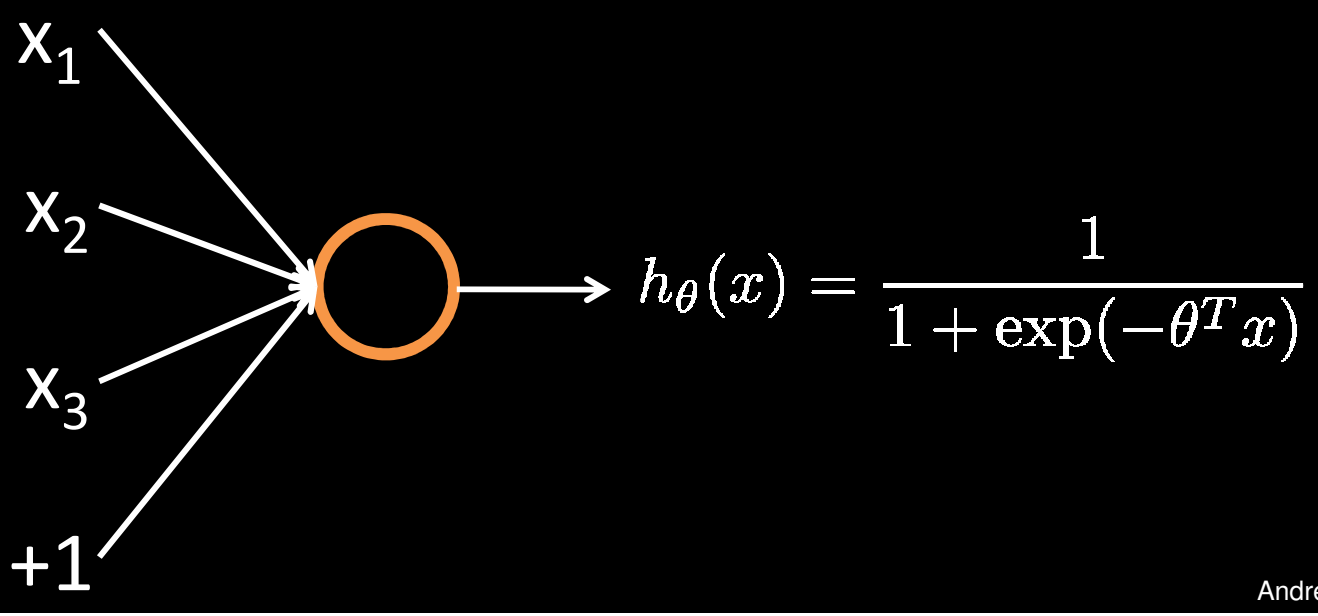
Logistic regression

Logistic regression has a learned parameter vector θ.
On input x, it outputs:

$$h_\theta(x) = \sigma(\theta^T x) = \frac{1}{1+\exp(-\theta^T x)}$$

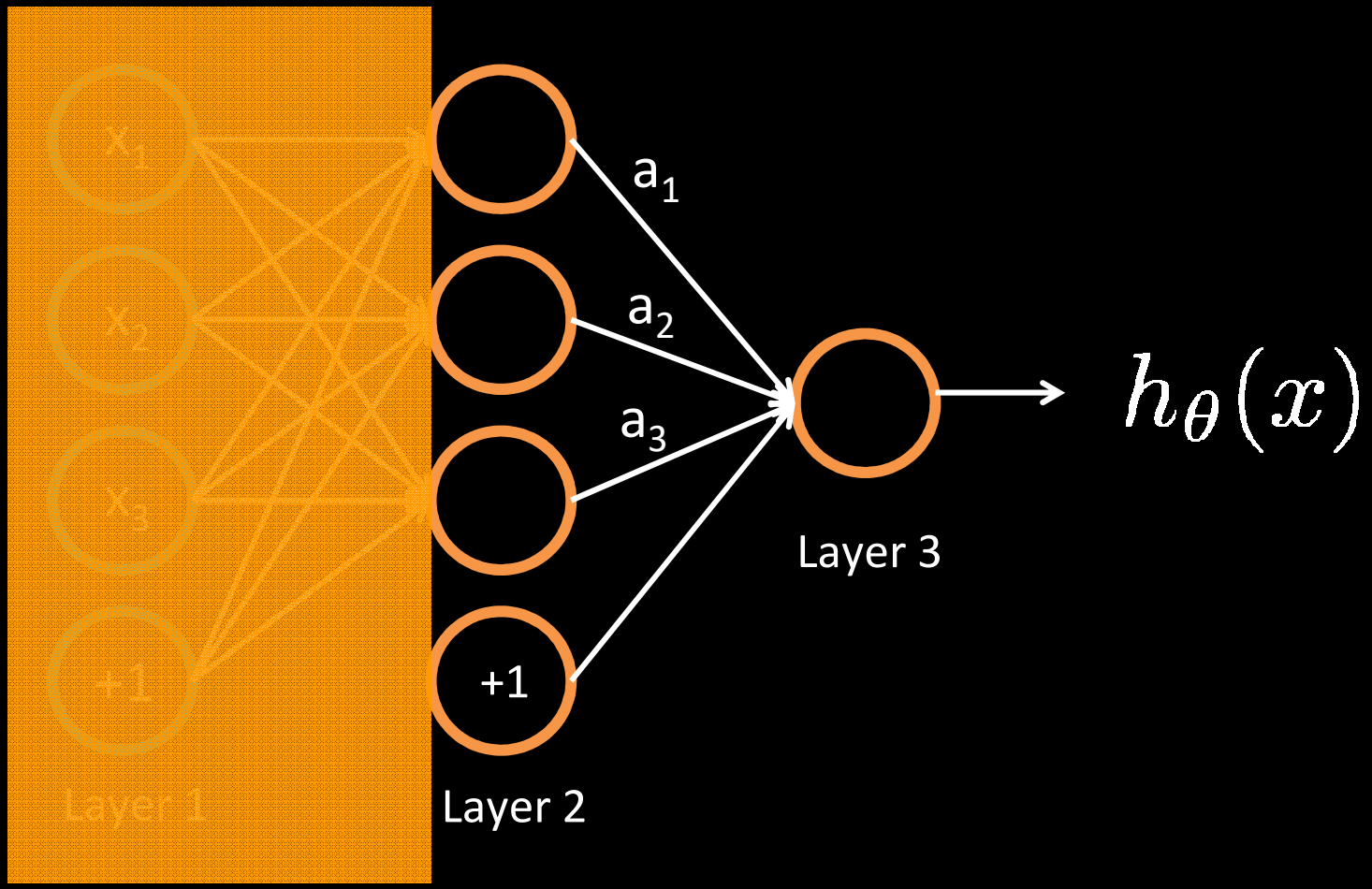
where $\sigma(z) = 1/(1+\exp(-z))$

Draw a logistic regression unit as:



