

AI i ML: tipus

Lloc: [Institut d'Ensenyaments a Distància de les Illes Balears](#)

Curs: Sistemes d'aprenentatge automàtic

Llibre: AI i ML: tipus

Imprès per: Carlos Sanchez Recio

Data: dimecres, 2 d'octubre 2024, 07:11

Taula de continguts

1. Tipus d'IA

- 1.1. IA general i IA particular
- 1.2. IA forta i IA feble
- 1.3. Hi pot haver IA forta sense cos?
- 1.4. La IA específica du a la IA general?
- 1.5. La IA abans de l'aprenentatge automàtic
- 1.6. IA artificial i IA humana

2. Aprendre a partir d'exemples

- 2.1. Formes d'aprenentatge
- 2.2. Aprenentatge supervisat

3. Aprenentatge automàtic

- 3.1. Tipus d'aprenentatge automàtic
- 3.2. AA vs. programació tradicional
- 3.3. Aprenentatge profund
- 3.4. Vídeo d'aprenentatge automàtic

4. Aprenentatge supervisat

- 4.1. Classificació
- 4.2. Regressió

5. Aprenentatge no supervisat

- 5.1. Anàlisi de clústers
- 5.2. Algorismes de clústering
- 5.3. Reducció de la dimensionalitat
- 5.4. PCA i PCA amb kernel
- 5.5. Aprenentatge supervisat vs. no supervisat

6. Aprenentatge per reforç

- 6.1. Aprendre de les recompenses
- 6.2. Aprenentatge per reforç passiu
- 6.3. Aprenentatge per reforç actiu
- 6.4. Aplicacions

7. IA generativa

- 7.1. Exemples
- 7.2. Sistemes de diferents empreses
- 7.3. Com funciona la IA generativa?
- 7.4. IA tradicional vs. IA generativa
- 7.5. Text
- 7.6. Text a imatge
- 7.7. Text a codi
- 7.8. Text a vídeo

8. Bibliografia

1. Tipus d'IA

Els filòsofs s'han estat plantejant grans qüestions des de fa molt de temps.

- Com funciona la ment?
- Les màquines podem actuar intel·ligentment així com ho fan les persones?
- Una màquina així, tendria una veritable ment conscient?

L'objectiu últim de la intel·ligència artificial (IA) és aconseguir que una màquina tingui una intel·ligència de tipus general similar a la humana. Aquesta és una de les fites més ambiciosos que s'ha plantejat la ciència. És d'una dificultat comparable a altres grans objectius científics com per exemple explicar l'origen de la vida o l'origen de l'univers, o bé conèixer l'estructura de la matèria. Al llarg dels darrers segles, aquest afany per construir màquines intel·ligents ha conduït a inventar models o metàfores del cervell humà.

Per exemple, el segle XVII, Descartes es va demanar si un complex sistema mecànic compost d'engranatges, politges i tubs podria, en principi, emular el pensament.

Dos segles després, la metàfora es va plasmar en els sistemes telefònics, ja que semblava que les seves connexions es podien assimilar a una xarxa neuronal.

Actualment el model dominant és el model computacional basat en l'ordinador digital.

1.1. IA general i IA particular

Hi ha aplicacions d'intel·ligència artificial molt bones en tasques específiques, com ara conduir un cotxe, jugar a escacs o reconèixer la parla.

Malgrat aquests èxits, alguns dels fundadors de la intel·ligència artificial (John McCarthy, Marvin Minsky, Nils Nilsson, Patrick Winston) posen més poc èmfasi a crear versions sempre millorades d'aquestes aplicacions. En canvi, pensen que la IA hauria de tornar a les seves arrels de cercar, en paraules de Simon, "**màquines que pensen, que aprenen i que creen**". Anomenen aquest projecte intel·ligència artificial de nivell humà (*human-level AI*, *HLAI*). Aquest esforç necessitarà bases de coneixement molt grans, que s'hauran d'obtenir de fonts diverses.

Una idea relacionada és el subcamp de la **intel·ligència artificial general** (*Artificial General Intelligence*, *AGI*), que va tenir la seva primera conferència el 2008. L'AGI cerca un algorisme universal per a aprendre i actuar en qualsevol entorn, i té les seves arrels en la feina de Ray Solomonoff (1964), un dels participants a la conferència de Dartmouth de 1956.

També hi ha interès a garantir que aquesta creació sigui una intel·ligència general amistosa (**Friendly AI**), (Yudkowsky, 2008; Omohundro, 2008)

La intel·ligència artificial estreta (Artificial Narrow Intelligence) inclou totes les aplicacions actuals del camp: reconeixement d'imatges, joc competitiu, diagnosi mèdica, recomanació de productes. Té com a avantatge l'especialització en la tasca d'interès, així com una precisió en la realització de la tasca comparable o superior a la dels humans, i l'inconvenient d'haver-se d'entrenar amb una gran quantitat d'informació.

La intel·ligència artificial general és encara un projecte, que permetria la seva aplicació en qualsevol camp, sense haver-se d'entrenar específicament per a una tasca concreta. Planteja el problema del control, com assegurar que els humans mantenim el domini de la situació de forma que l'activitat de la IA mai no pugui ser nociva per a les persones. En aquest sentit, Stuart Russell defensa que cal alinear la IA amb els valors humans.

A l'article [Cap a la intel·ligència artificial](#), Ramon López de Mántaras ens ofereix la següent distinció entre IA general i específica.

