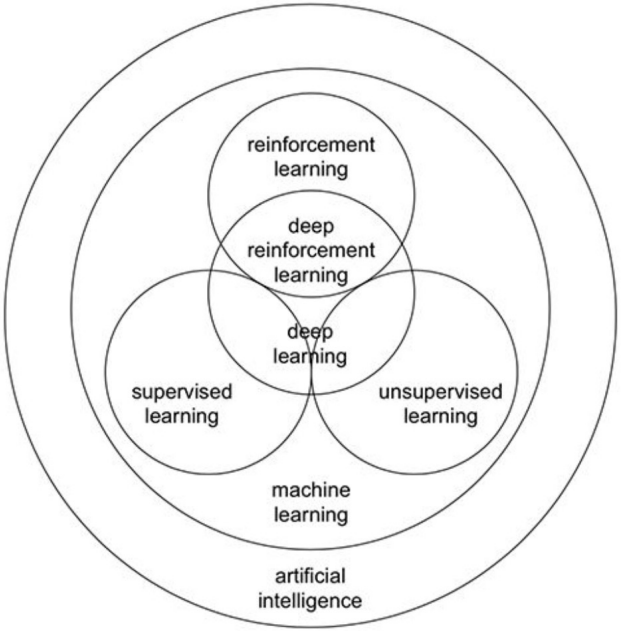
Introduzione al Machine Learning

Docente: Tommaso Muraca

Cos'è il Machine Learning

L'apprendimento automatico (anche detto machine learning in inglese) è una **branca dell'intelligenza artificiale** che raccoglie metodi che utilizzano **metodi statistici** per migliorare la **performance** di un **algoritmo nell'identificare pattern nei dati.**

Nell'ambito dell'informatica, l'apprendimento automatico è una **variante alla programmazione tradizionale** nella quale in una macchina si predispone **l'abilità di apprendere qualcosa dai dati** in maniera **autonoma**, senza istruzioni esplicite.



Importanza del Machine Learning

L'intelligenza artificiale plasmerà il nostro futuro in modo più potente di qualsiasi altra innovazione di questo secolo.

Il tasso di accelerazione è già sorprendente, già nel **2015**, **Google** aveva formato un **agente conversazionale (AI)** che non solo poteva **interagire in modo convincente con gli esseri umani** come helpdesk di supporto tecnico, ma anche **discutere** di moralità, esprimere opinioni e rispondere a domande generali basate sui fatti.

Lo stesso anno, **DeepMind** ha sviluppato un **agente** che ha **superato le prestazioni a livello umano in 49 giochi Atari**, ricevendo come input solo i pixel e il punteggio del gioco. Poco dopo, nel **2016, DeepMind** ha reso **obsoleti i propri risultati** rilasciando un nuovo metodo di gioco all'avanguardia chiamato **A3C**.

Nel frattempo, **AlphaGo** ha **sconfitto** uno dei migliori **giocatori umani a Go**: un risultato straordinario in un gioco dominato dagli umani per due decenni dopo che le macchine avevano conquistato per la prima volta gli scacchi.

Molti **esperti del gioco** non riuscivano a capire come era **possibile per una macchina cogliere tutte le sfumature** e la complessità di questo antico gioco di strategia di guerra cinese, con le sue 10^{170} possibili posizioni sul tabellone (ci sono solo 10^{80}atomi nell'universo).

Importanza del Machine Learning

Nel **marzo 2017, OpenAI** ha creato agenti che hanno inventato il proprio linguaggio per cooperare e raggiungere in modo più efficace il proprio obiettivo. **Poco dopo**, secondo quanto riferito, **Facebook** ha addestrato con successo gli agenti a **negoziare e persino a mentire .**

Poco dopo, l'11 agosto 2017, **OpenAI** ha raggiunto un altro incredibile traguardo **sconfiggendo i migliori professionisti** del mondo in **partite 1v1** del gioco multiplayer online **Dota 2.**