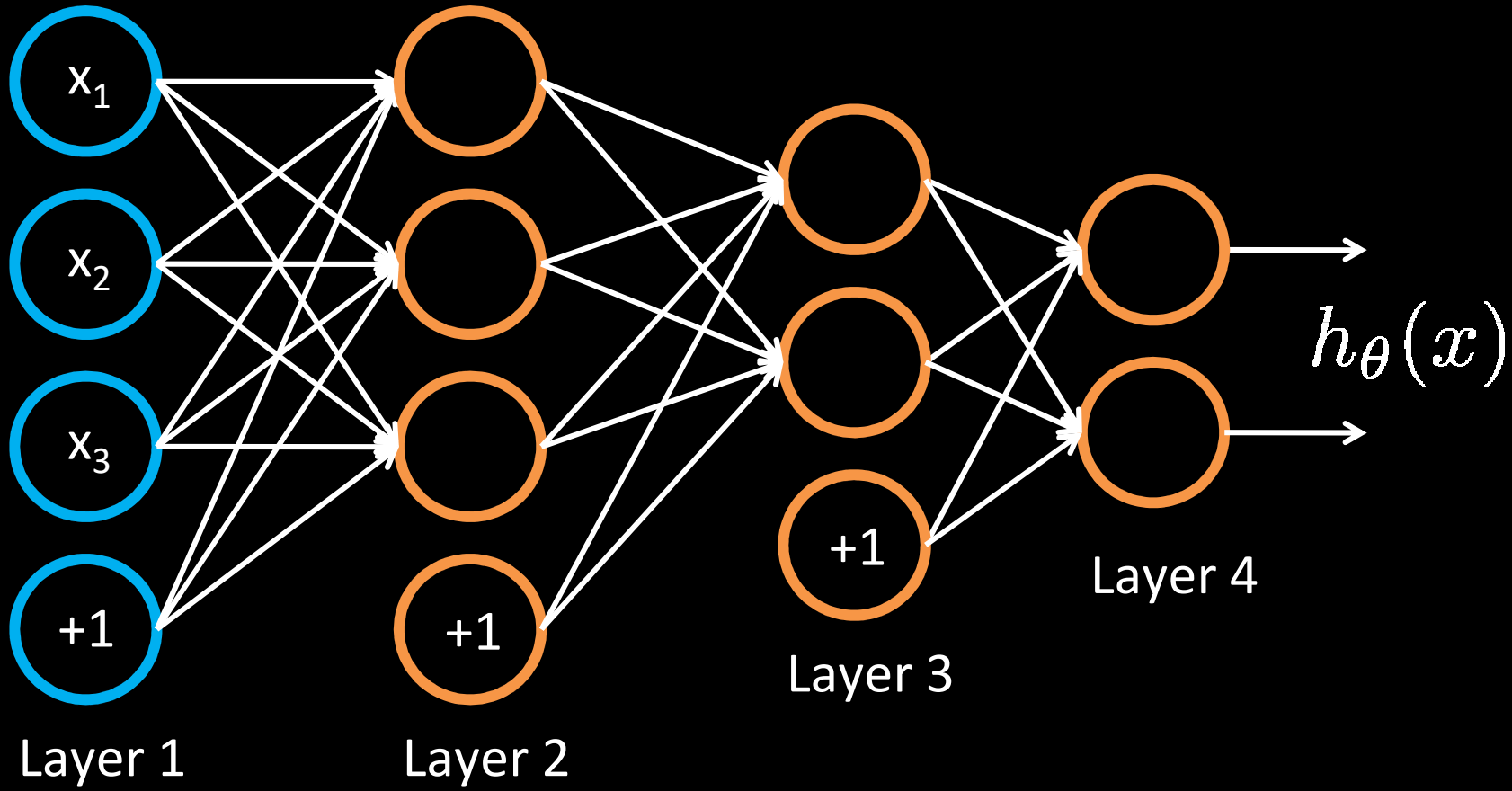
Neural Network

String a lot of logistic units together. Example 3 layer network:

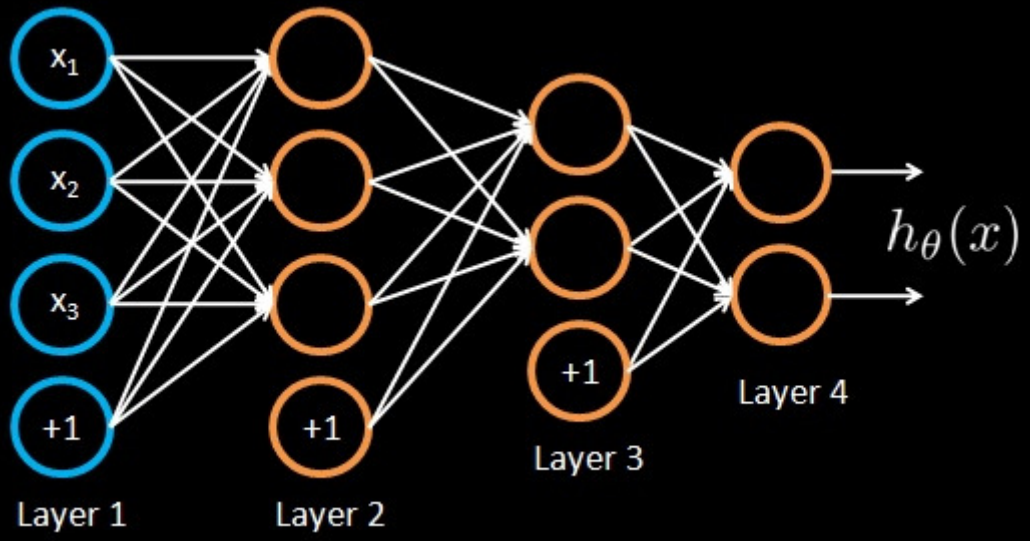


Neural Network

Example 4 layer network with 2 output units:



Training a neural network



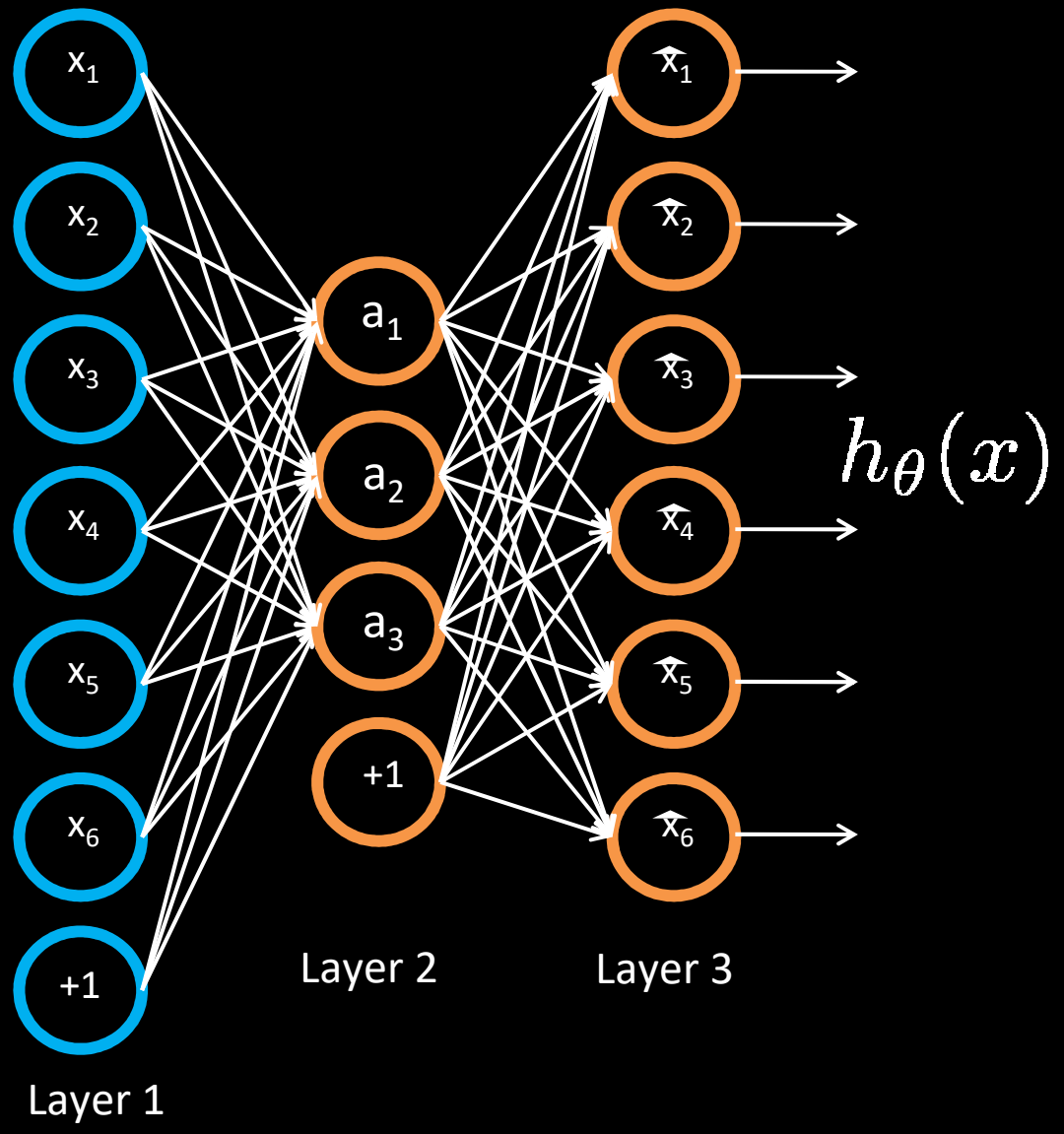
Given training set (x_1, y_1), (x_2, y_2), (x_3, y_3), ….

Adjust parameters θ (for every node) to make:

$$h_\theta(x_i) \approx y_i$$

(Use gradient descent. “Backpropagation” algorithm. Susceptible to local optima.)

Unsupervised feature learning with a neural network



Autoencoder.

Network is trained to output the input (learn identify function).

$$h_\theta(x) \approx x$$

Trivial solution unless:
- Constrain number of units in Layer 2 (learn compressed representation), or
- Constrain Layer 2 to be **sparse**.

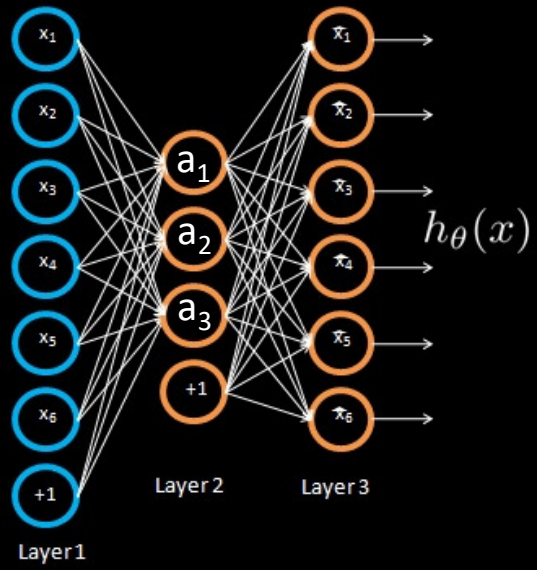
Andrew Ng

Unsupervised feature learning with a neural network

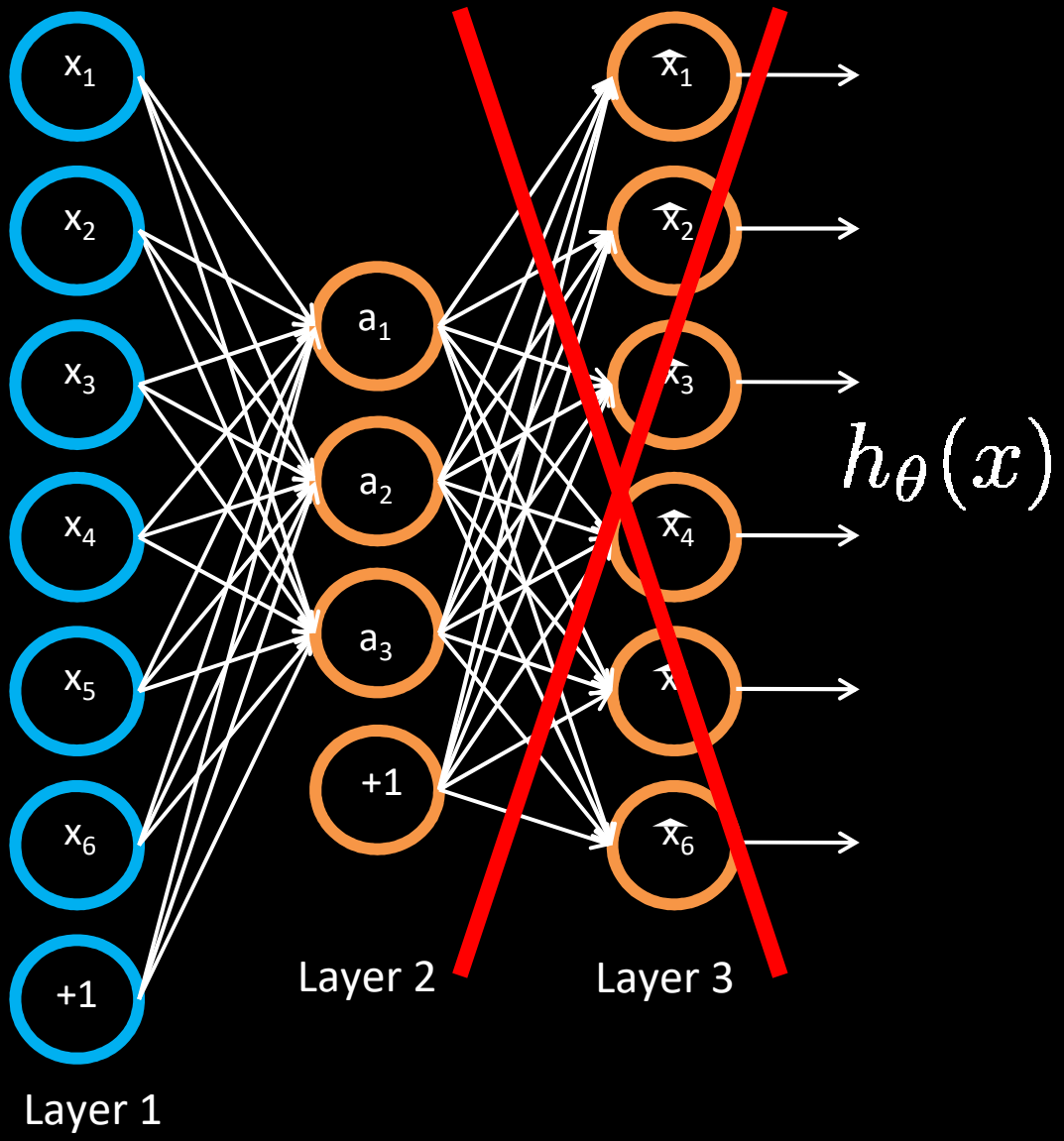
Training a sparse autoencoder.