Allen Newell i Herbert Simon van formular la hipòtesi segons la qual tot sistema de símbols físics posseeix els mitjans necessaris i suficients per dur a terme accions intel·ligents (Newell i Simon, 1976). D'altra banda, com que els éssers humans som capaços de mostrar conductes intel·ligents, d'acord amb la hipòtesi, nosaltres som també sistemes de símbols físics. Convé aclarir a què es refereixen Newell i Simon. Un sistema de símbols físics consisteix en un conjunt d'entitats anomenades símbols que, mitjançant relacions, poden ser combinats per formar estructures més grans –com els àtoms que es combinen formant molècules– i que poden ser transformats aplicant un conjunt de procediments. Aquests procediments poden crear nous símbols, crear i modificar relacions entre aquests, emmagatzemar-ne, comparar si dos són iguals o diferents, etcètera. Aquests símbols són físics en tant que tenen un substrat físicoelectrònic (en el cas dels ordinadors) o físicobiològic (en el cas dels éssers humans). Efectivament, en el cas dels ordinadors els símbols es realitzen mitjançant circuits electrònics digitals i en el cas dels éssers humans, mitjançant xarxes de neurones. En definitiva, d'acord amb la hipòtesi del sistema de símbols físics, la naturalesa del substrat (circuitos electrònics o xarxes neuronals) no té importància sempre que aquest permeti processar símbols. No oblidem que es tracta d'una hipòtesi i per tant la seva validesa o refutació s'haurà de verificar d'acord amb el mètode científic. La intel·ligència artificial és precisament el camp científic dedicat a intentar verificar aquesta hipòtesi en el context dels ordinadors, és a dir, verificar si un ordinador convenientment programat és capaç o no de tenir una conducta intel·ligent de tipus general.

A continuació, es matisa que

s'hauria de tractar d'intel·ligència de tipus general, i no una d'específica, ja que la intel·ligència dels éssers humans és de tipus general. Exhibir intel·ligència específica és una altra cosa ben diferent. Per exemple, els programes que juguen als escacs al nivell de gran mestre són incapaços de jugar a les dames. Es requereix un programa diferent perquè el mateix ordinador jugui a les dames; és a dir, aquest no pot aprofitar el fet que juga als escacs per adaptar-se i jugar també a les dames. En el cas dels éssers humans, qualsevol jugador d'escacs pot aprofitar els seus coneixements sobre aquest joc per a jugar a les dames perfectament.

1.2. IA forta i IA feble

El 1980, el filòsof John Searle introduí la distinció entre **IA feble** (*weak AI*) i **IA forta** (*strong AI*).

La idea de la **intel·ligència artificial feble** és que les màquines poden actuar **com si** fossin intel·ligents.

La hipòtesi de la **intel·ligència artificial forta** és l'afirmació que les màquines que actuen intel·ligentment estan **pensant realment, no** només **simulant** el pensament.

En el camp de la intel·ligència artificial, la majoria d'investigadors donen per feta la hipòtesi de la IA feble, i no tenen en compte la hipòtesi de la IA forta. L'èmfasi es posa en el bon funcionament dels programes, independentment que ho considerem **pensament simulat o real**.

Amb el temps, la definició d'**IA forta** ha passat a referir-se a l'anomenada **IA de nivell humà** (*Human Level AI*) o **IA general**: programes que poden resoldre una varietat de tasques arbitràriament àmplia, incloent-ne de noves, i que ho fan tan bé com un humà.

Ramon López de Mántares distingeix entre IA general i IA forta, assenyalant la connexió només en un sentit. Tota IA forta és general però hi pot haver IA general que no sigui IA forta. Seria una IA capaç d'exhibir intel·ligència general com una ment però sense ésser una ment.

En canvi, sí que podem identificar IA feble amb IA específica:

La IA feble, d'altra banda, consistiria, segons Searle, a construir programes que realitzin tasques específiques. La capacitat dels ordinadors per a realitzar tasques específiques fins i tot millor que les persones ja s'ha demostrat àmpliament en certs dominis, com ara buscar solucions a fórmules lògiques amb moltes variables i altres aspectes relacionats amb la presa de decisions. També s'associa amb la IA feble el fet de formular i provar hipòtesis sobre aspectes relacionats amb la ment (per exemple, la capacitat de raonar deductivament, d'aprendre inductivament, etc.) mitjançant la construcció de programes que duen a terme aquestes funcions encara que sigui mitjançant processos completament diferents dels que duu a terme el cervell. Absolutament tots els avenços aconseguits fins ara en el camp de la IA són manifestacions de la **IA feble i específica**.

Ramon López de Mántares, [Cap a la intel·ligència artificial](#)

1.3. Hi pot haver IA forta sense cos?

El filòsof **Hubert Dreyfus** afirmà que l'objectiu final de la IA, una IA forta de tipus general, és tan inabastable com l'objectiu dels alquimistes del segle XVII, transformar el plom en or. Dreyfus explicava que el cervell processa la informació de forma global i contínua; en canvi, un ordinador usa un conjunt finit i discret d'operacions aplicades a un conjunt de dades. A més, Dreyfus també considera crucial el paper que el cos representa respecte del desenvolupament de la intel·ligència. Considera necessari que la intel·ligència formi part d'un cos a través del qual pugui interactuar amb el món. Així, la intel·ligència dels éssers vius seria conseqüència del fet d'estar situats a un entorn amb el qual poden interactuar. Filosòficament, aquesta necessitat de corporeïtat es basa en la **fenomenologia de Heidegger**, que subratlla el paper fonamental del cos amb les seves necessitats, desitjos, plaers, penes i formes de moure's i d'actuar.

Així, la IA hauria de modelitzar tots aquests aspectes per poder assolir l'objectiu de la IA forta, és a dir, ésser una ment. Com a conclusió, Dreyfus no nega la possibilitat de la IA forta, però la considera impossible d'implantar amb els mètodes clàssics de la IA no corpòria.

1.4. La IA específica du a la IA general?

Seguint López de Mántaras,

Pràcticament tots els projectes en IA s'han centrat a construir intel·ligències artificials especialitzades i els èxits assolits en només seixanta anys d'existència, i en particular durant l'últim decenni, són molt impressionants, principalment gràcies a la conjunció de dos elements: la disponibilitat d'enormes quantitats de dades i l'accés a la computació d'altres prestacions per a poder analitzar-les.

Tanmateix, aquests avenços no han aconseguit arribar a la IA general.

El que semblava més difícil (diagnosticar malalties, o jugar a escacs i a Go al més alt nivell) ha resultat factible i el que semblava més fàcil (comprendre el significat profund del llenguatge o interpretar una escena visual) encara no s'ha assolit.

