

Mathematical notations and terminology

Some notations used in this course are adapted from the notations of the Stanford CS230 course. Reference: <https://cs230.stanford.edu/files/Notation.pdf>

General notations:

(i)	Example number.
m	The number of examples in the dataset.
n_x	Number of features or input samples (input size).
n_y	Number of classes (output size).
$X \in \mathbb{R}^{n_x \times m}$	Input matrix i.e. matrix with input features n_x for all examples m . ¹
$x^{(i)} \in \mathbb{R}^{n_x}$	Column vector of the i^{th} example.
$x_j^{(i)}$	Scalar value of the j^{th} feature for example i^{th} .
$Y \in \mathbb{R}^{n_y \times m}$	Target matrix i.e. matrix with targets n_y for all examples m .
$y^{(i)} \in \mathbb{R}^{n_y}$	Target label for the i^{th} example.
$\hat{y}^{(i)} \in \mathbb{R}^{n_y}$	The predicted output vector from the classifier.
$\underline{y} \in \mathbb{R}^m$	A vector of scalar targets for all examples m .
h	The hypothesis function.
f	Target function i.e. the function we aim to learn.
\hat{h}	The estimated target function using the hypothesis function h .
J	Cost function i.e. cost function for all m examples. ²
E	Error i.e. for a single example.
$\mathcal{N}(\mu, \sigma)$	Normal distribution with mean μ and standard deviation σ .
$w \in \mathbb{R}^{n_x}$	Weights vector in linear and logistic regression.

¹ With exception of the lecture on linear regression the $X \in \mathbb{R}^{m \times n_x}$.

² The function that we aim to minimize or maximize is called the objective function. As we are minimizing it is often called equivalently the cost function, loss function, or error function. The term "cost function" usually refers to an optimization problem and "loss function" usually refers to parameter estimation.



Notations specific for Neural Networks:

Hyperparameters in NN:

α	Learning rate.
β	Momentum
p	Mini batch size
K	Number of iterations for gradient descent.
$n_h^{[l]}$	Number of hidden units of the l^{th} layer.
L	Number of layers in a neural network.
$g^{[l]}$	Activation function for layer l .
k	Learning rate decay
	Features scaling method
	Other model specific hyperparameters (e.g. convolution kernel width in CNN.)

NN variables:

$W^{[l]} \in \mathbb{R}^{n_h^{[l]} \times n_h^{[l-1]}}$	Weight matrix for layer l .
$w_j^{[l]} \in \mathbb{R}^{n_h^{[l]}}$	Weight vector for j^{th} activation at layer l .
$w_{jk}^{[l]} \in \mathbb{R}$	k^{th} weight coefficient for j^{th} activation at layer l i.e. element of $W^{[l]}$ at (j, k)
$b^{[l]} \in \mathbb{R}^{n_h^{[l]}}$	Bias vector at layer l .
$b_j^{[l]} \in \mathbb{R}$	j^{th} bias activation at layer l .
$a^{[l]} \in \mathbb{R}^{n_h^{[l]}}$	Activation vector at layer l .
$a_j^{[l]} \in \mathbb{R}$	j^{th} activation at layer l .

Terminology

Example	Refers to a set of features describing an observation.
Target	The label we are aiming to learn to predict.
Hypothesis class	A space of possible hypotheses for mapping inputs to outputs.
Hypothesis function	An instance of the hypothesis class that maps inputs to outputs.

Acronyms

SGD	Stochastic gradient descent.
BGD	Batch gradient descent.