Given unlabeled training set $x_1, x_2, \ldots$

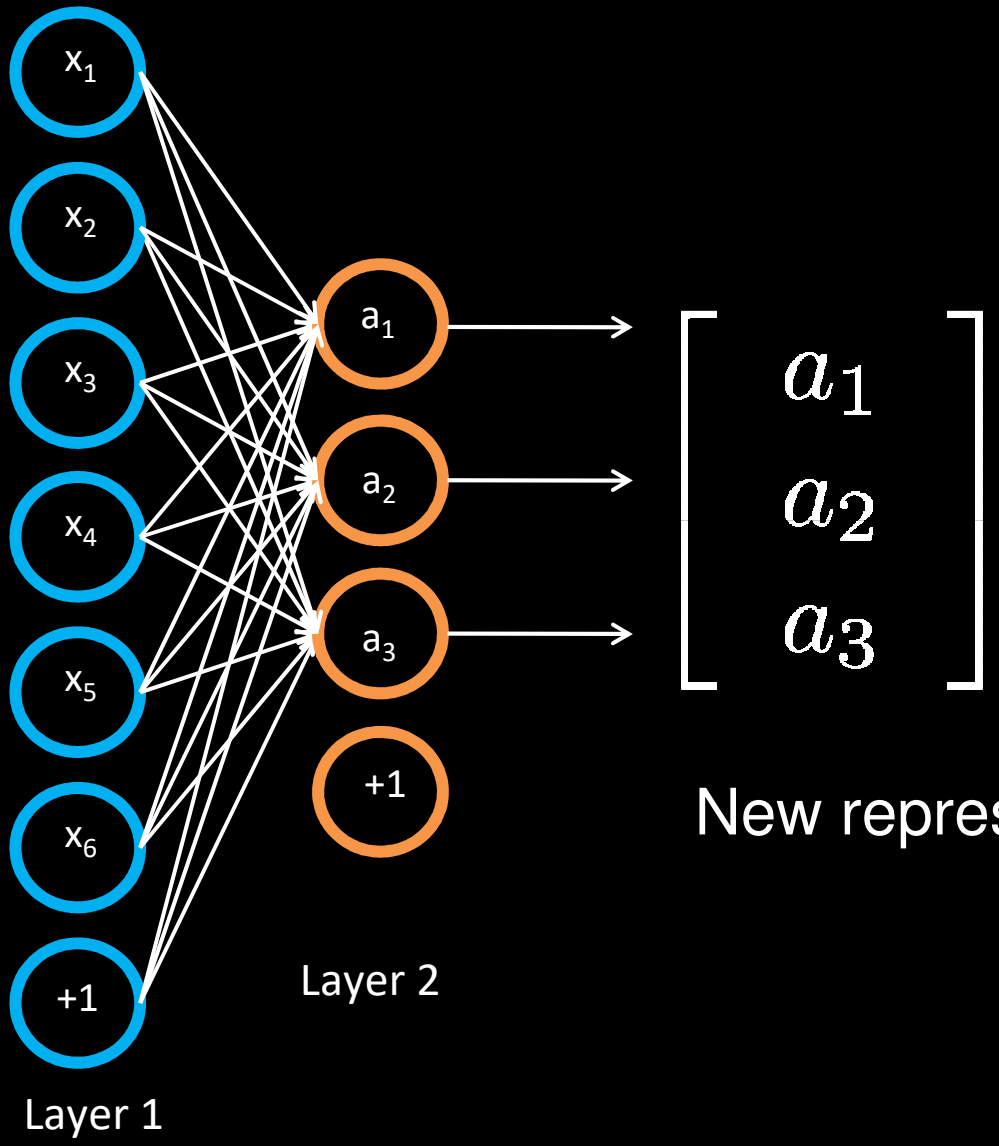
$$\min_{\theta} \underbrace{\|h_\theta(x) - x\|^2}_{\text{Reconstruction error term}} + \lambda \underbrace{\sum_i |a_i|}_{L_1 \text{ sparsity term}}$$