L'explicació a aquesta aparent contradicció cal buscar-la en la dificultat de dotar les màquines de coneixements de sentit comú. El sentit comú és el requisit fonamental per aconseguir IA similar a la humana pel que fa a generalitat i profunditat. Els coneixements de sentit comú són fruit de vivències i experiències obtingudes interactuant amb el nostre entorn. Sense aquests coneixements no és possible una comprensió profunda del llenguatge ni una interpretació profunda del que capta un sistema de percepció visual, entre altres limitacions. Les capacitats més complicades d'assolir són aquelles que requereixen interaccionar amb entorns no restringits ni prèviament preparats. Dissenyar sistemes que tinguin aquestes capacitats requereix integrar desenvolupaments en moltes àrees de la IA. En particular, necessitem llenguatges de representació de coneixements que codifiquin informació sobre molts tipus diferents d'objectes, situacions, accions, etc., així com de les seves propietats i de les relacions entre ells.

A més,

També necessitem nous algorismes que, a partir d'aquestes representacions, puguin respondre, de forma robusta i eficient, a preguntes sobre pràcticament qualsevol tema. Finalment, com que necessitaran conèixer un nombre pràcticament il·limitat de coses, aquests sistemes han de ser capaços d'aprendre nous coneixements de manera contínua al llarg de tota la seva existència.

Així, la construcció d'IA general passa per la integració de **percepció, representació, raonament, acció i aprenentatge**.

1.5. La IA abans de l'aprenentatge automàtic

Fins molt recentment, el model dominant en IA ha estat el simbòlic. Aquest model té les arrels en la hipòtesi del sistema de símbols físics. Encara continua sent molt important i actualment es considera el model "clàssic" en IA. És un model descendent (*top-down*) que es basa en el **raonament lògic** i la **cerca heurística** com a pilars per a la resolució de problemes, sense que el sistema intel·ligent necessiti formar part d'un cos ni estar situat en un entorn real. És a dir, la IA simbòlica opera amb representacions abstractes del món real que es modelen mitjançant llenguatges de representació basats principalment en la lògica matemàtica i les seves extensions. Per aquest motiu, els primers sistemes intel·ligents resolien sobretot problemes que no requerien interactuar directament amb l'entorn, com ara demostrar senzills teoremes matemàtics o jugar a escacs. Això no vol dir que la IA simbòlica no es pugui usar per programar el mòdul de raonament d'un robot físic situat en un entorn real, però en els primers anys de la IA no hi havia llenguatges de representació del coneixement ni de programació que permetessin fer-ho de forma eficient. Actualment, la IA simbòlica se segueix usant per demostrar teoremes o jugar a escacs, però també per a aplicacions que requereixen percebre l'entorn i actuar sobre aquest, com ara l'aprenentatge i la presa de decisions en robots autònoms.

1.6. IA artificial i IA humana

Al seu llibre Python Deep Learning, Jordi Torres acaba la distinció entre IA feble i forta amb la interessant reflexió següent.

Hi ha autors que consideren que fins i tot tan sols amb la intel·ligència artificial feble ens dirigim ràpidament cap a una situació en la qual els sistemes informàtics prendran decisions per nosaltres, i demanen que reflexionem què succeirà quan aquestes sistemes deixin de banda l'estratègia humana en favor d'alguna cosa totalment desconeguda per a nosaltres. Crec que tots i cada un de nosaltres hi hem de pensar.

2. Aprendre a partir d'exemples

Un agent aprèn si millora la seva funcionalitat després de fer observacions sobre el seu entorn. L'aprenentatge pot anar des del més trivial, com per exemple apuntar una llista de la compra, fins al més profund, quan Albert Einstein va inferir una nova teoria de l'univers.

Quan l'agent és una computadora, parlem d'**aprenentatge automàtic**: un ordinador observa unes dades, construeix un **model** en base a aquestes dades i llavors utilitza el model com a hipòtesi sobre el món i com a peça de programari per resoldre problemes.

I per què volem que les màquines aprenguin? Per què no les programem simplement de la forma correcta, per començar? Hi ha dos motius principals.

En primer lloc, els dissenyadors no poden anticipar totes les possibles situacions futures. Per exemple, un robot dissenyat per navegar laberints ha d'aprendre la configuració de cada laberint que troba; un programa per predir el preu de les accions al mercat borsari ha d'aprendre a adaptar-se quan les condicions canvien d'alcistes a baixistes.

En segon lloc, de vegades els dissenyadors no tenen idea de com programar una solució. La majoria de persones poden reconèixer les cares dels seus familiars, però ho fan de forma subconscient, de forma que fins i tot els millors programadors no saben com programar un ordinador per reconèixer cares, llevat de fent servir algorismes d'aprenentatge automàtic.

En aquest lliurament, farem una exposició dels conceptes principals de l'aprenentatge automàtic, seguint el capítol 19 del llibre **Artificial Intelligence, A Modern Approach**, de Norvig i Russell. Quan haurem fet una introducció a Python, al mòdul de Programació d'IA, podrem entrar en l'ús de llibreries, concretament scikit-learn, per treballar en detall els diferents tipus d'aprenentatge automàtic (supervisat, no supervisat, per reforçament). Entrarem en detall amb exemples de codi, seguint el llibre de Sebastian Raschka **Python Machine Learning**.

2.1. Formes d'aprenentatge

Qualsevol component d'un programa agent pot millorar mitjançant aprenentatge automàtic. Les millores, i les tècniques utilitzades per aconseguir-les, depenen dels factors següents:

- Quin component s'ha de millorar.
- Quin coneixement a priori té l'agent, que influeix en el model que construeix.
- Quines dades i retroacció (feedback) sobre aquestes dades està disponible.

Els components dels agents inclouen:

1. Un mapejat directe de les condicions de l'estat actual a les accions.
2. Un mecanisme per inferir les propietats rellevants de l'entorn a partir de la seqüència de percepcions.
3. Informació sobre la forma en què el món evoluciona i sobre els resultats de les possibles accions que l'agent pot prendre.
4. Informació d'utilitat indicant la desitjabilitat dels estats del món.
5. Informació sobre el valor de les accions de l'agent, indicant com de desitjables són aquestes accions.
6. Objectius que descriuen els estats més desitjables.
7. Un generador de problemes, un crític i un element d'aprenentatge que permeten que el sistema millori.

