# **02 - MODEL AND COST FUNCTION**

## **Model representation**

Immagine che contiene testo, Carattere, schermata, linea

Descrizione generata automaticamente

***Notazione***

* x: variabili di input (indipendente) 🡪 features
* y: variabile di output 🡪 variabile di target
* (x, y): un training example
* (x(i), y(i)): l’i-esimo **training example** (riga i)
* m: numero di training examples (numero di righe) 🡪 il numero di (x(i), y(i))

Diagram

Description automatically generated

Quindi l’algoritmo, dato un training set, andrà a trovare una funzione chiamata **ipotesi** “h” *(la curva di fitting)* che mappi un input x ad un output y nel modo migliore possibile.

Dati gli esempi x io vorrei avere come output dei valori molto simili a quelle che sono le etichette delle x.   
Di conseguenza vorrei che l’algoritmo si comporti bene anche con valori nuovi.

Se si parla di **regressione lineare univariata**, la h è una funzione lineare: ***hθ (x) = θ0 + θ1 x***

dove θ0 : intercetta b,  θ1 : pendenza w

## Immagine che contiene testo, linea, schermata, Carattere Descrizione generata automaticamente(**Univariate) Linear regression**

* regressione: y continuo
* lineare: uso una retta
* univariata: abbiamo una sola variabile indipendente (x)

## **Cost function (linear regression)**

Il nostro obiettivo è trovare la retta che fitti nel modo migliore possibile i dati. Per farlo mi serve una funzione di costo.

Quindi avremo:

* un’ipotesi: hθ(x) = θ0 + θ1 x
* θi: i parametri

Variando i parametri cambia la retta.

Devo minimizzare la seguente **funzione di costo** (square cost function):

Text

Description automatically generated

Faccio la differenza tra il valore predetto e il valore vero. Prendo il quadrato di queste differenze dividendolo per due e poi lo faccio per tutti i training examples.

Consideriamo un attimo una **versione semplificata** del problema togliendo il parametro θ0 :

Diagram, text, schematic

Description automatically generatedText

Description automatically generated with low confidence

*Provando diversi valori di θ1 posso andare a capire come l’ipotesi riesca a intercettare i dati e di conseguenza come varia il valore della funzione di costo.*

Chart

Description automatically generated

Il nostro obiettivo è trovare il minimo della funzione di costo

Ritornando al problema originale con i **due parametri**

Chart, scatter chart

Description automatically generated

*Quella che prima era funzione di un solo parametro, ora lo è di due e quindi la funziona di costo diventa una superficie parabolica*

Chart, surface chart

Description automatically generated

Molto spesso si vanno ad utilizzare i **contour plot**: curve altimetriche, tutti i punti sullo stesso ovale hanno lo stesso valore. Più sono piccoli gli ovali, più sto andando sul fondo.

Immagine che contiene testo, linea, diagramma, schermata

Descrizione generata automaticamente

*Quello che vorrei è che la macchina facesse questo per me: trovare θ0 e θ1 che portino la mia curva a configurarsi in maniera ideale -> porta la funzione di costo ad avere il valore minimo. Problema di* ***ottimizzazione****.*