Unsupervised feature learning with a neural network

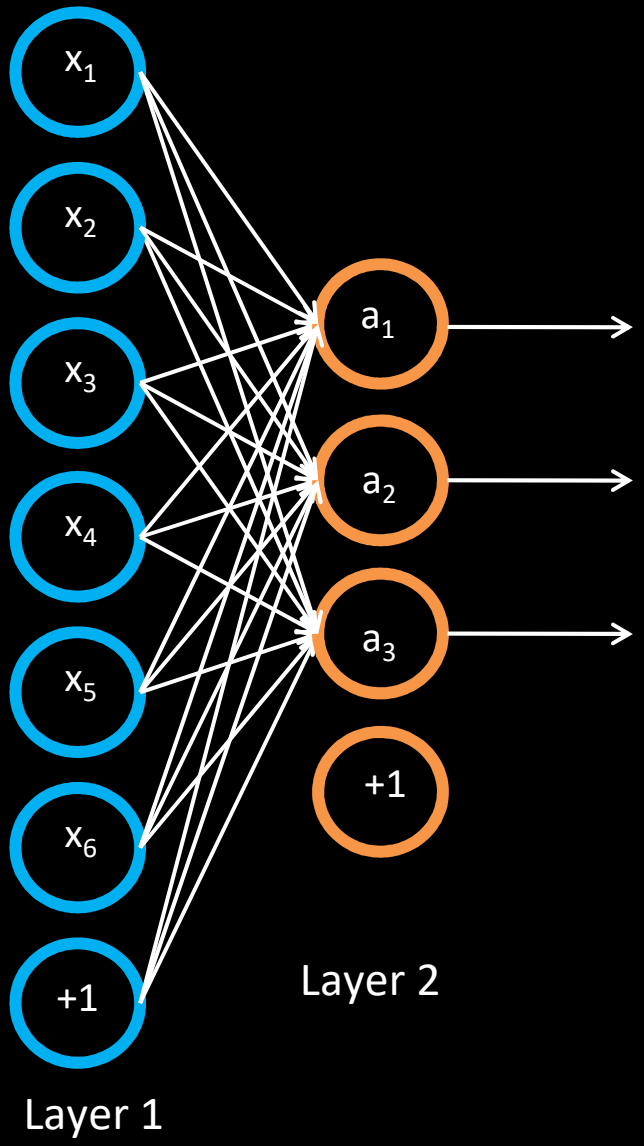


Unsupervised feature learning with a neural network

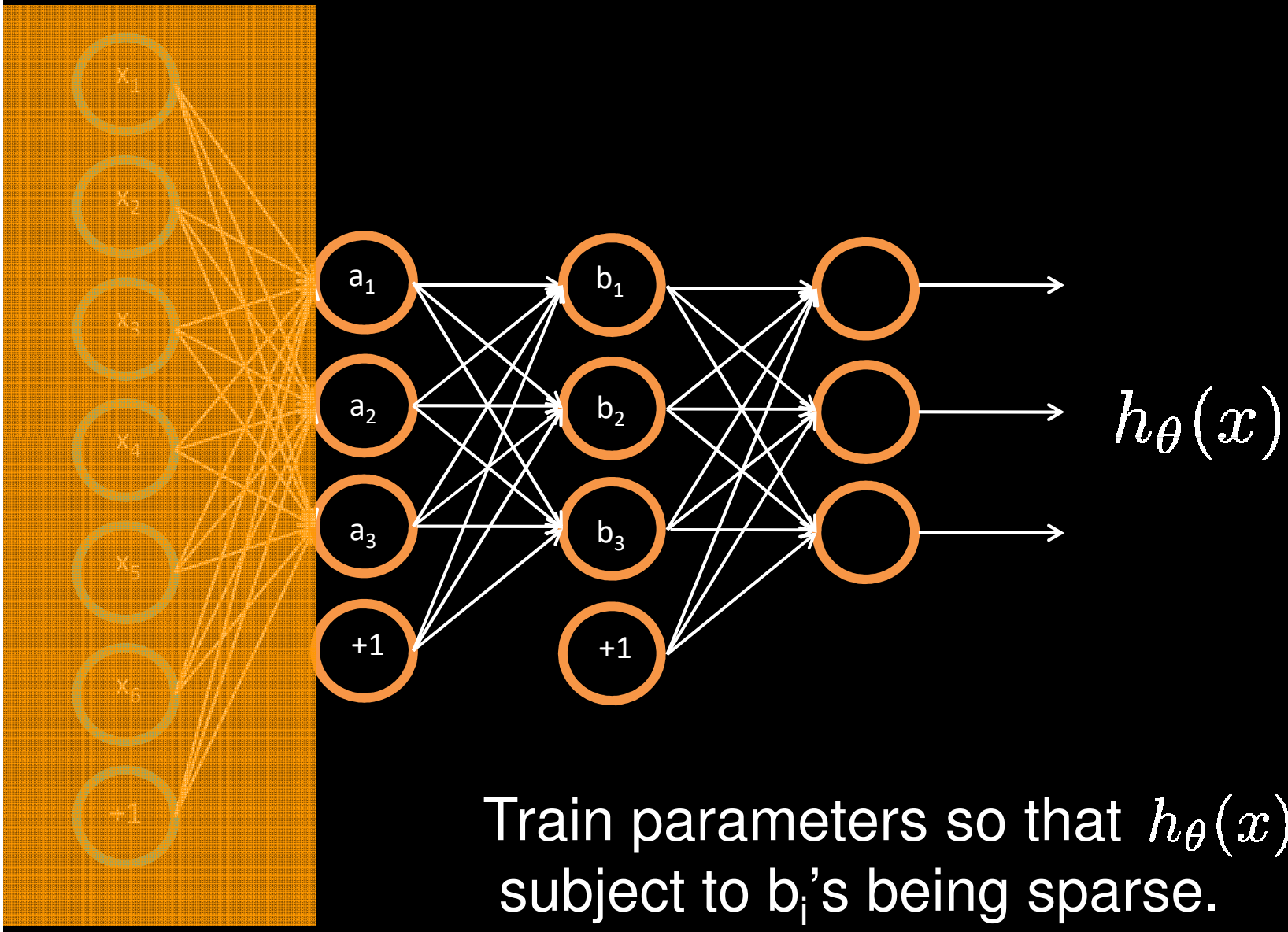


New representation for input.