Aquests elements s'entendran millor després d'haver vist el segon lliurament de Models d'IA. De moment, observem que cada un d'aquests components es pot aprendre. Considerem, per exemple, un agent d'un vehicle autopilotat que aprèn a través de l'observació d'un conductor humà. Cada vegada que el conductor frena, l'agent pot aprendre una regla de condició-acció per decidir quan frenar (component 1). Mirant moltes imatges de càmera que se li diu que contenen autobusos, aprèn a reconèixer-los (component 2). Intentant accions i observant-ne els resultats (per exemple, frenar fortament en un terra banyat) pot aprendre els efectes de les seves accions (component 3). A més, quan rep queixes de passatges que han estat estamenejats durant el viatge, pot aprendre un component útil de la seva funció d'utilitat global (component 4).

La tecnologia d'aprenentatge automàtic ha esdevingut una part estàndard de l'enginyeria del programari. Sempre que es construeix un sistema de programari, fins i tot si no està concebut com un agent d'intel·ligència artificial, els components del sistema es poden millorar, potencialment, usant aprenentatge automàtic.

Hi ha diferents exemples de models per als components dels agents: atòmic, factoritzat i relacional, basat en lògica o probabilitat. Hi ha algorismes d'aprenentatge per a tots ells.

En aquest apartat suposam poc coneixement a priori per part de l'agent: comença de zero i aprèn a partir de les dades. Suposam, però, que el disseny del sistema inclou un model que pot aprendre efectivament.

La **inducció** consisteix a anar d'un conjunt específic d'observacions a una regla general: de les observacions que el sol ha sortit cada dia fins ara, induïm que el sol sortirà demà. Això és diferent de la deducció, perquè les conclusions de la inducció poden ser incorrectes, mentre que les conclusions deductives està garantit que són correctes si les premisses són correctes.

Ens centram en problemes en què l'entrada té una **representació factoritzada**: un vector de valors d'atributs. Si no, l'entrada podria ser també **atòmica** o **relacional**.

Quan la sortida pren un valor entre un conjunt finit de valors (com ara assolellat/nuvolós/plujós o cert/fals) el problema d'aprenentatge s'anomena **classificació**. Quan és un nombre (com per exemple la temperatura prevista demà, mesurada sigui com un nombre enter o real) el problema d'aprenentatge rep el nom de **regressió**.

Hi ha tres tipus de **retroacció** (*feedback*) que poden acompanyar les entrades, que determines els tres tipus principals d'aprenentatge.

En **aprenentatge supervisat**, l'agent observa parells entrada-sortida i aprèn una funció que relaciona l'entrada i la sortida. Per exemple, les entrades podrien ser imatges de càmera, cada una acompanyada per una sortida indicant "autobús" o "vianant", etc. Una sortida com aquesta s'anomena **etiqueta**. En el cas d'accions de frenada (component 1 d'abans) una entrada és l'estat actual (velocitat i direcció del cotxe, condicions de la carretera) i una sortida és la distància que ha tardat a frenar. En aquest cas l'agent pot obtenir un conjunt de valors de sortida a partir de les seves pròpies percepcions (a posteriori); l'entorn és el mestre, i l'agent aprèn una funció que relaciona els estats a la distància de frenada.

En **aprenentatge no supervisat** l'agent aprèn patrons de l'entrada sense cap retroacció explícita. La tasca no supervisada més habitual és la **clusterització**: detectar clústers potencialment útils als exemples d'entrada. Per exemple, després d'observar milions d'imatges preses d'Internet, un sistema de visió per computadora pot identificar un gran grup d'imatges similars que podríem associar a l'etiqueta "gat".

En **aprenentatge per reforç** l'agent aprèn d'una sèrie de reforços, positius i negatius: recompenses i càstigs. Per exemple, al final d'una partida d'escacs a l'agent se li diu que ha guanyat (reforç) o que ha perdut (càstig). Es deixa en mans de l'agent decidir quines accions prèvies al reforçament hi tenen més efecte, així com modificar les seves accions per aconseguir millors recompenses en el futur.

2.2. Aprenentatge supervisat

La tasca d'aprenentatge supervisat és la següent.

Donat un conjunt d'entrenament de N exemples de parells entrada-sortida

$$(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$$

on cada parell ha estat generat per una funció desconeguda $y = f(x)$, descobrir una funció h que aproxima la funció f .

La funció h rep el nom d'**hipòtesi** sobre el món. Surt d'un **espai d'hipòtesis** H de possibles funcions. Per exemple, l'espai d'hipòtesis podria ser el conjunt de polinomis de grau 3; o el conjunt de funcions Javascript; o el conjunt de funcions lògiques booleanes.

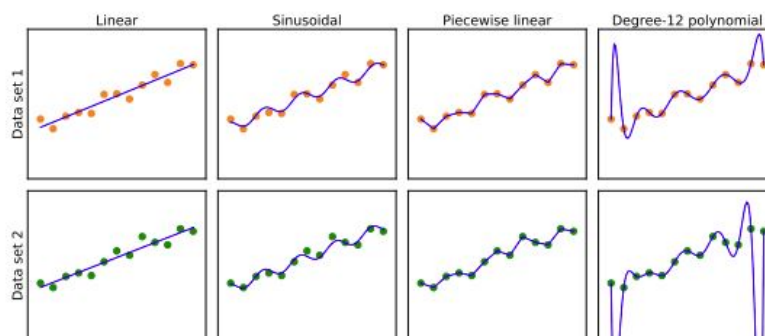
Podem dir que h és un **model** de les dades, tret d'una **classe de models** H , o també podem dir que és una **funció** treta d'una **classe de funcions**. La sortida y_i és el valor de referència (**ground truth**) que la nostra funció mira de predir.

Com triam un espai d'hipòtesis? Pot ser que tenguem qualche coneixement a priori sobre el procés que ha generat les dades. Si no, podem realitzar una **anàlisi exploratòria de les dades**: un examen de les dades amb tests estadístics i visualitzacions per tenir-ne una idea i alguna intuïció sobre quin espai d'hipòtesis podria ser adequat. O simplement podem assajar molts d'espais d'hipòtesis i avaluar quin és el que funciona més bé.

Com triam una bona hipòtesi dins l'espai d'hipòtesis? Podem desitjar una **hipòtesi consistent**: una h tal que cada x_i del conjunt d'entrenament té $h(x_i) = y_i$. Amb sortides contínues això no és possible en principi, en canvi se cerca una funció de millor ajust (*best-fit function*) amb la qual cada $h(x_i)$ sigui proper a y_i , en un sentit que formalitzarem més endavant.

