NEURAL NETWORKS

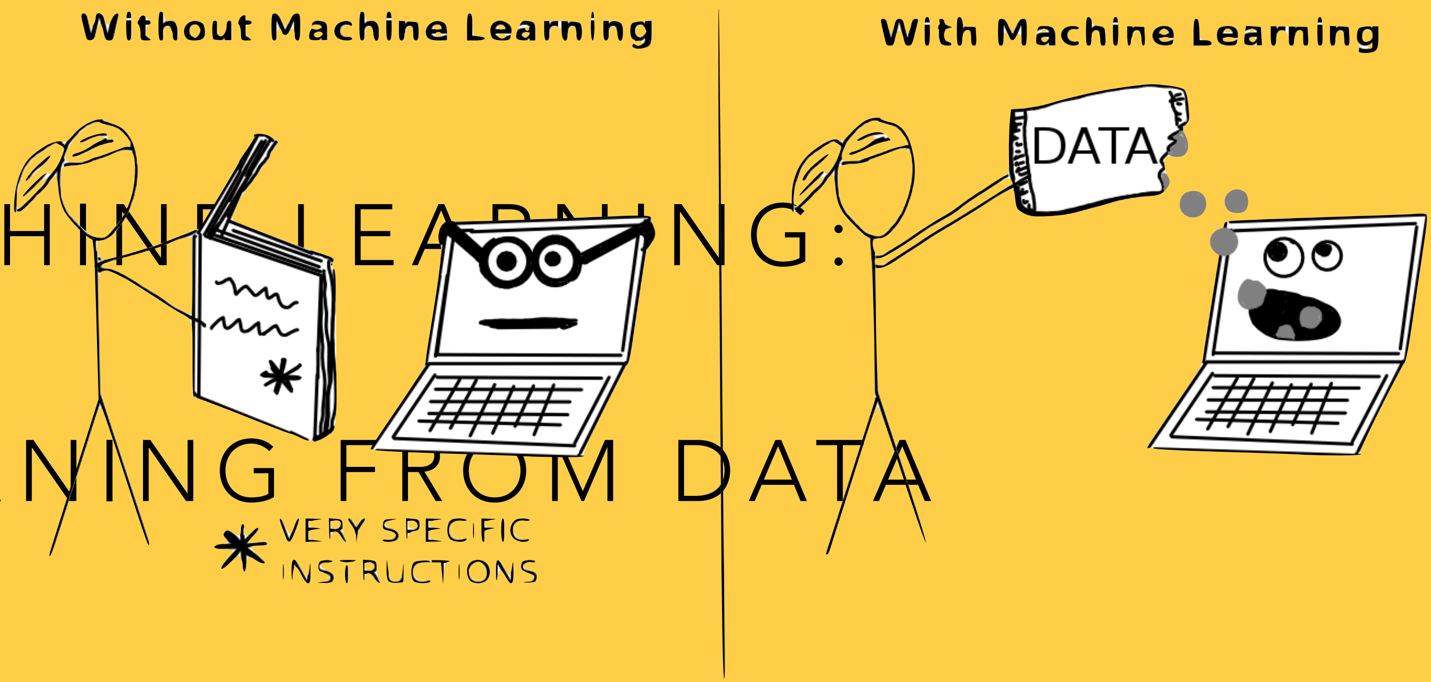
AND DEEP LEARNING

MICHELLE KUCHERA
DAVIDSON COLLEGE

AI4NP WINTER SCHOOL
11 JANUARY 2020

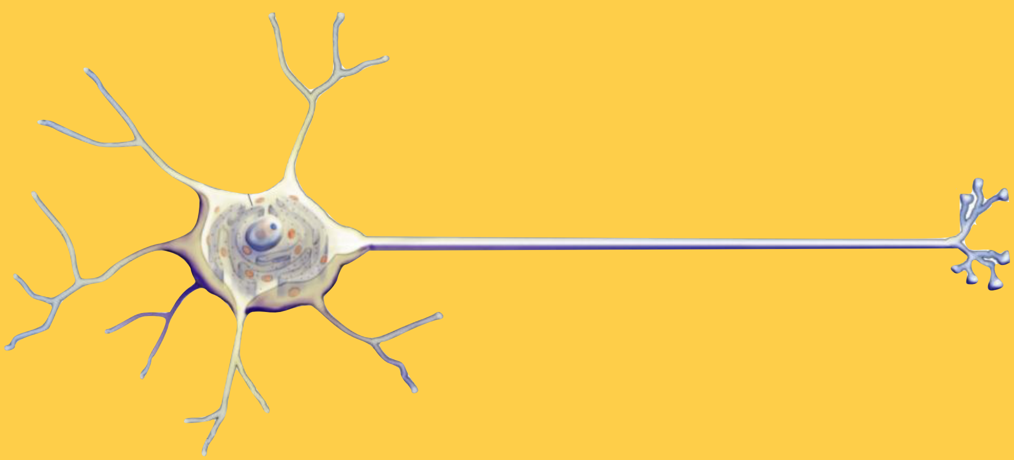
MACHINE LEARNING:

LEARNING FROM DATA




CAMERA_01
& Micromegas

NEURON



MATHEMATICS

ELSEVIER

Neural Networks
Volume 4, Issue 2, 1991, Pages 251-257

Approximation capabilities of multilayer feedforward networks

Kurt Hornik

Show more

Share Cite

https://doi.org/10.1016/0893-6080(91)90009-T

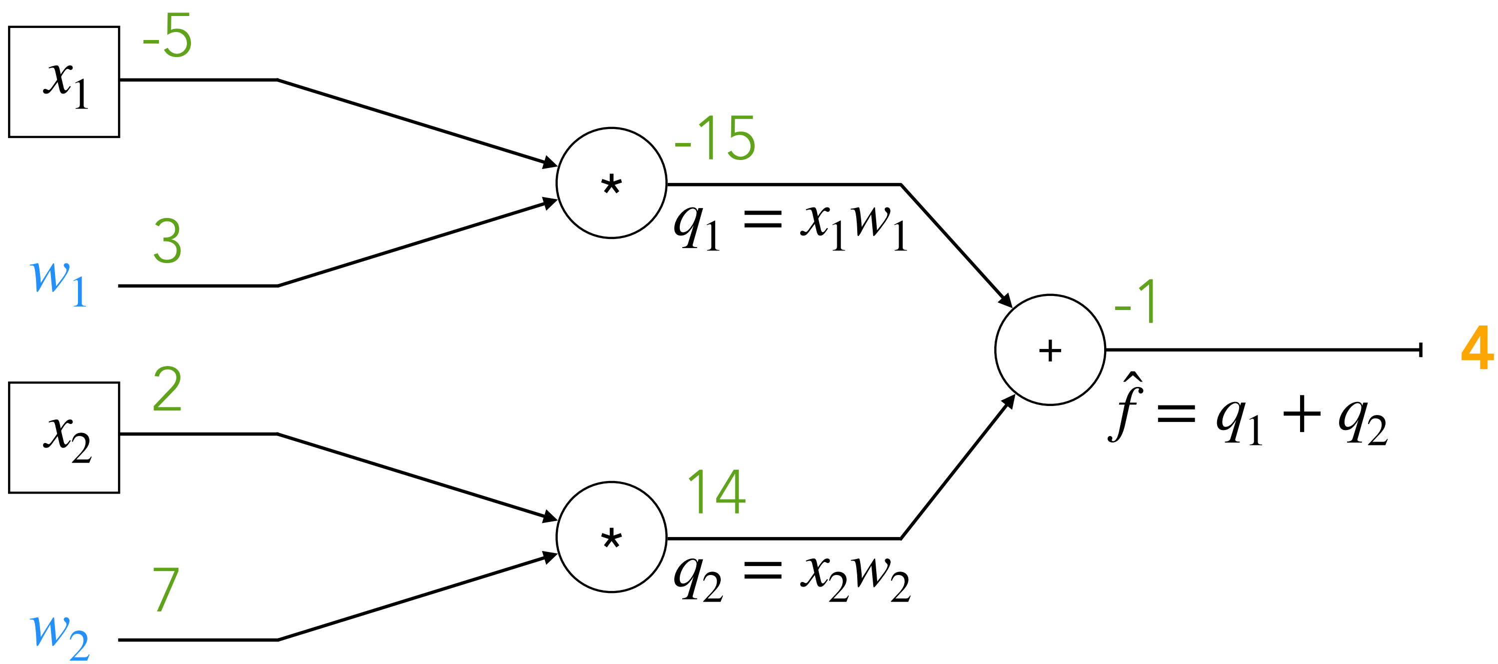
Get rights and content

Abstract

We show that standard multilayer feedforward networks with as few as a single hidden layer and arbitrary bounded and nonconstant activation function are universal approximators with respect to $L^p(\mu)$ performance criteria, for arbitrary finite input environment measures μ, provided only that sufficiently many hidden units are available. If the activation function is continuous, bounded and nonconstant, then continuous mappings can be learned uniformly over compact input sets. We also give very general conditions ensuring that networks with sufficiently smooth activation functions are capable of arbitrarily accurate approximation to a function and its derivatives.