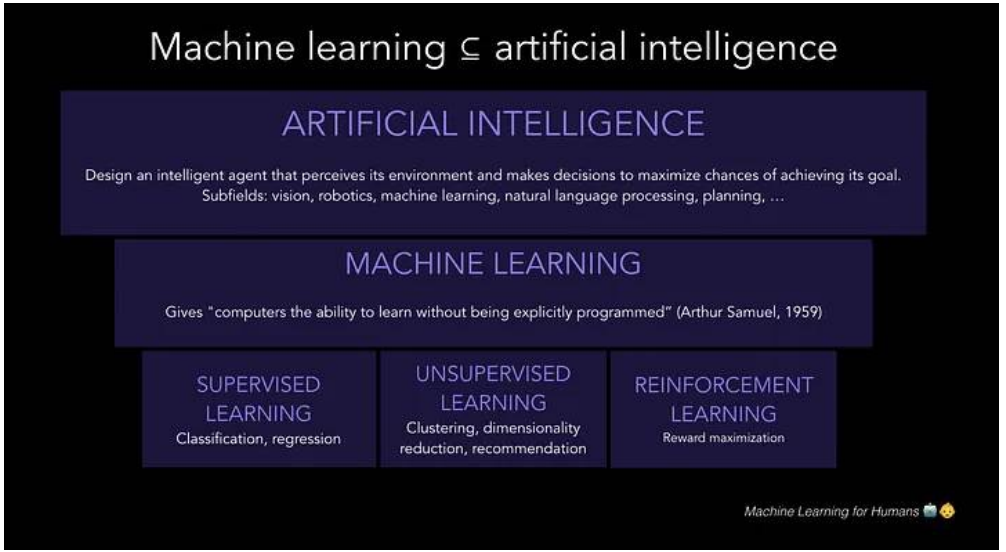
Gran parte della nostra **tecnologia quotidiana** è alimentata **dall'intelligenza artificiale**. Punta la fotocamera sul **menu** durante il tuo prossimo **viaggio a Taiwan** e le selezioni del ristorante appariranno magicamente in italiano tramite **l'app Google Translate**.

Oggi **l'intelligenza artificiale** viene utilizzata per progettare piani di trattamento basati sull'evidenza per i **malati di cancro**, **analizzare istantaneamente** i risultati dei **test medici** per rivolgersi immediatamente allo specialista appropriato e condurre ricerche scientifiche per la scoperta di farmaci.

Intelligenza Artificiale e Machine Learning

L'intelligenza artificiale è lo studio degli agenti che percepiscono il mondo che li circonda, formulano piani e prendono decisioni per raggiungere i **propri obiettivi**. I suoi **fondamenti** includono **matematica, logica, filosofia, probabilità, linguistica, neuroscienze e teoria delle decisioni.** Molti campi rientrano sotto l'egida dell'intelligenza artificiale, come la visione artificiale, la robotica, l'apprendimento automatico e l'elaborazione del linguaggio naturale.

Machine Learning è un **sottocampo dell'intelligenza artificiale**. Il suo **obiettivo è consentire ai computer di apprendere da soli.** L'algoritmo di apprendimento automatico consente di identificare modelli nei dati osservati, costruire modelli che spiegano il mondo e prevedere cose senza avere regole e modelli espliciti preprogrammati.



Intelligenza Artificiale e Machine Learning

Cosa può effettivamente essere definito “intelligenza artificiale”?

Lo standard esatto per la tecnologia che si qualifica come “AI” è **un po’ confuso** e le interpretazioni cambiano nel tempo. L’etichetta AI tende a **descrivere macchine che svolgono compiti tradizionalmente di competenza umana**. È interessante notare che, una volta che i computer capiscono come svolgere uno di questi compiti, gli esseri umani hanno la **tendenza a spostare l’asticella,** questo è noto come **Effetto dell'intelligenza artificiale**.

Ad esempio, **quando Deep Blue dell'IBM sconfisse** il campione mondiale di scacchi Garry Kasparov nel 1997, **la gente si lamentava** del fatto che utilizzava metodi di "forza bruta" e che **non si trattava affatto di "vera" intelligenza**. Come ha scritto Pamela McCorduck, "Fa parte della storia del campo dell'intelligenza artificiale che ogni volta che qualcuno capisce come far fare qualcosa a un computer - giocare bene a dama, risolvere problemi semplici ma relativamente informali – si alza un coro di critici che dicono: 'questo non è pensare'" (McCorduck, 2004).