La veritable mesura d'una hipòtesi no és realment com es comporta sobre el conjunt d'entrenament, sinó com és capaç de manejar entrades que encara no ha vist. Això ho podem avaluar amb una segona mostra de (x_i, y_i) que anomenam **conjunt de prova** (*test set*). Deim que h **generalitza** bé si prediu acuradament les sortides del conjunt de prova.

La figura següent mostra que la funció h que l'algorisme d'aprenentatge descobreix depèn de l'espai d'hipòtesis H considerat i del conjunt de dades d'entrenament donat. Cada un dels quatre gràfics de la filera de dalt té el mateix conjunt d'entrenament de 13 punts al pla (x, y) . Els quatre gràfics de la filera de baix tenen un segon conjunt de 13 punts; els dos conjunts són representatius de la mateixa funció desconeguda $f(x)$. Cada columna mostra la hipòtesi de millor ajust h d'un espai d'hipòtesis diferent.



Imatge: Quatre models d'hipòtesi - AIMA

- Columna 1: Línies rectes; funcions de la forma $h(x) = w_1 x + w_0$. Cap recta no pot ser una hipòtesi consistent per als punts de dades. No pot passar per tots ells simultàniament.
- Columna 2: funcions sinusoidals de la forma $h(x) = w_1 x + \sin(w_0 x)$. Aquesta opció no és consistent de tot, però s'ajusta bastant bé als dos conjunts de dades.
- Columna 3: Funcions lineals a trams en què cada segment recte connecta els punts de dos en dos. Aquestes funcions sempre són consistentes.

- Columna 4: Polinomis de grau 12, $h(x) = \sum_{i=0}^{12} w_i x^{i_0}$: Són consistents, sempre podem trobar un polinomi de grau 12 que passi per 13 punts. Però només perquè la hipòtesi sigui consistent no significa que sigui una bona opció.

Una forma d'analitzar els espais d'hipòtesis és pel biaix que imposen (independentment del conjunt de dades) i la variància que produeixen (d'un conjunt d'entrenament a un altre).

Entenem per **biaix** (en anglès, *bias*) la tendència d'una hipòtesi predictiva a desviar-se del valor esperat quan es fa el promig en diferents conjunts de dades. El biaix sovint és una conseqüència de les restriccions imposades per l'espai d'hipòtesis. Per exemple, l'espai d'hipòtesis de les funcions lineals introdueix un biaix fort: només permet funcions que són línies rectes. Si hi ha cap altre patró a les dades que no sigui un creixement o decreixement lineal, la funció lineal no serà capaç de representar-lo. D'altra banda, la funció lineal a trams té un biaix baix, la forma de la funció ve marcada per les dades.

Entenem per **variància** (en anglès, *variance*) la magnitud del canvi en la hipòtesi degut a la fluctuació en les dades d'entrenament. Les dues fileres de la figura anterior representen conjunts de dades que han estat presos a partir de mostres d'una mateixa funció $f(x)$. Els conjunts de dades resulta que són una mica diferents. A les tres primeres columnes, la petita diferència a les dades resulta en una petita diferència a la hipòtesi. A aquesta situació li deim **variància baixa**.

Sovint hi ha un **compromís biaix-variància** (*bias-variance tradeoff*): una tria entre hipòtesis més complexes, de baix biaix (*low bias*) (que s'ajusten bé a les dades, i hipòtesis més simples, de baixa variància (*low variance*) que poden generalitzar més bé. Albert Einstein va dir el 1933 que "l'objectiu suprem de tota teoria és aconseguir els elements bàsics irreductibles tan simples i tan pocs com sigui possible sense renunciar a representar adequadament una sola dada d'experiència". En altres paraules, Einstein recomanava triar la hipòtesi més simple que s'ajusta a les dades. Aquesta idea ja ve del filòsof del segle XIV Guillem d'Ockham. El seu principi que "la pluralitat d'entitats no s'ha de suposar sense necessitat" rep el nom de la **navalla d'Ockham**, ja que "afaita" explicacions dubtoses.

No és fàcil definir la simplicitat. Sembla clar que un polinomi amb només dos paràmetres és més simple que un altre amb tretze paràmetres. No obstant això, quan treballem amb xarxes neuronals veurem que els models d'aprenentatge profund generalitzen prou bé, tot i que siguin molt complexos, amb milers de milions de paràmetres. Així, el nombre de paràmetres no és per si mateix una bona mesura de l'adequació d'un model.

Quina hipòtesi és la millor a la figura anterior dels quatre models? No ho podem saber segur. Si sabéssim que les dades són les visites a una pàgina web, que van creixent i oscil·len segons l'hora del dia, ens podríem decantar pel model sinusoidal. Si sabéssim que les dades no són cíclics però tenen un soroll elevat, podríem optar per la funció lineal.

En ocasions, no volem dir que una hipòtesi és possible o impossible, sinó com és de probable. L'aprenentatge supervisat es pot dur a terme triant la hipòtesi més probable donades les dades.

$$h^* = \operatorname{argmax}_{h \in H} P(h|data)$$

Per la regla de Bayes, això és equivalent a

$$h^* = \operatorname{argmax}_{h \in H} P(data|h)P(h)$$

Aleshores podem dir que la probabilitat a priori $P(h)$ és alta per a un polinomi suau de primer o segon grau i més baixa per a un polinomi de grau 13 amb molts pics. Acceptam funcions amb aspecte inusual quan les dades realment les justifiquen, però les desafavorim assignant-hi una probabilitat baixa a priori.

I per què l'espai d'hipòtesis H no podria ser la classe de tots els programes d'ordinador, o bé totes les màquines de Turing? El problema és que hi ha un compromís entre l'expressivitat d'un espai d'hipòtesis i la complexitat computacional de trobar una bona hipòtesi dins aquest espai. Per exemple, ajustar una línia recta a unes dades és una computació senzilla; ajustar polinomis de grau elevat és una mica més difícil; ajustar màquines de Turing és indecidible. Una segona raó per preferir hipòtesis d'espais simples és que presumiblement volem usar la funció h després d'haver-la apresada. Aleshores, calcular $h(x)$ quan h és una funció lineal és ràpid, mentre que computar un programa d'una màquina de Turing ni tan sols està garantit que acabi.