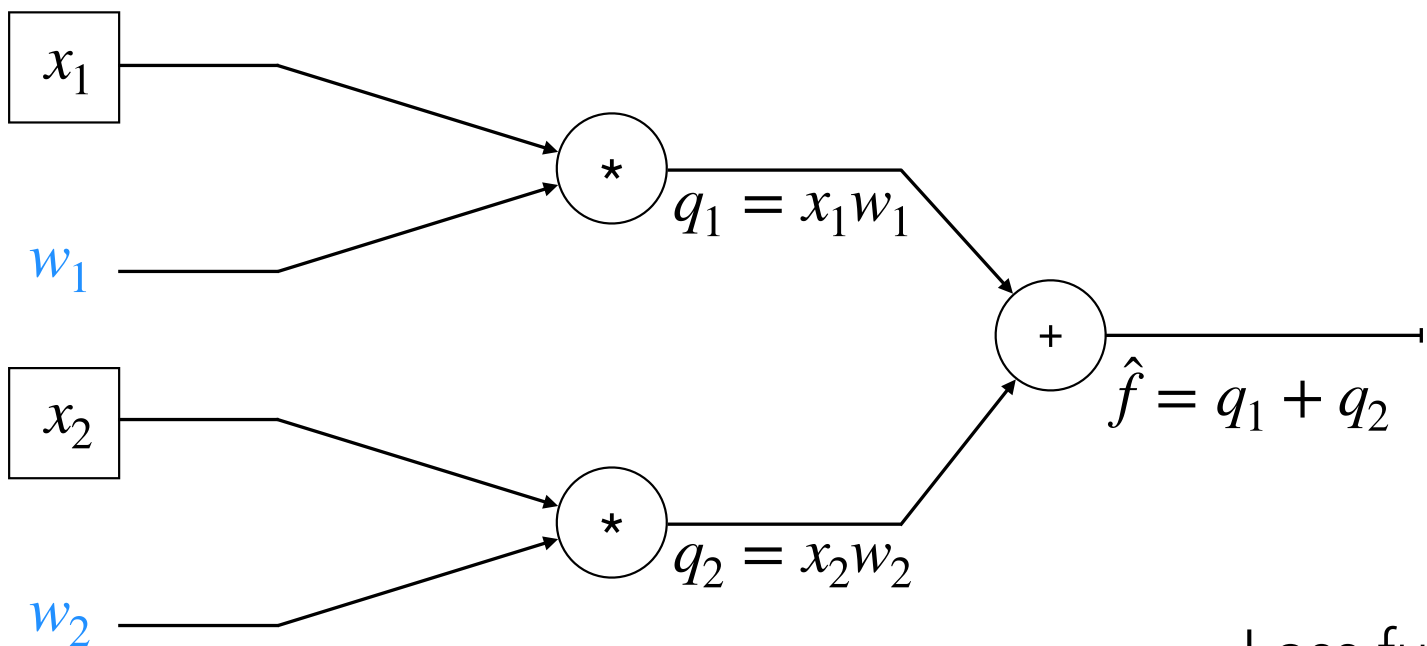
MATHEMATICS

COMPUTATIONAL GRAPH



$$\hat{f} = x_1 w_1 + x_2 w_2$$

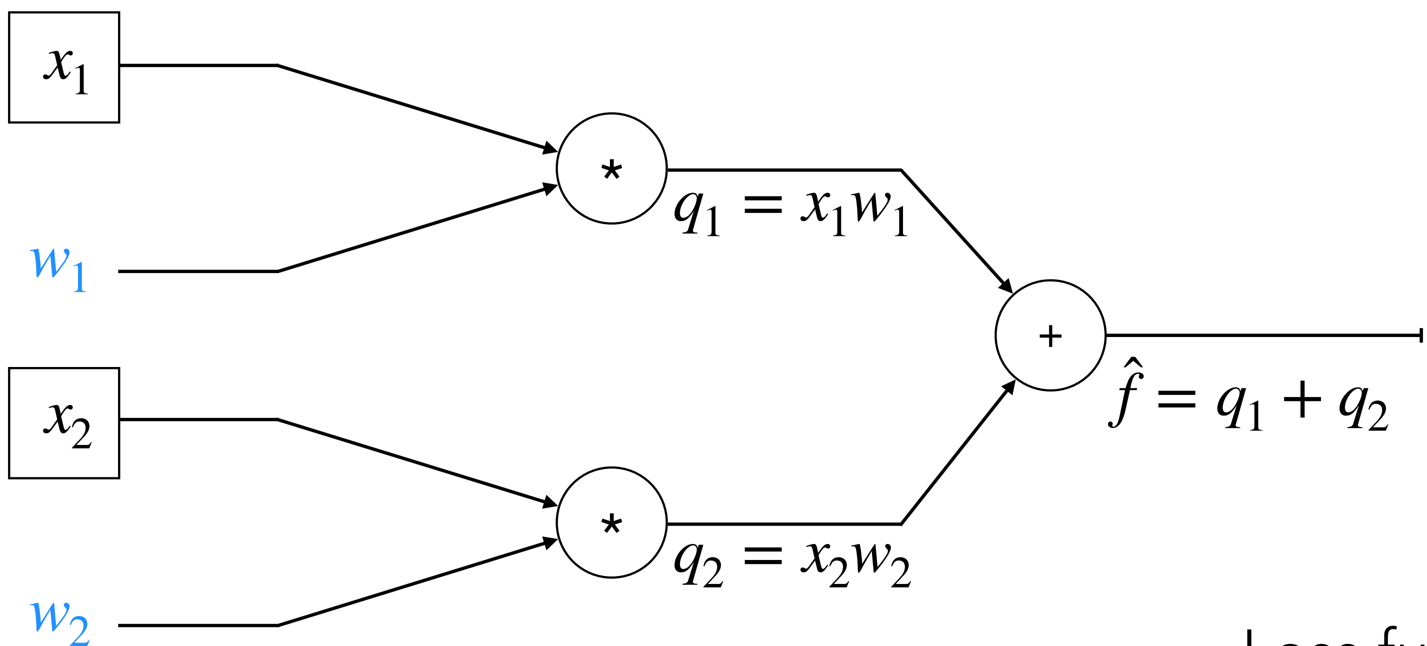
REGRESSION



Loss function

$$\hat{f} = x_1 w_1 + x_2 w_2 \qquad J(w) = f - \hat{f}$$

SUPERVISED LEARNING



Loss function

$\hat{f} = x_1 w_1 + x_2 w_2$

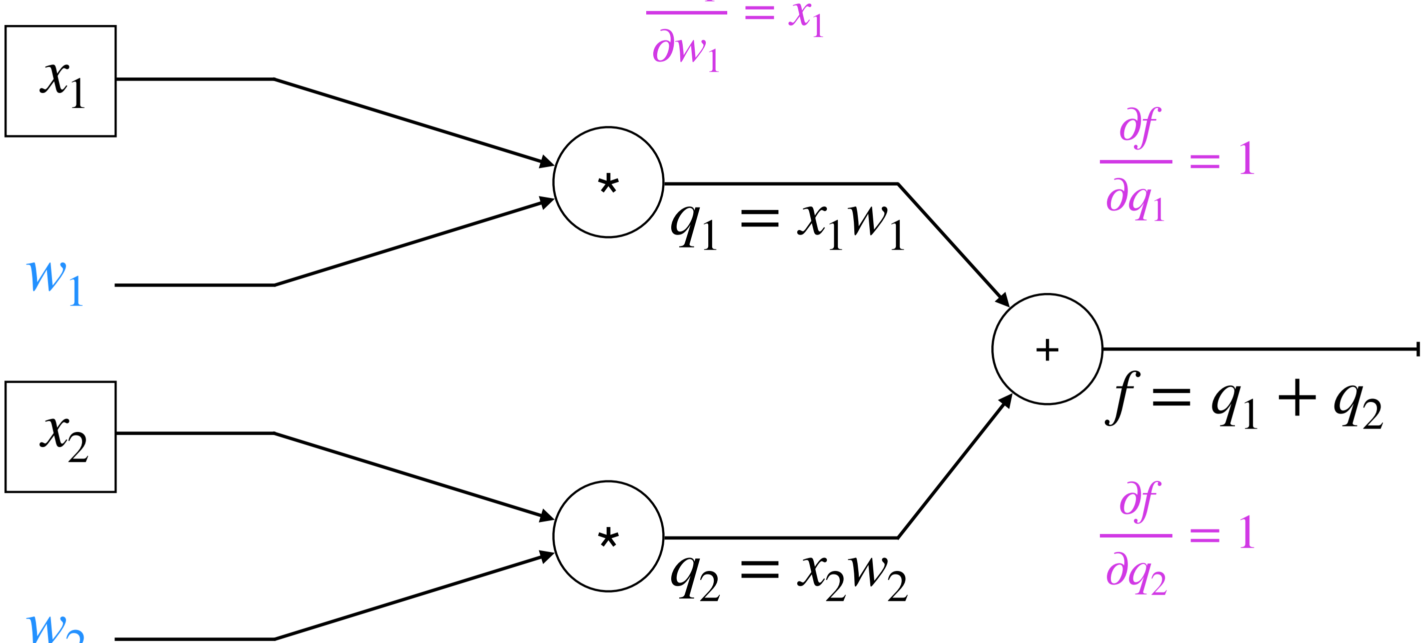
$J(w) = f - \hat{f}$

BACKPROPAGATION