Forse c’è un argomento inerente a ciò che le persone accettano in modo affidabile come “intelligenza artificiale”:

"L'intelligenza artificiale è tutto ciò che non è stato ancora fatto." -Douglas Hofstadter

Quindi una **calcolatrice conta come AI?** Forse secondo qualche interpretazione. Che **un’auto a guida autonoma**? Oggi sì. In futuro, forse no. La nuova fantastica **startup di chatbot che automatizza un diagramma di flusso**? Certo, perché no.

Intelligenza Generale Artificiale (AGI)

Le **tecnologie discusse finora** sono esempi di **intelligenza artificiale ristretta (ANI)** , che può svolgere efficacemente un compito ben definito.

Nel frattempo, continuiamo a compiere progressi fondamentali verso **l'intelligenza generale artificiale (AGI)** a livello umano, nota anche come **intelligenza artificiale forte** . La definizione di **AGI** è un'intelligenza artificiale in **grado di eseguire con successo qualsiasi compito intellettuale** che un essere umano può fare , incluso l'apprendimento, la pianificazione e il processo decisionale in condizioni di incertezza, comunicare in linguaggio naturale, fare battute, manipolare le persone, scambiare azioni o programmare.

E quest'ultimo è un grosso problema. Una volta creata un'intelligenza artificiale in grado di migliorarsi, sbloccherà un ciclo di **auto-miglioramento ricorsivo** che potrebbe portare a un'esplosione di intelligenza in un periodo di tempo sconosciuto, che va da molti decenni a un solo giorno.

Questo evento viene chiamato **singolarità** . Il termine è preso in prestito dalla singolarità gravitazionale che si verifica al centro di un buco nero, un punto unidimensionale infinitamente denso dove le leggi della fisica come le comprendiamo iniziano a infrangere.

Nel **2017 il Future of Humanity Institute** ha intervistato un gruppo di ricercatori di intelligenza artificiale sulle tempistiche per l'AGI e ha scoperto che "**i ricercatori ritengono che ci sia una probabilità del 50% che l'intelligenza artificiale superi gli esseri umani in tutti i compiti in 45 anni**". Leggendo le impressioni dei professionisti dell'intelligenza artificiale ora però ci sono alcuni che prevedono tempi molto più lunghi (il limite superiore è "mai"), e altri le cui tempistiche sono allarmanti: solo pochi anni.

Super Intelligenza Artificiale (ASI)

L'avvento di una **superintelligenza artificiale (ASI)** di **livello superiore a quello umano** potrebbe essere una delle cose migliori o peggiori che possano accadere alla nostra specie. Porta con sé l'immensa sfida di specificare **cosa vorrà l'intelligenza artificiale** in un modo che sia amichevole per gli esseri umani.

Anche se è **impossibile dire cosa riserva il futuro**, una cosa è certa, ormai ogni momento è un buon momento per iniziare a capire come pensano le macchine. Per andare oltre le astrazioni di un filosofo in poltrona e modellare in modo intelligente le nostre tabelle di marcia e le nostre politiche rispetto all'intelligenza artificiale, dobbiamo impegnarci con i dettagli di come le macchine vedono il mondo: cosa "vogliono", i loro potenziali pregiudizi e modalità di fallimento, il loro peculiarità del temperamento, proprio come studiamo la psicologia e le neuroscienze per capire come gli esseri umani apprendono, decidono, agiscono e sentono.

L'apprendimento automatico può essere messo al centro di questo viaggio verso l'intelligenza artificiale generale e, nel frattempo, **cambierà ogni settore e avrà un impatto enorme sulla nostra vita quotidiana.**

Apprendimento Supervisionato

Quanti soldi guadagneremo spendendo più dollari in sponsorizzate? Ha senso concedere un prestito a questa persona? Cosa accadrà domani al mercato azionario?