Per aquestes raons, la major part de la feina en aprenentatge s'ha enfocat a les representacions simples.

Recentment hi ha hagut molt d'interès en l'aprenentatge profund (*deep learning*), en què les representacions no són simples però la computació de $h(x)$ pren un nombre acotat de passes en un maquinari adequat.

El compromís entre expressivitat i complexitat no és senzill: sovint un llenguatge expressiu permet que una hipòtesi simple s'ajusti a les dades, mentre que en un llenguatge menys expressiu qualsevol hipòtesi consistent ha de ser complexa.

3. Aprenentatge automàtic

Crear màquines intel·ligents per transformar dades en coneixement

Sebastian Raschka, al seu llibre Python Machine Learning, descriu l'aprenentatge automàtic (Machine Learning) de la forma següent.

Amb la tecnologia de què disposam actualment, generam un recurs abundant: una gran quantitat de dades **estructurades i no estructurades**. A la segona meitat del segle XX, l'**aprenentatge automàtic** (*machine learning*) evolucionà com un subcamp de la **intel·ligència artificial** que implicava algorismes d'autoaprenentatge que derivaven el coneixement a partir de dades per crear prediccions. En lloc de necessitar l'expertesa humana per per derivar de forma manual les regles, l'aprenentatge automàtic ofereix una alternativa més eficient per capturar el coneixement en dades, crear models a partir d'anàlisis de grans quantitats de dades, millorar gradualment el rendiment dels models predictius i prendre decisions basades en aquestes dades.

L'aprenentatge automàtic no només és cada vegada més important en la investigació de la ciència dels computadors, sinó que fa un paper cada vegada més important en la nostra vida diària. Gràcies a l'aprenentatge automàtic, tenim filtres potents contra el correu no desitjat (*spam*), programari pràctic de reconeixement de veu i text, motors de cerca fiables, programes avançats per jugar a escacs i vehicles autopilotats.

Trobam una exposició més detallada, que inclou una primera descripció dels tres grans tipus d'aprenentatge automàtic, a l'entrada sobre [intel·ligència artificial](#) de la Stanford Encyclopedia of Philosophy.

Una gran part del creixement de l'IA en aplicacions ha estat possible gràcies a la invenció de nous algorismes en el subcamp de l'aprenentatge automàtic. L'aprenentatge automàtic s'ocupa de construir sistemes que millorin el seu rendiment en una tasca quan es donen exemples de rendiment ideal en la tasca, o milloren el seu rendiment amb experiència repetida en la tasca. Els algorismes de l'aprenentatge automàtic s'han utilitzat en sistemes de reconeixement de veu, filtres de correu brossa, sistemes de detecció de frau en línia, sistemes de recomanació de productes, etc.

L'aprenentatge automàtic es pot dividir en tres grans àrees.

Aprenentatge supervisat: una forma d'aprenentatge en què un ordinador intenta aprendre una funció f donats exemples, les dades d'entrenament T , dels seus valors en diversos punts del seu domini. Una tasca de mostra seria intentar etiquetar imatges de cares amb el nom d'una persona. La supervisió en l'aprenentatge supervisat es presenta en forma del valor de la funció $f(x)$ en diversos punts x en alguna part del domini de la funció. Normalment es dona en forma d'un conjunt fix de parells d'entrada i sortida per a la funció. Sigui h la "funció apresada". L'objectiu de l'aprenentatge supervisat és que h coincideixi com sigui possible amb la funció real f en el mateix domini. L'error es defineix normalment en termes d'una funció d'error, sobre les dades d'entrenament T . Són possibles altres formes de supervisió i objectius d'aprenentatge. Per exemple, en l'aprenentatge actiu, l'algoritme d'aprenentatge pot sol·licitar el valor de la funció per a entrades arbitràries. L'aprenentatge supervisat domina el camp de l'aprenentatge automàtic i s'ha utilitzat en gairebé totes les aplicacions pràctiques esmentades anteriorment.

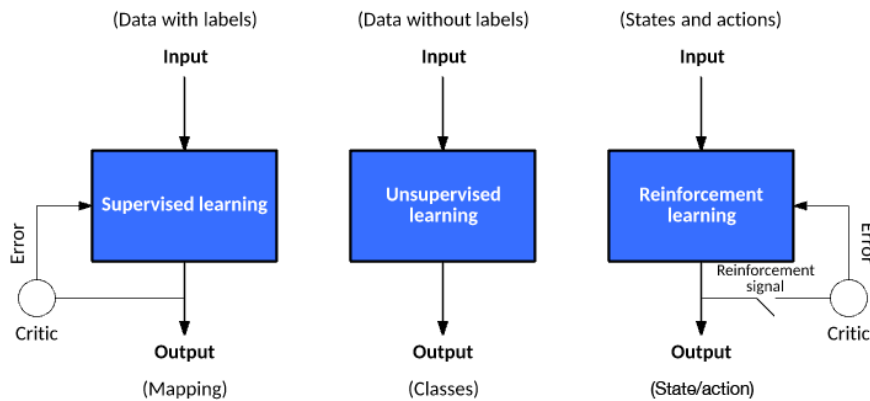
Aprenentatge no supervisat: aquí la màquina intenta trobar coneixement o informació útil quan se li donen algunes dades en brut. No hi ha cap funció associada a l'entrada que s'hagi d'aprendre. La idea és que la màquina ajudi a descobrir patrons o informació interessants que es podrien amagar a les dades. Un ús de l'aprenentatge no supervisat és la **minería de dades**, on es cerquen grans volums de dades per obtenir informació interessant. **PageRank**, un dels primers algorismes utilitzats pel motor de cerca de Google, es pot considerar un sistema d'aprenentatge no supervisat que classifica les pàgines sense cap supervisió humana.