$$w_1 = w_1 + \eta * \frac{\partial J}{\partial f}\frac{\partial f}{\partial q_1}\frac{\partial q_1}{\partial w_1}$$

$$\frac{\partial q_1}{\partial w_1} = x_1$$

$$\frac{\partial f}{\partial q_1} = 1$$



$$\frac{\partial J}{\partial f} = 1$$

$$\frac{\partial f}{\partial q_2} = 1$$

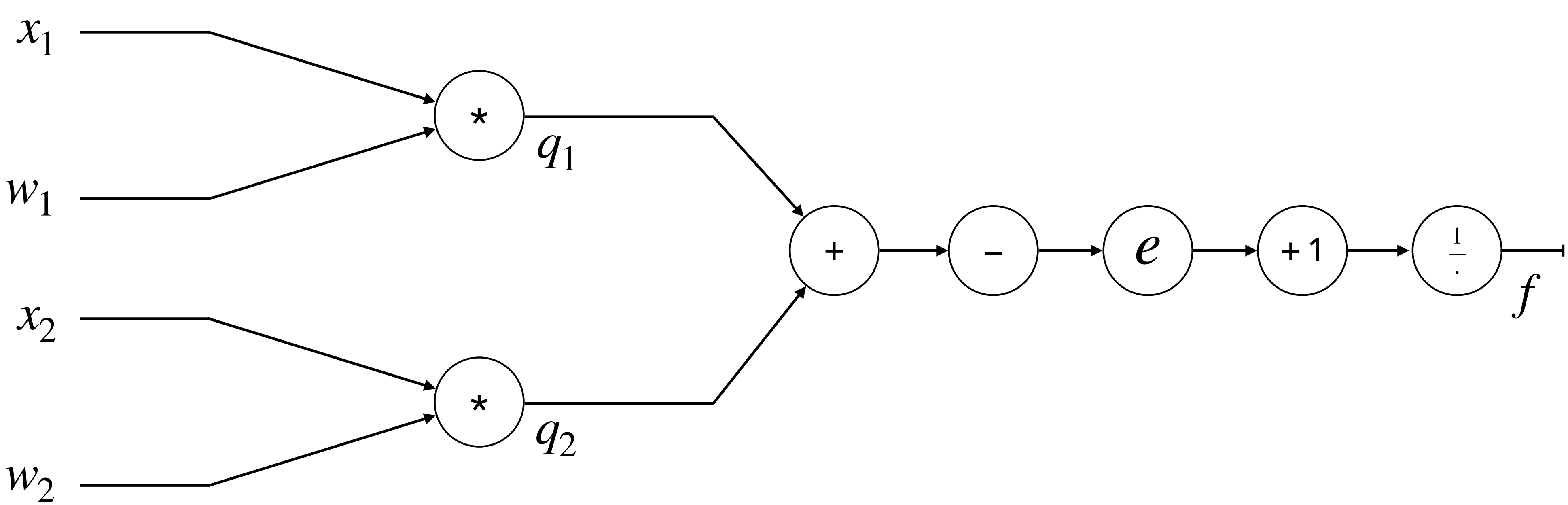
$$\frac{\partial q_2}{\partial w_2} = x_2$$