Nei **problemi di apprendimento supervisionato**, iniziamo con un **set di dati** contenente **esempi di training con etichette corrette** associate . Ad esempio, quando si impara a **classificare le cifre scritte a mano**, un **algoritmo di apprendimento supervisionato** scatta migliaia di immagini di cifre scritte a mano insieme ad etichette contenenti il numero corretto rappresentato da ciascuna immagine. **L'algoritmo apprenderà quindi la relazione tra le immagini e i numeri associati** e applicherà tale relazione appresa per classificare **immagini completamente nuove (senza etichette)** che la macchina non ha mai visto prima. **Ecco come puoi far leggere del testo alla tua fotocamera!**

Per illustrare **come funziona l'apprendimento supervisionato**, esaminiamo il problema di **prevedere il reddito annuo in base al numero di anni di istruzione superiore completati da qualcuno**. In termini più formali, vorremmo costruire un modello che approssimi la relazione f tra il numero di anni di istruzione superiore X e il corrispondente reddito annuo Y .

$$Y = f(X) + \epsilon$$

X (input) = anni di istruzione superiore **, Y (output)** = reddito annuo , **f** = funzione che descrive la relazione tra X e Y ,

ϵ (epsilon) = termine di errore casuale (positivo o negativo) con media nulla. **Epsilon** quindi rappresenta un **errore irriducibile nel modello**, che è un limite teorico alle prestazioni del tuo algoritmo a **causa del rumore intrinseco nei fenomeni che stai cercando di spiegare**. Ad esempio, immagina di costruire un modello per prevedere il risultato del lancio di una moneta.

Apprendimento Supervisionato

Un metodo per prevedere il reddito **potrebbe essere** quello di creare un **modello rigido basato su regole per la relazione tra reddito e istruzione**. Ad esempio: "**Stimo che per ogni anno in più di istruzione superiore, il reddito annuo aumenta di 5.000 dollari".**

reddito = ($ 5.000 * anni_di_istruzione) + reddito_di_base

Questo approccio è un **esempio di ingegneria di una soluzione** (tra un po' parleremo invece **dell'apprendimento di una soluzione**, realizzato con il metodo di **regressione lineare**).

Potremmo elaborare un **modello più complesso** includendo alcune **regole sul tipo di laurea, anni di esperienza lavorativa, livelli scolastici, ecc**. Ad esempio: "**Se hanno conseguito una laurea o un titolo superiore, fornisci alla stima del reddito un moltiplicatore 1,5x".**

Ma **questo tipo** di programmazione esplicita basata su regole **non funziona bene con dati complessi**. Immaginiamo di **provare a progettare un algoritmo di classificazione delle immagini** composto da **istruzioni if-then** che descrivono le combinazioni di **luminosità dei pixel** che dovrebbero essere etichettate come **"gatto" o "non gatto"**.

L'apprendimento automatico supervisionato risolve questo problema facendo sì che il **computer svolga il lavoro per te**. Identificando **modelli nei dati**, la macchina è in grado di **formare proprie soluzioni**. La **differenza** principale tra questo e **l'apprendimento umano** è che **l'apprendimento automatico** funziona sull'hardware del computer ed è meglio **compreso attraverso la lente dell'informatica e della statistica**, mentre il pattern-matching umano avviene in un cervello biologico (pur raggiungendo gli stessi obiettivi).

Nell'apprendimento supervisionato, la **macchina tenta di apprendere da zero** la relazione tra reddito e istruzione , eseguendo dati di formazione etichettati attraverso un algoritmo di apprendimento. Questa **funzione appresa** può essere utilizzata per stimare il reddito di persone il cui **reddito Y** non è noto, a patto di avere come input gli **anni di istruzione X**. In altre parole, possiamo applicare il nostro modello ai dati di test senza etichetta per stimare Y.

L'obiettivo dell'apprendimento supervisionato è prevedere Y nel modo più accurato possibile quando vengono **forniti nuovi esempi in cui X è noto e Y è sconosciuto**.