Unsupervised feature learning with a neural network

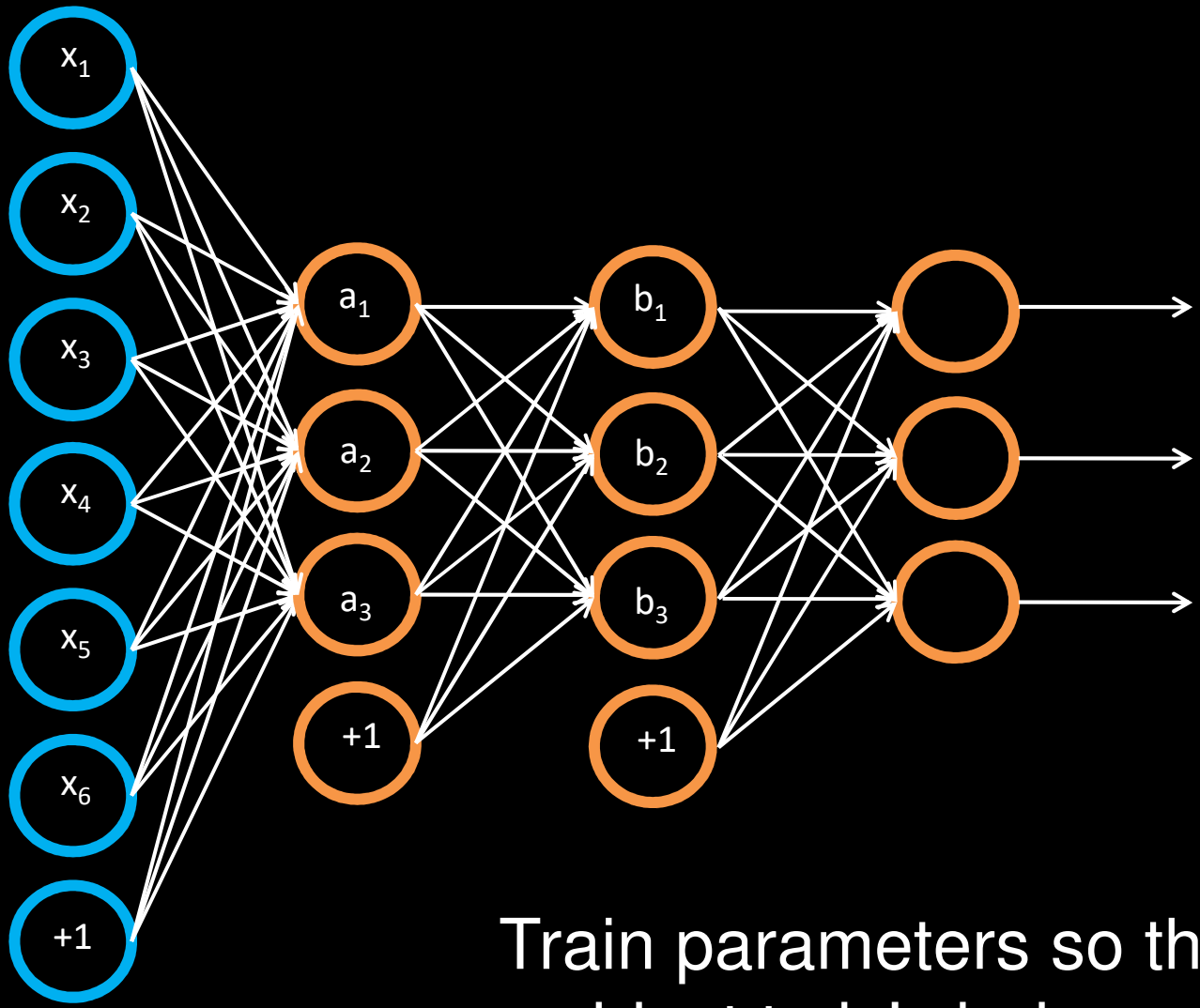


Unsupervised feature learning with a neural network



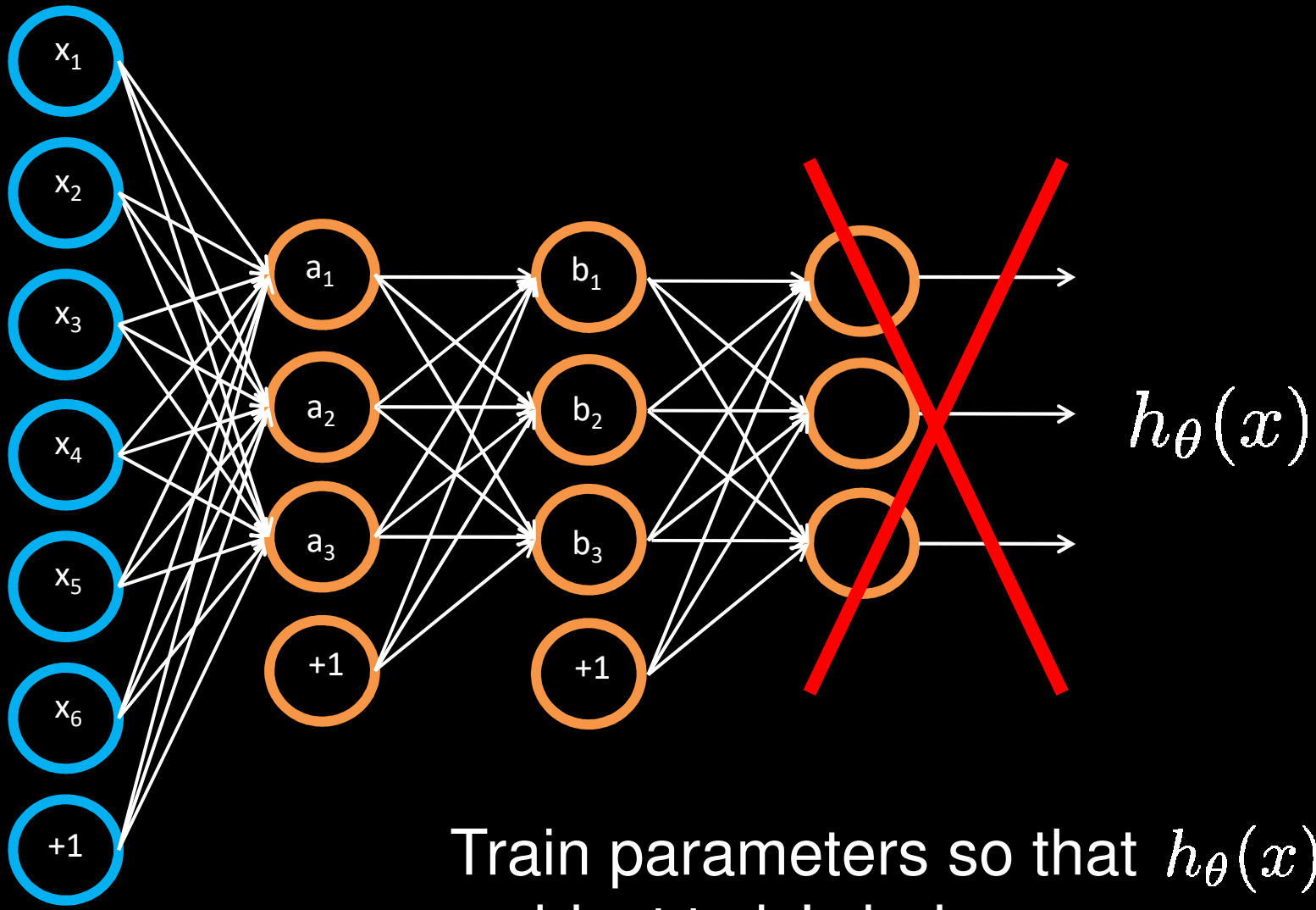
Train parameters so that $h_\theta(x) \approx a$, subject to b_i's being sparse.