Aprenentatge per reforç: aquí una màquina es deixa anar en un entorn on actua i percep constantment (semblant a la visió de Russell/Hutter anterior) i només de tant en tant rep comentaris sobre el seu comportament en forma de recompenses o càstigs. La màquina ha d'aprendre a comportar-se racionalment a partir d'aquest feedback. Un ús de l'aprenentatge de reforç ha estat en agents de construcció per jugar a jocs d'ordinador. L'objectiu aquí és crear agents que mapein les dades sensorials del joc en cada instant a una acció que ajudi a guanyar en el joc o a maximitzar el gaudi del joc per part d'un jugador humà. En la majoria de jocs, sabem el bé que estem jugant només al final de la partida o només a intervals poc freqüents al llarg de la partida (p. ex., una partida d'escacs que creiem que estem guanyant podria tornar-se ràpidament en contra al final). En l'aprenentatge supervisat, les dades d'entrenament tenen parells d'entrada-sortida ideals. Aquesta forma d'aprenentatge no és adequada per a agents de construcció que han d'operar durant un període de temps i que es jutgen no per una acció sinó per una sèrie d'accions i els seus efectes sobre el medi ambient. L'àmbit de l'aprenentatge per reforç intenta abordar aquest problema mitjançant una varietat de mètodes. Sutton i Barto (1998) ofereixen una introducció completa al camp.

3.1. Tipus d'aprenentatge automàtic

Els sistemes d'aprenentatge automàtic poden pertànyer a una o més de les categories següents segons com aprenen a realitzar prediccions o generar contingut:

- Aprenentatge supervisat
- Aprenentatge no supervisat
- Aprenentatge per reforçament

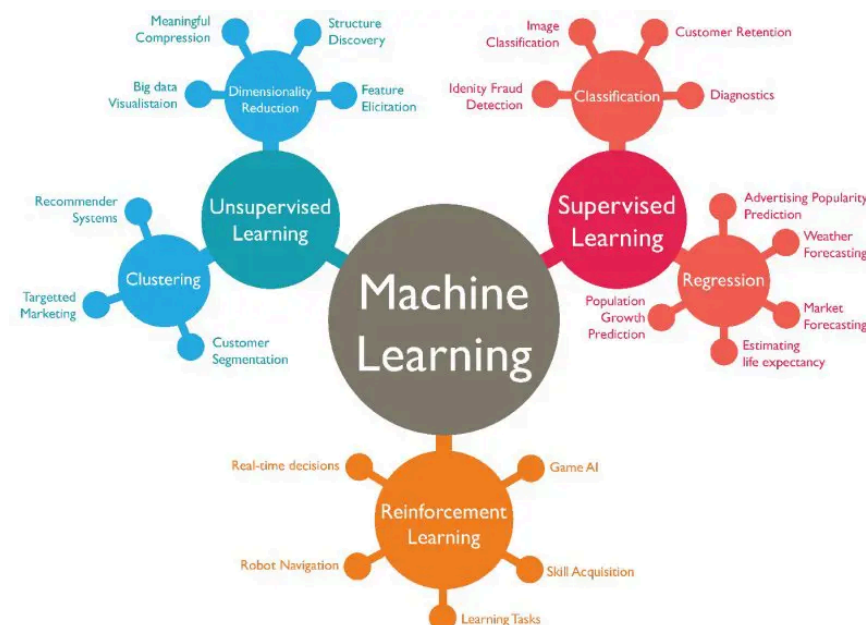


Imatge: <https://developer.ibm.com/articles/cc-models-machine-learning/>

També hi ha fonts que consideren com un tipus d'aprenentatge automàtic, o bé una aplicació de l'aprenentatge automàtic, la **intel·ligència artificial generativa**, orientada a la producció de contingut.

Aplicacions

El diagrama següent il·lustra diverses aplicacions de l'aprenentatge automàtic, classificat segons el tipus: supervisat, no supervisat i de reforç.



Imatge: <https://datasciencedojo.com/blog/machine-learning-101/>

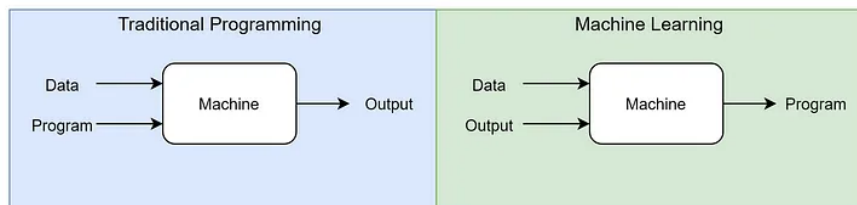
3.2. AA vs. programació tradicional

La diferència fonamental entre el paradigma de programació tradicional i el paradigma d'aprenentatge automàtic (*machine learning*) resideix en la manera com les tasques s'aborden i resolen.

En el paradigma de programació tradicional, els programadors han de crear explícitament un conjunt de regles i instruccions que especifiquin com una màquina o programa ha de processar les dades i quins resultats ha de produir. Aquestes regles són deterministes i es basen en la lògica humana. És com ensenyar-li a una màquina mitjançant un conjunt rígid d'instruccions.

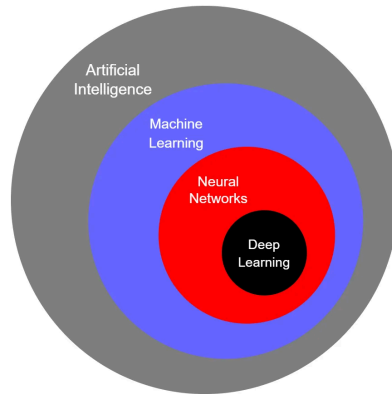
En canvi, en el paradigma d'aprenentatge automàtic, la màquina aprèn a partir de les dades. En lloc de programar explícitament les regles, s'alimenta la màquina amb una gran quantitat de dades i un algoritme d'aprenentatge automàtic analitza aquestes dades per trobar patrons i fer prediccions. La màquina "aprèn" a partir de les dades i millora la seva capacitat de prendre decisions a mesura que s'exposa a més informació.

Així, en el paradigma de programació tradicional, els programadors especifiquen les regles, mentre que en el paradigma d'aprenentatge automàtic, les màquines aprenen a partir de les dades. Això fa que l'aprenentatge automàtic sigui especialment útil en situacions on les regles o els patrons són difícils de definir mitjançant la programació tradicional, com ara la visió per ordinador o el processament del llenguatge natural.