Regressione

I **due** compiti **dell'apprendimento supervisionato** sono la **regressione e la classificazione**:

Regressione: prevedere un **valore numerico continuo**. A quanto verrà venduta quella casa?

Classificazione: assegnare **un'etichetta**. Questa è la foto di un gatto o di un cane?

Partiamo dalla **regressione** e spieghiamola nel dettaglio, prevede una variabile **target continua Y** e consente di **stimare un valore**, ad esempio i prezzi delle case o la durata della vita umana, **sulla base dei dati di input X**.

La **variabile target** indica la **variabile sconosciuta che ci interessa prevedere** e **continuo** significa che **non ci sono lacune (discontinuità) nel valore che Y può assumere**. Il peso e l'altezza di una persona **sono valori continui**. Le **variabili discrete** , d'altro canto, possono assumere solo un **numero finito di valori**: ad esempio, il numero di figli che qualcuno ha è una variabile discreta.

La **previsione del reddito** è un **classico problema di regressione**. I dati di **input X** includono **tutte le informazioni rilevanti** sugli individui nel set di dati che possono essere utilizzate per prevedere il reddito, come **anni di istruzione**, anni di **esperienza lavorativa**, **titolo professionale** o **codice postale**. Questi attributi sono chiamati **caratteristiche** , che **possono essere numerici** (es. anni di esperienza lavorativa) **o categoriali** (es. titolo professionale o campo di studio).

Avrai bisogno di quante **più osservazioni di addestramento possibili** mettendo in **relazione queste funzionalità con l'output target Y**, in modo che il tuo **modello possa apprendere la relazione f tra X e Y**.

I dati sono suddivisi in un **set di dati di training e un set di dati di test** . Il set di **training ha etichette**, quindi il tuo **modello può imparare da questi esempi etichettati.** Il **set di test non ha etichette**, ovvero non conosci ancora il valore che stai cercando di prevedere. È importante che il tuo **modello possa generalizzare a situazioni mai incontrate** prima in modo che possa funzionare bene sui dati di test.

Regressione

Regressione

Y = f(X) + ϵ , dove X = (x1, x2…xn)

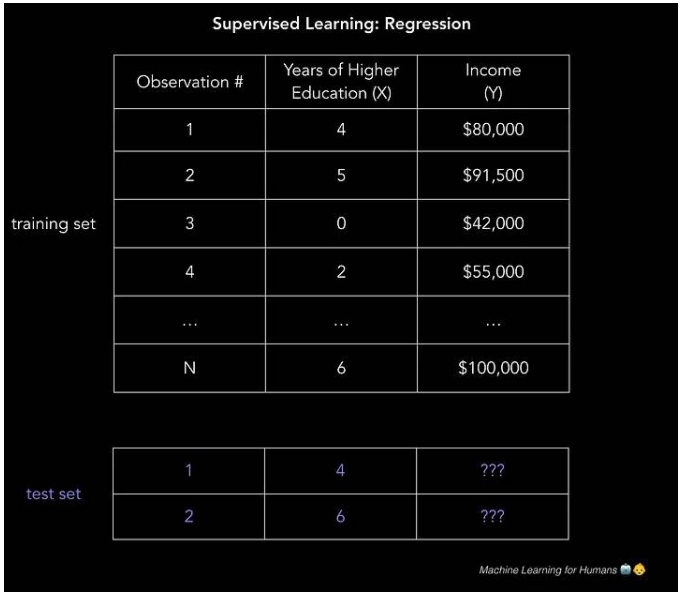
Addestramento : la macchina apprende f dai dati di addestramento etichettati

Test: la macchina prevede Y da dati di test senza etichetta

Si noti che **X può essere un tensore con un numero qualsiasi di dimensioni**. Un **tensore 1D** è un vettore **(1 riga, molte colonne),** un **tensore 2D** è una **matrice (molte righe, molte colonne)**, e quindi si possono avere tensori con 3, 4, 5 o più dimensioni (es. un tensore 3D con righe , colonne e profondità).