Unsupervised feature learning with a neural network



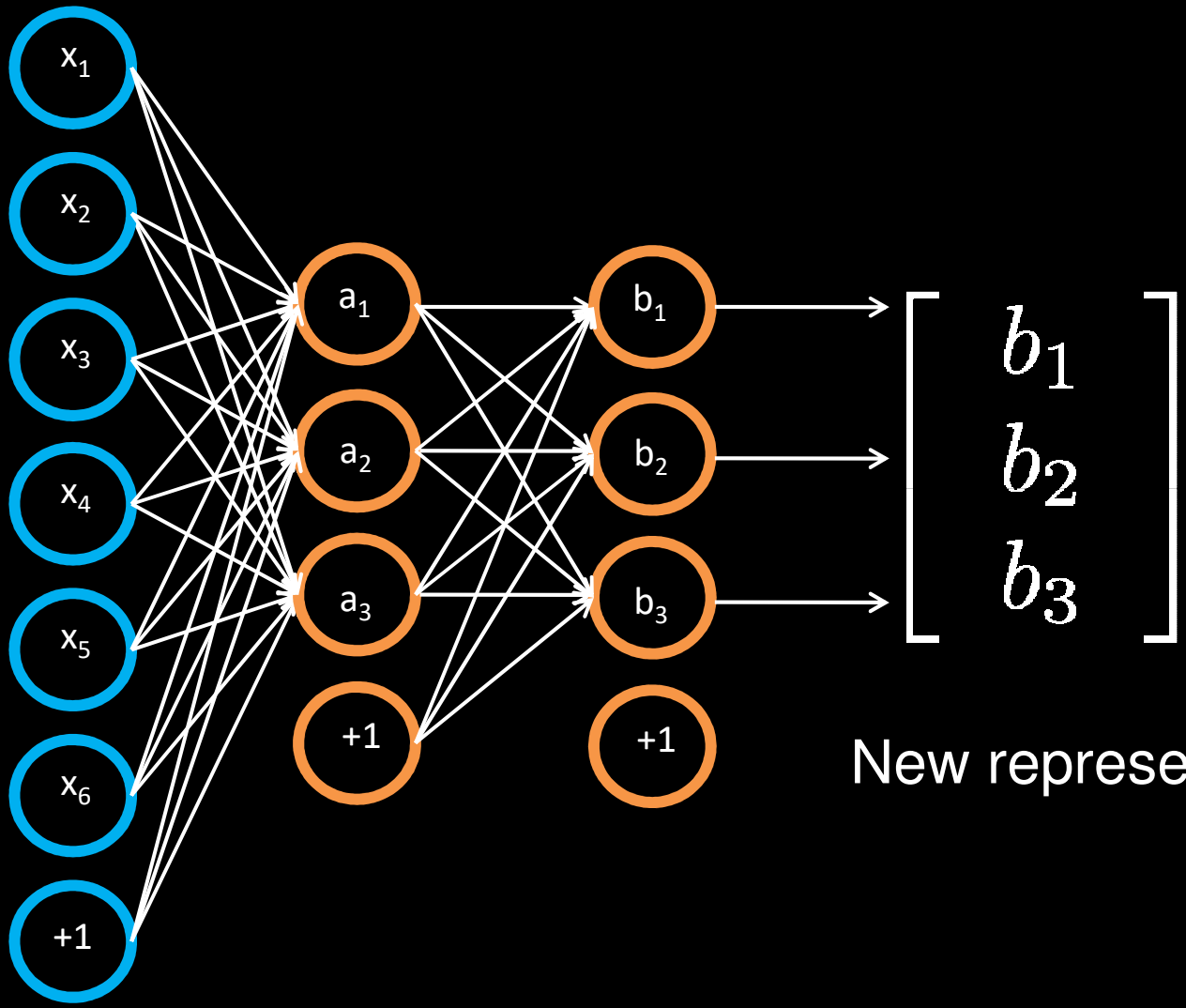
Train parameters so that $h_\theta(x) \approx a$, subject to b_i's being sparse.

Unsupervised feature learning with a neural network



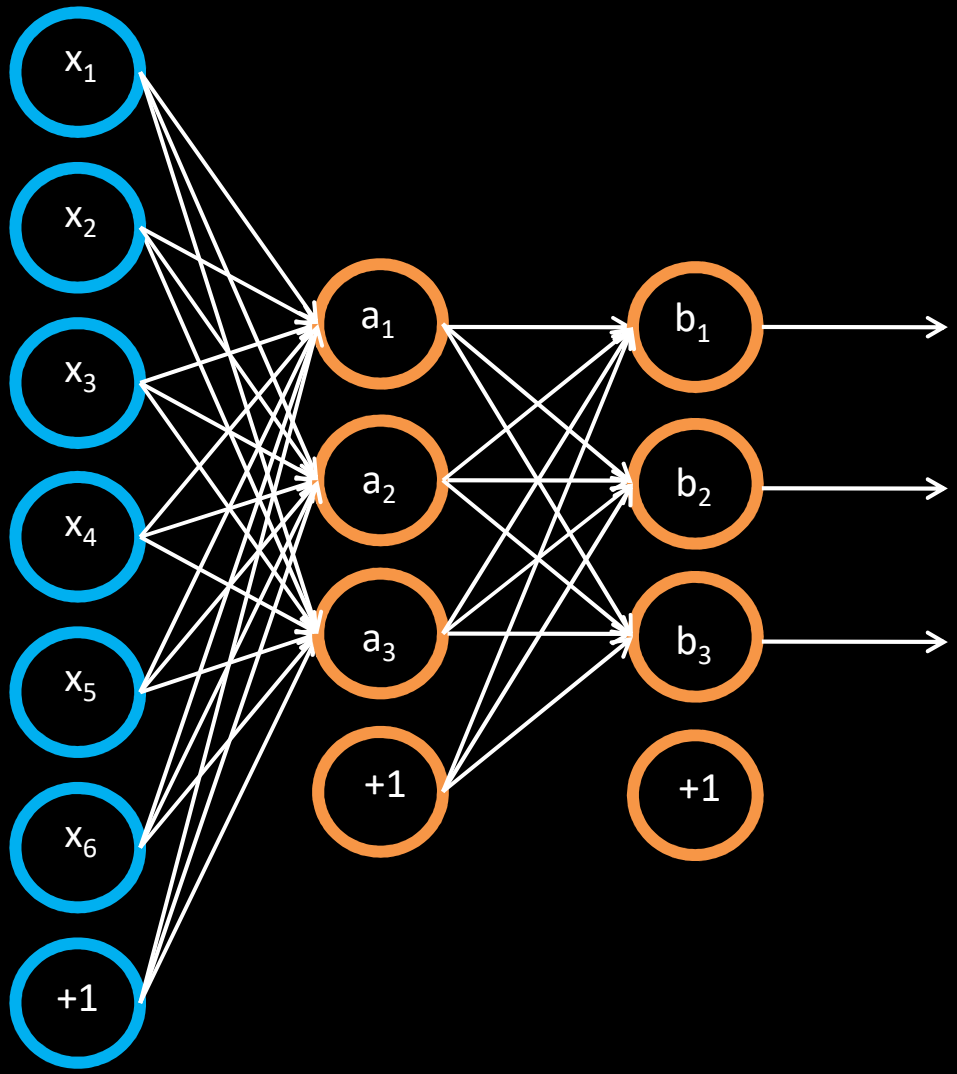
Train parameters so that $h_\theta(x) \approx a$, subject to b_i's being sparse.