Loss function

$$J(w) = f - \hat{f}$$

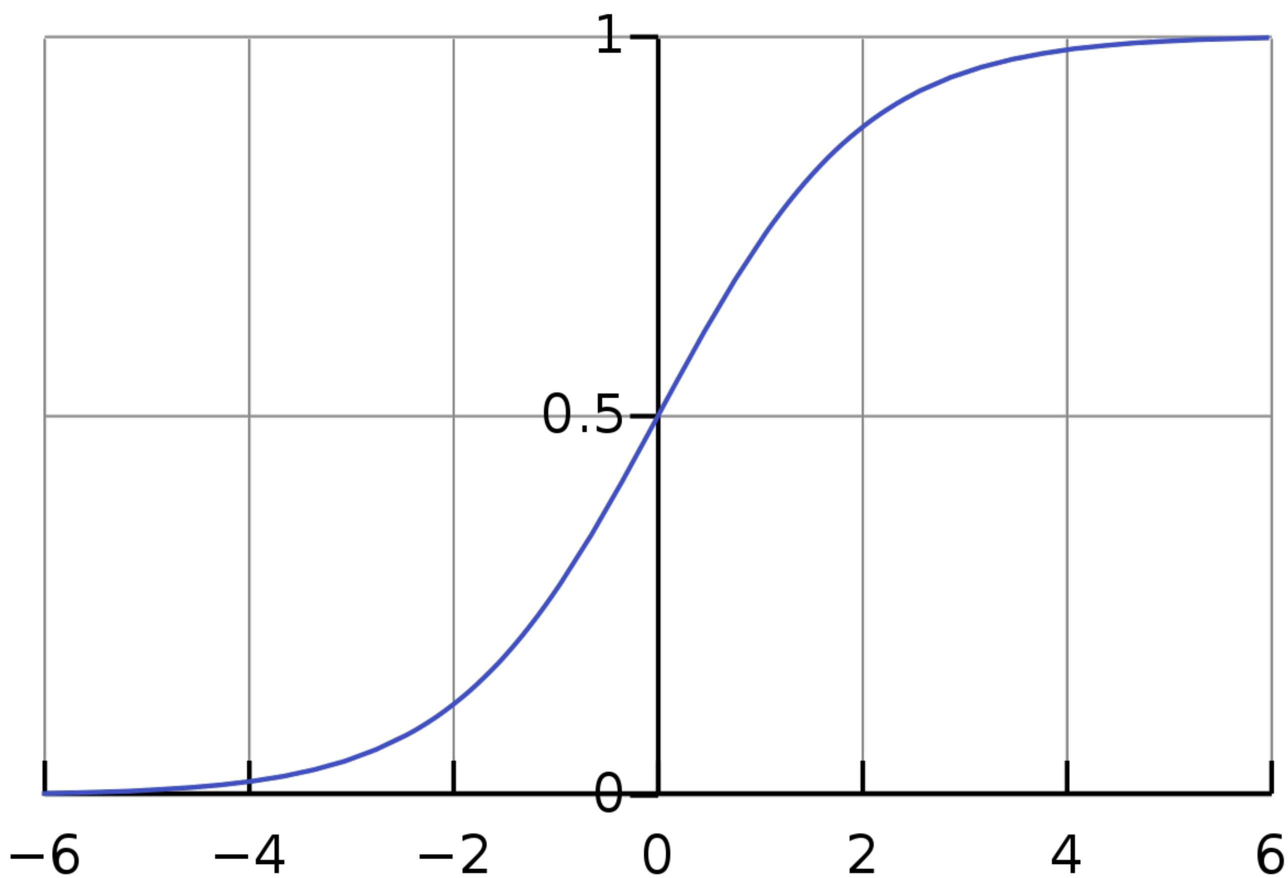
$$w_2 = w_2 + \eta * \frac{\partial J}{\partial f}\frac{\partial f}{\partial q_2}\frac{\partial q_1}{\partial w_2}$$