Nel nostro **banalmente semplice esempio 2D**, ciò **potrebbe assumere la forma di un file .csv** in cui ogni riga contiene il livello di istruzione e il reddito di una persona. **Aggiungendo più colonne con più funzionalità e avremo un modello più complesso, ma forse più accurato.**

Come possiamo costruire modelli che forniscano previsioni accurate e utili nel mondo reale? Lo facciamo utilizzando **algoritmi di apprendimento supervisionato.**



Regressione Lineare

Come **primo passo** ci concentreremo sulla **risoluzione del problema della previsione del reddito** con un **algoritmo di regressione lineare**, poiché i **modelli lineari non funzionano bene con le attività di riconoscimento delle immagini (questo è il dominio del deep learning, che esploreremo più avanti).**

Abbiamo il nostro **set di dati X** e i corrispondenti valori target Y. L'**obiettivo** della **regressione dei minimi quadrati ordinari (OLS)** è **apprendere un modello lineare** che possiamo utilizzare per **prevedere un nuovo y dato un x mai visto prima** con il **minor errore possibile.** Vogliamo indovinare quanto reddito guadagna qualcuno in base a quanti anni di istruzione ha ricevuto.

X_train = [4, 5, 0, 2, …, 6] # anni di istruzione post-secondaria

y_train = [80, 91,5, 42, 55, …, 100] # redditi annuali corrispondenti, in migliaia di dollari

La **regressione lineare** è un **metodo parametrico** , il che significa che **fa un'ipotesi sulla forma della funzione** che mette in **relazione X e Y**. Il nostro modello sarà una funzione che prevede ŷ dato un x specifico :

$$\hat{y} = \beta_0 + \beta_1 * x + \epsilon$$

β0 è l'intercetta y e **β1 è la pendenza della nostra retta**, ovvero di **quanto aumenta (o diminuisce) il reddito** con un anno di istruzione in più.

Il nostro **obiettivo è apprendere i parametri del modello (in questo caso, β0 e β1) che riducono al minimo l'errore nelle previsioni del modello.**

Regressione Lineare

Per trovare i parametri migliori:

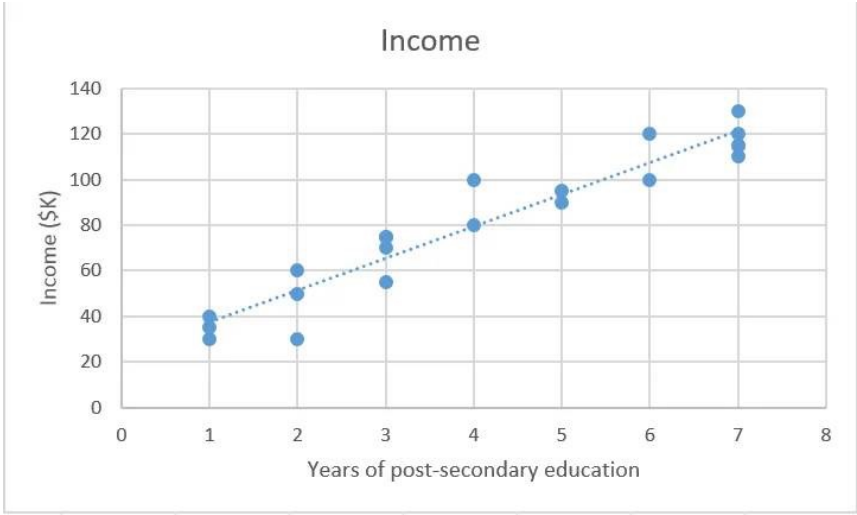
1. Definire una **funzione di costo** , o **funzione di perdita** , che **misuri quanto siano imprecise le previsioni del nostro modello.**

2. **Trovare i parametri che minimizzano le perdite** , ovvero rendere il nostro **modello quanto più accurato possibile.**

Graficamente, in due dimensioni, ciò si traduce in una **linea di migliore adattamento**. In tre dimensioni disegneremmo un piano e così via con iperpiani di dimensioni superiori.