Unsupervised feature learning with a neural network

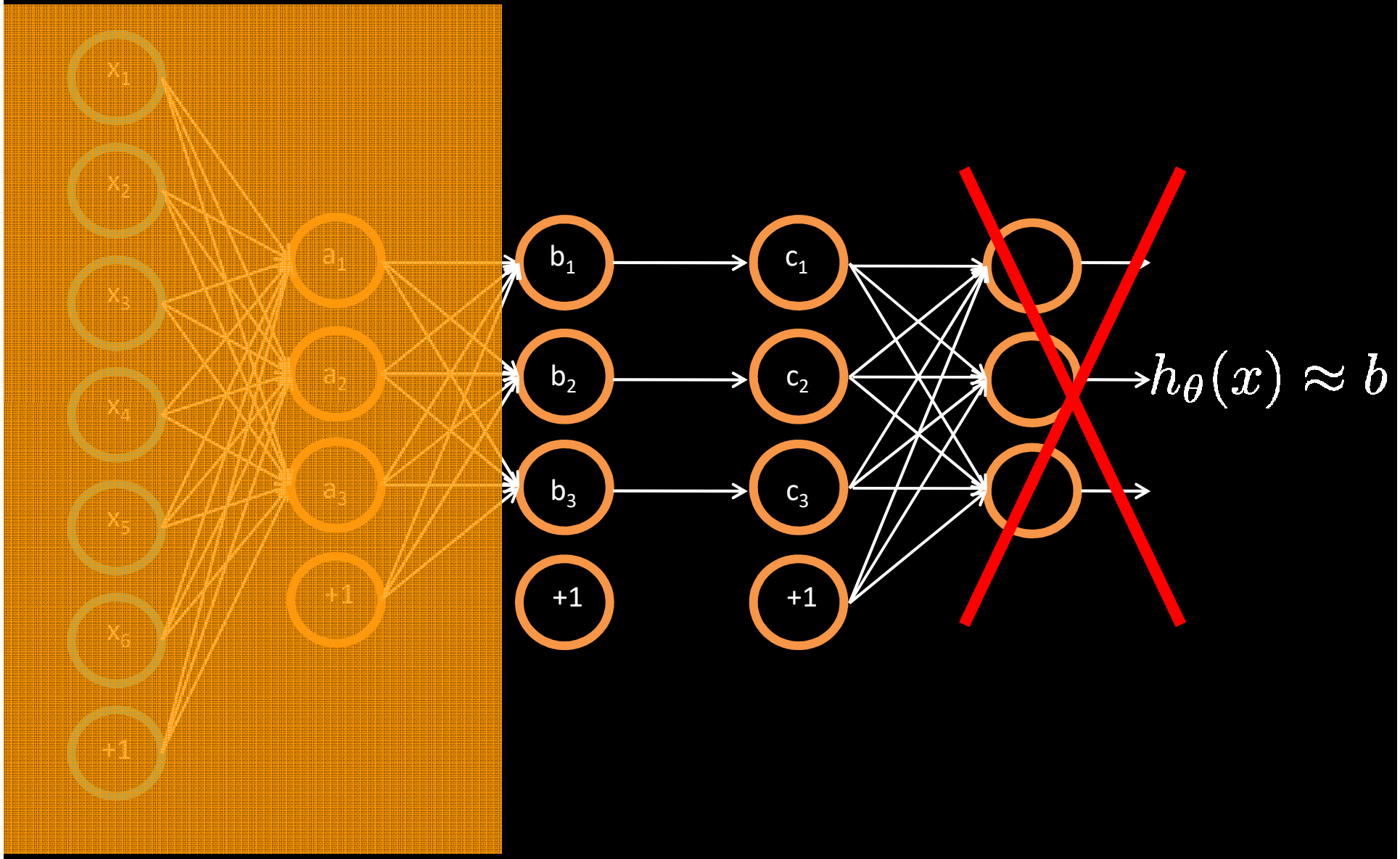


New representation for input.

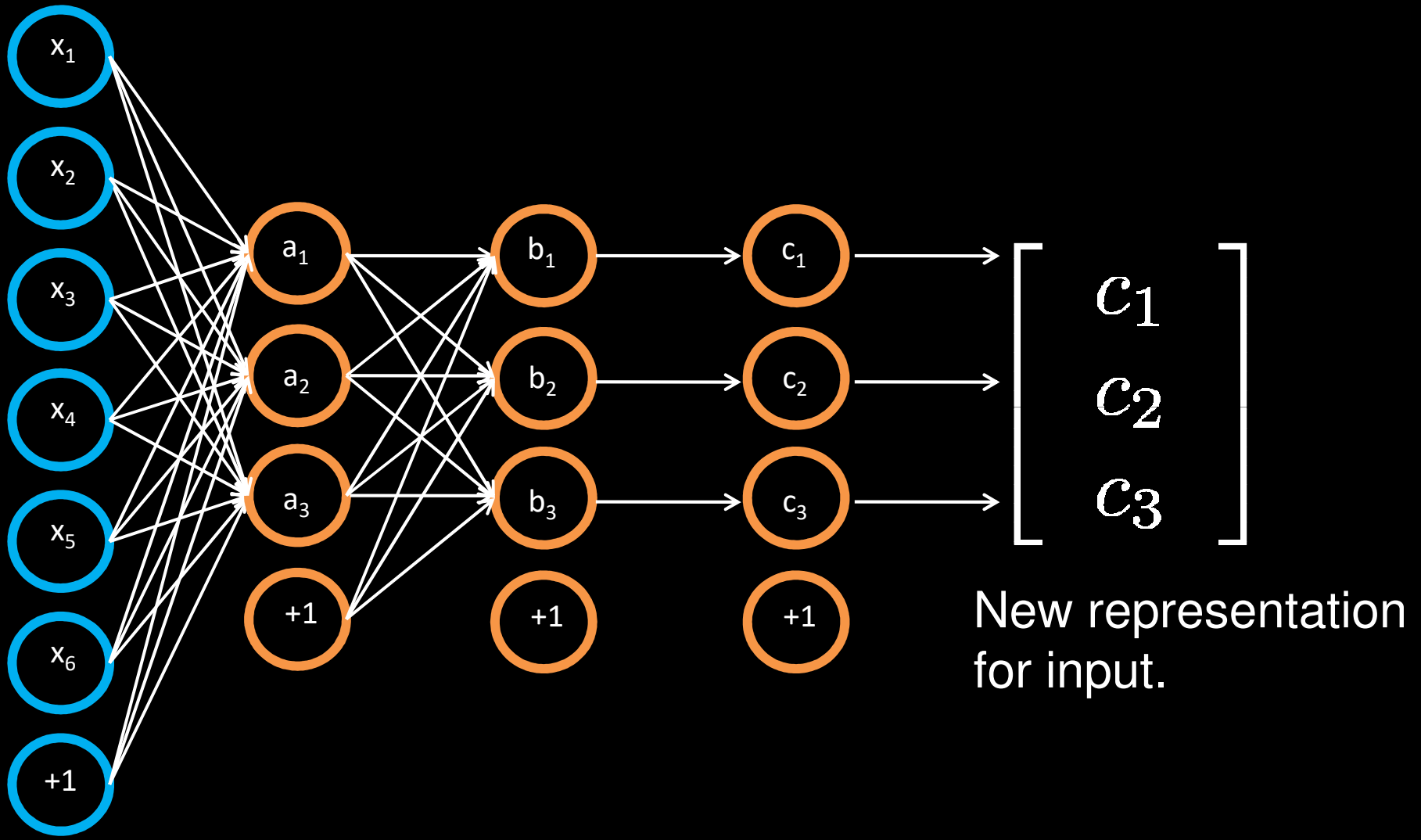
Unsupervised feature learning with a neural network



Unsupervised feature learning with a neural network



Unsupervised feature learning with a neural network



New representation for input.

Use $[c_1, c_3, c_3]$ as representation to feed to learning algorithm.

Andrew Ng