LOGISTIC REGRESSION

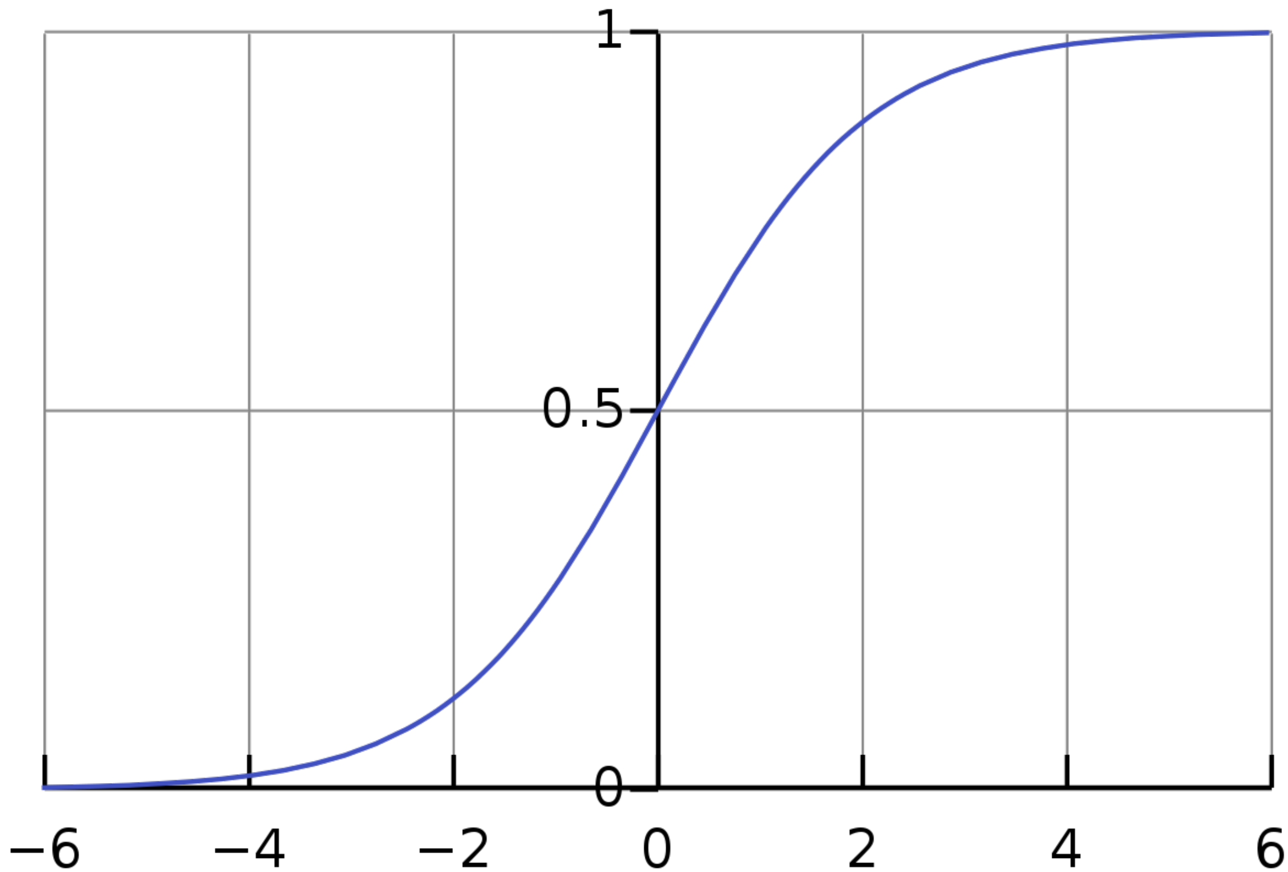


$$f = \frac{1}{1 + e^{-(x_1 w_1 + x_2 w_2)}}$$

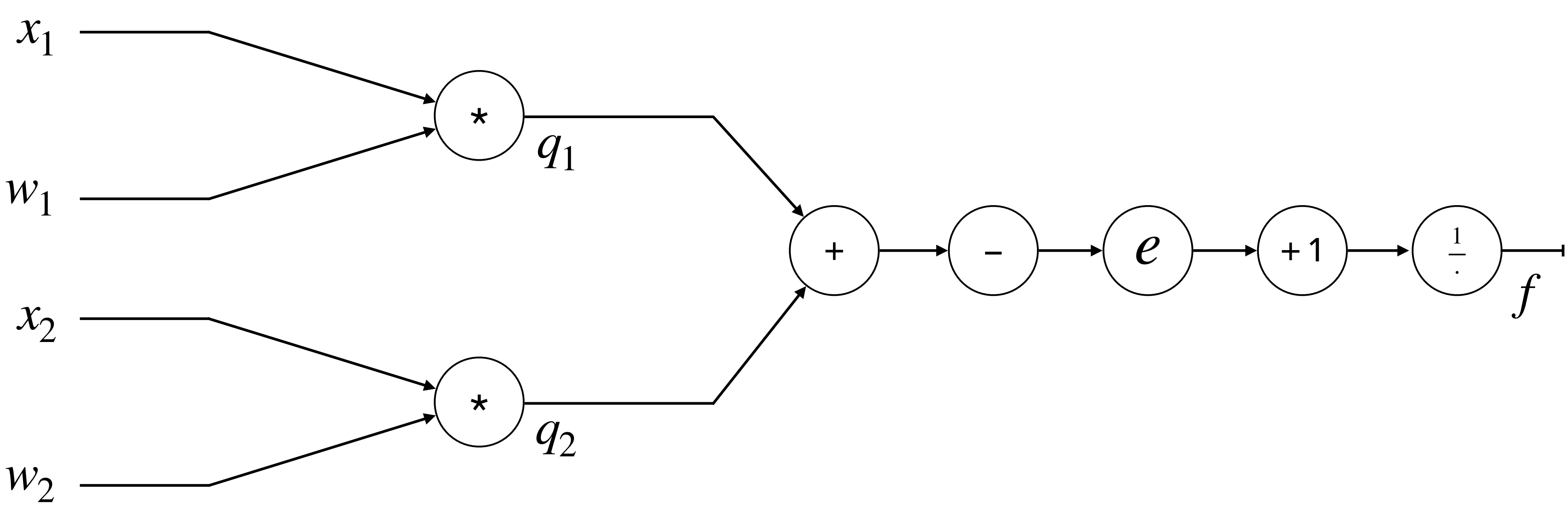
LOGISTIC REGRESSION



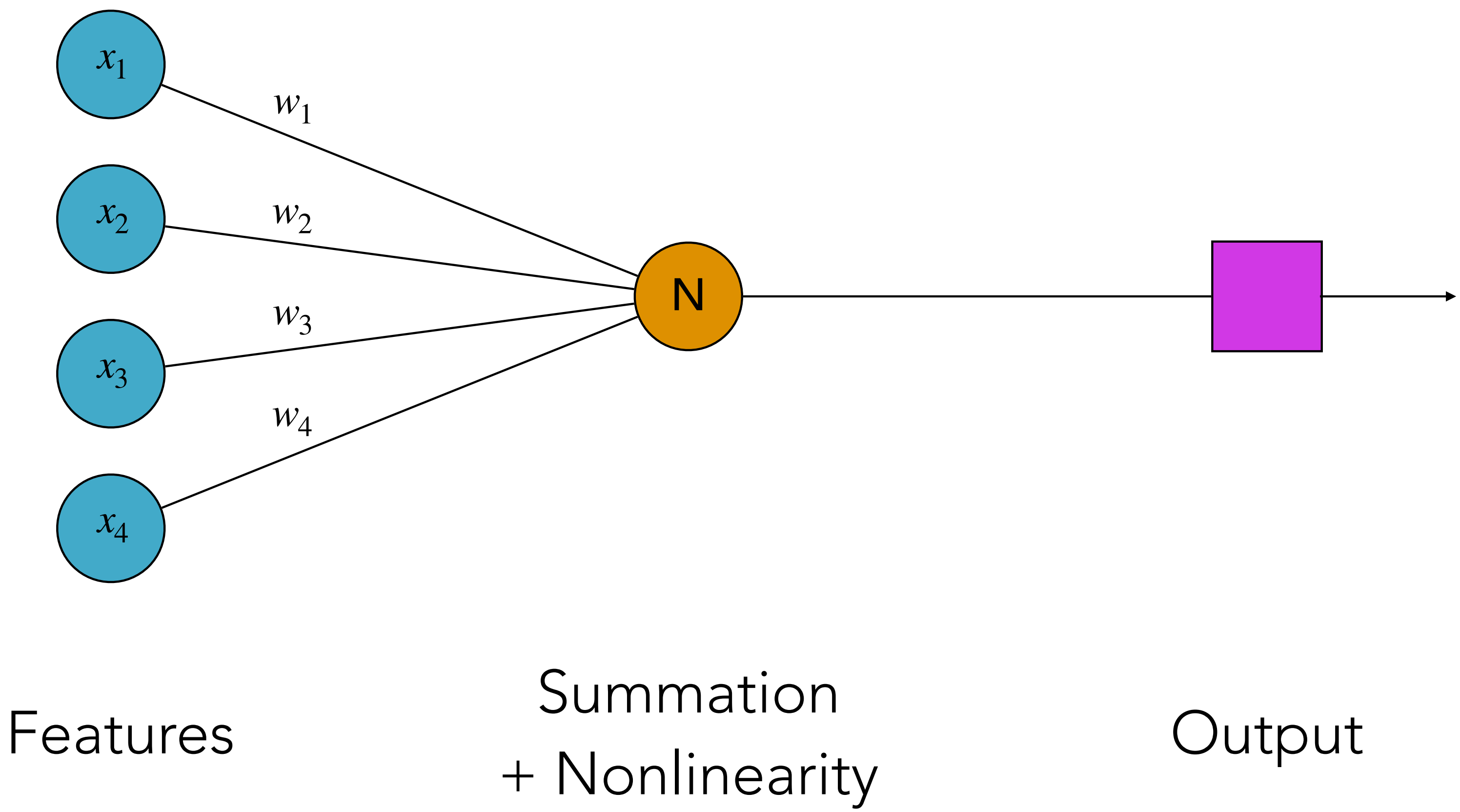
CLASSIFICATION

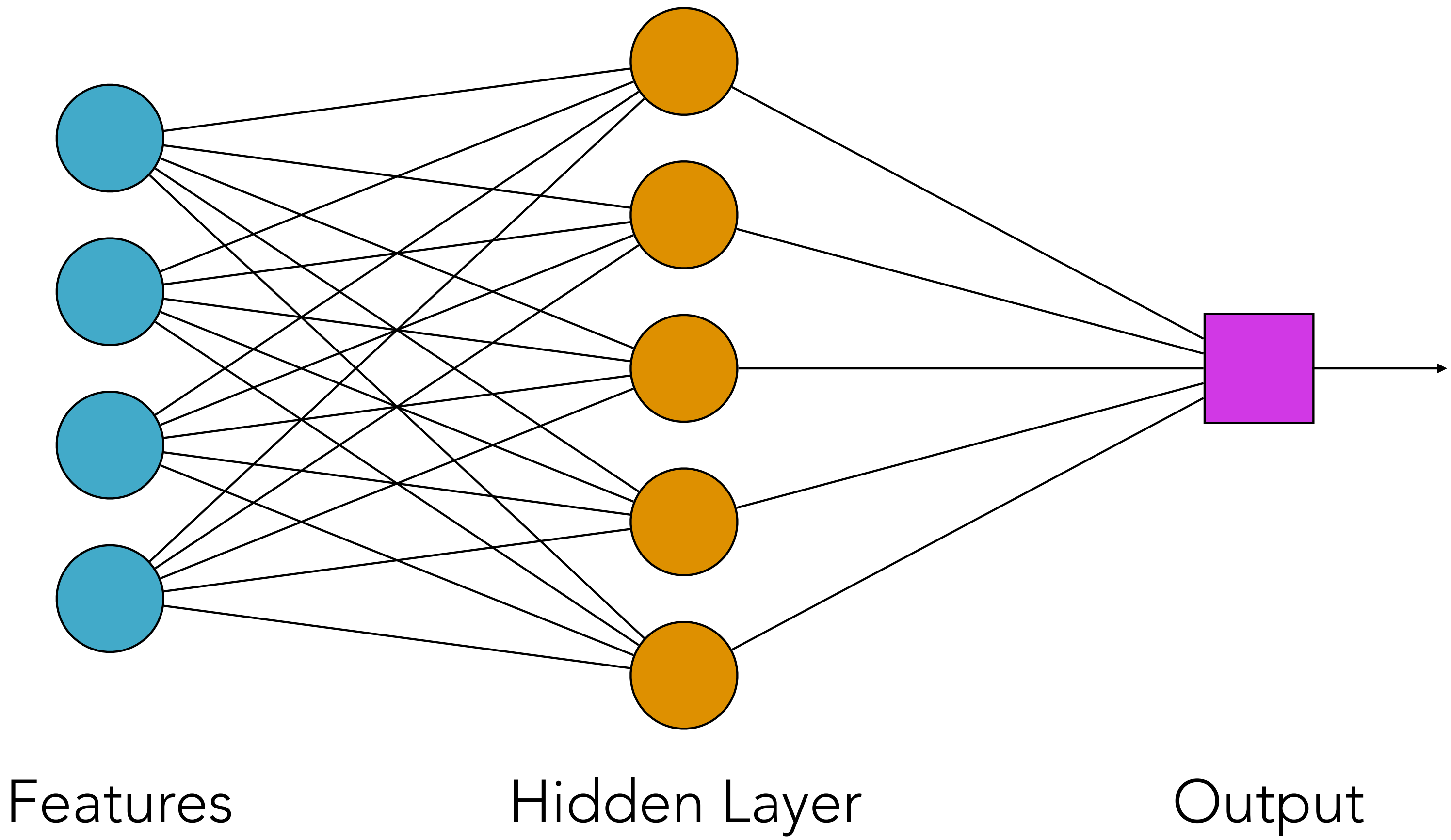