Matematicamente, osserviamo la **differenza tra ciascun punto dati reale (y) e la previsione del nostro modello (ŷ)**. **Elevare al quadrato queste differenze per evitare numeri negativi e penalizzare differenze maggiori, quindi sommarle e ottenere la media**. Questa è una misura di quanto bene i nostri dati si adattano alla linea.

$$Cost = \frac{\sum_1^n ((\beta_1 x_i + \beta_0) - y_i))^2}{2 * n}$$



n = numero di osservazioni. **Usare 2*n** invece di n fa sì che i calcoli funzionino in modo più pulito quando si prende la derivata per ridurre al minimo la perdita, anche se alcuni esperti di statistica dicono che questa è una bestemmia.

Regressione Lineare e discesa del gradiente

Per un **problema semplice** come questo, possiamo **calcolare una soluzione in forma chi**usa utilizzando il calcolo infinitesimale per **trovare i parametri beta ottimali** che minimizzano la nostra funzione di perdita. Ma **man mano che la funzione di costo diventa più complessa**, **trovare una soluzione in forma chiusa con il calcolo infinitesimale non è più fattibile**. Questa è la **motivazione per un approccio iterativo chiamato discesa del gradiente** , che ci consente di **minimizzare una funzione di perdita complessa**.

L'obiettivo della **discesa del gradiente** è **trovare il minimo della funzione di perdita** del nostro modello ottenendone iterativamente un'approssimazione sempre migliore.

Immaginando di camminare attraverso una valle con una benda sugli occhi con l'obiettivo è trovare il fondo della valle. Come lo faresti?

Un **approccio ragionevole** sarebbe quello di **toccare il terreno intorno a noi e muoverci nella direzione in cui il terreno è in pendenza più ripida**. Faremo un passo e ripeteremo lo stesso procedimento continuamente **finché il terreno non sarà piatto**. Allora sapremo di aver raggiunto il **fondo di una valle,** se **spostandoci in qualsiasi direzione** da dove ci troviamo, ci ritroveremo alla stessa altitudine o più in salita.

Tornando alla **matematica**, il **terreno diventa la nostra funzione di perdita** e la **quota del fondovalle è il minimo di tale funzione**.

Se ripensiamo attentamente alla **funzione di perdita** ci rendiamo conto che questa in realtà una **funzione di due variabili: β0 e β1** . Tutte le **restanti variabili vengono determinate**, poiché X, Y e n vengono fornite **durante l'addestramento**. Vogliamo provare a **ridurre al minimo questa funzione.**

La **funzione è f(β0,β1)=z.** Per **iniziare la discesa del gradiente**, si fa qualche **ipotesi sui parametri β0 e β1** che **minimizzano la funzione**.

Successivamente, trovi le **derivate parziali della funzione di perdita rispetto a ciascun parametro beta: [dz/dβ0, dz/dβ1]**. Una derivata parziale indica di quanto la perdita totale aumenta o diminuisce se si aumenta β0 o β1 di una quantità molto piccola.

Regressione Lineare e discesa del gradiente

In altre parole, **quanto aumentando la nostra stima del reddito annuo assumendo un livello di istruzione superiore pari a zero (β0) aumenterebbe la perdita (cioè l'imprecisione) del nostro modello?**

Allo stesso modo, **se aumentiamo la stima di quanto ogni anno di istruzione influisce sul reddito (β 1), quanto ciò aumenta la perdita (z)?**

Se la **derivata parziale dz/β1** è un **numero negativo** , allora **β1 è positivo perché ridurrà la perdita totale**. Se la **derivata** è un **numero positivo , dobbiamo diminuire β1** . Se è **zero, non dobbiamo modificare β1 perché significa che abbiamo raggiunto un ottimo punto**.

Continuiamo a farlo finché non **raggiungiamo il fondo**, ovvero l'algoritmo converge e la **perdita è stata ridotta al minimo**. Esistono molti trucchi e casi eccezionali che esulano dallo scopo di questo lavoro, ma in generale è così che si trovano i parametri ottimali per il modello parametrico.

Andiamo ora a vedere qualche esempio pratico.