LOGISTIC REGRESSION



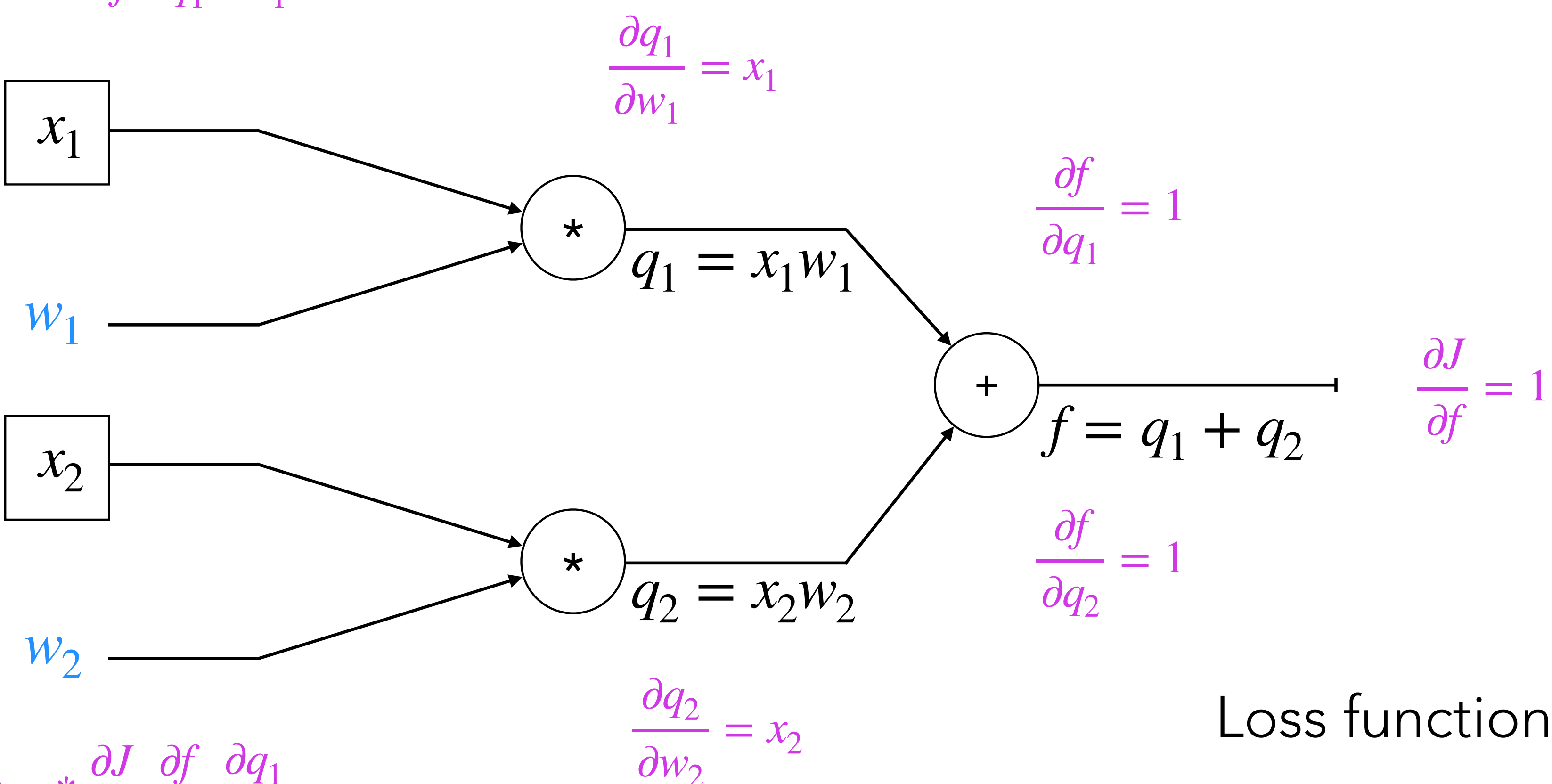
$$f = \frac{1}{1 + e^{-(x_1 w_1 + x_2 w_2)}}$$


x_1
x_2
x_3
x_4
w_1
w_2
w_3
w_4
N
Features
Summation
+ Nonlinearity
Output


Features
Hidden Layer
Output

BACKPROPAGATION

$$w_1 = w_1 + \eta * \frac{\partial J}{\partial f}\frac{\partial f}{\partial q_1}\frac{\partial q_1}{\partial w_1}$$



$$w_2 = w_2 + \eta * \frac{\partial J}{\partial f}\frac{\partial f}{\partial q_2}\frac{\partial q_1}{\partial w_2}$$

Loss function

$$J(w) = f - \hat{f}$$

AUTOMATIC DIFFERENTIATION