3.3. Aprenentatge profund

Un dels termes més emprats avui en dia, en el context de l'aprenentatge automàtic (machine learning), és el de l'aprenentatge profund (deep learning). Ho veurem més amb detall quan tractem en profunditat les xarxes neuronals. De moment, donem-ne una visió general a través del diagrama següent.



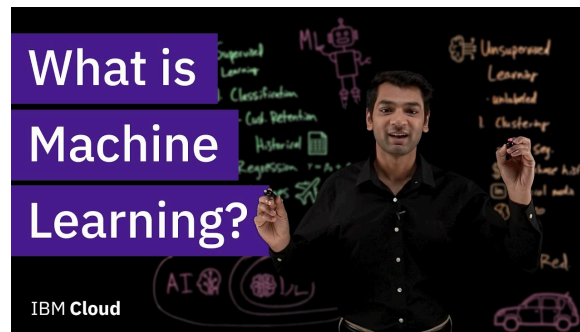
Els conceptes d'Intel·ligència Artificial (IA), Aprenentatge Automàtic (Machine Learning), Xarxes Neuronals (Neural Networks) i Aprenentatge Profund (Deep Learning) estan relacionats i s'inclouen mútuament de la següent manera:

1. **Intel·ligència Artificial (IA):** La Intel·ligència Artificial és el camp més ampli d'estudi que es centra en la creació de sistemes o màquines que poden realitzar tasques que requereixen intel·ligència.
2. **Aprenentatge Automàtic (Machine Learning):** El Machine Learning és una subàrea dins de la IA que se centra específicament en el desenvolupament d'algorismes i models que poden millorar la seva actuació en tasques a través de l'aprenentatge a partir de les dades. És una tècnica fonamental dins de la IA i inclou mètodes com la regressió, classificació, clústering i d'altres.
3. **Xarxes Neuronals (Neural Networks):** Les Xarxes Neuronals són un subconjunt de tècniques de Machine Learning que estan inspirades en la manera en què funcionen les xarxes de neurones en el cervell humà. Aquestes xarxes s'utilitzen per a tasques de processament de dades que requereixen una alta capacitat d'aprenentatge i adaptabilitat, com la visió per computador o el processament de llenguatge natural. Les xarxes neuronals són una part de l'arsenal d'eines utilitzades en el Machine Learning.
4. **Aprenentatge Profund (Deep Learning):** El Deep Learning és una subàrea de les Xarxes Neuronals que fa referència a xarxes neuronals profundes amb més d'una capa oculta entre la capa d'entrada i la capa de sortida. Aquest enfocament ha guanyat molta atenció i èxit en problemes complexos com la visió per computador, el processament de llenguatge natural i la conducció autònoma. El Deep Learning, per tant, és una part específica de les Xarxes Neuronals.

Així, la Intel·ligència Artificial és un camp més ampli que engloba el Machine Learning, que és una tècnica dins de la IA. Les Xarxes Neuronals són una tècnica específica dins del Machine Learning, i el Deep Learning és una subàrea de les Xarxes Neuronals que s'enfoca en xarxes neuronals profundes.

3.4. Vídeo d'aprenentatge automàtic

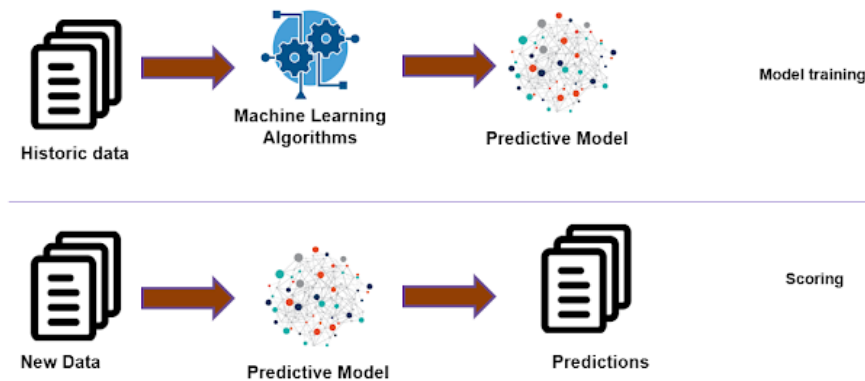
Després d'aquesta primera exposició de l'aprenentatge automàtic, és bon moment per veure aquest vídeo oferit per IBM. S'hi poden traduir els subtítols automàticament.



4. Aprenentatge supervisat

Fer prediccions sobre el futur amb l'aprenentatge supervisat

L'objectiu de l'aprenentatge supervisat és aprendre un model, a partir de dades d'entrenament etiquetades, que permeten fer prediccions sobre dades futures o no vistes. Aquí, el terme supervisat es refereix a un conjunt de mostres en què els senyals de sortida desitjats (etiquetes) ja es coneixen.



Considerant l'exemple del filtre de correu no desitjat, podem entrenar un model utilitzant un algorisme d'aprenentatge supervisat en un corpus de correus electrònics etiquetats (com a *spam* o no *spam*) per predir si un nou correu electrònic pertany a una categoria o a l'altra. Una tasca d'aprenentatge supervisat amb etiquetes de classe discreta és una tasca de **classificació**.

Una altra subcategoria de l'aprenentatge supervisat és la **regressió**, en què el senyal de sortida és un valor continu. Encara que el senyal de sortida sigui, per exemple, un enter de 8, 16 o més bits i sigui, per tant, discret, sempre que correspongui a una digitalització d'un senyal analògic del món real és pràctic seguir-lo considerant com un valor continu, encara que estrictament no ho sigui.

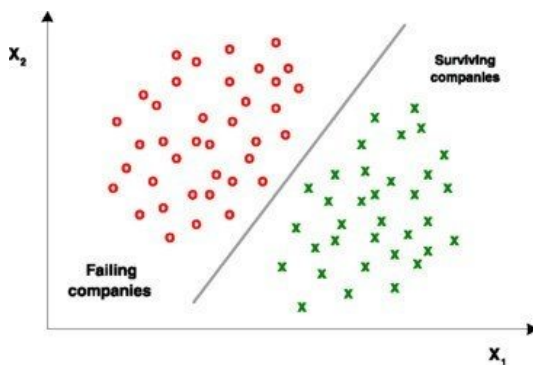
4.1. Classificació

Classificació per predir etiquetes de classe

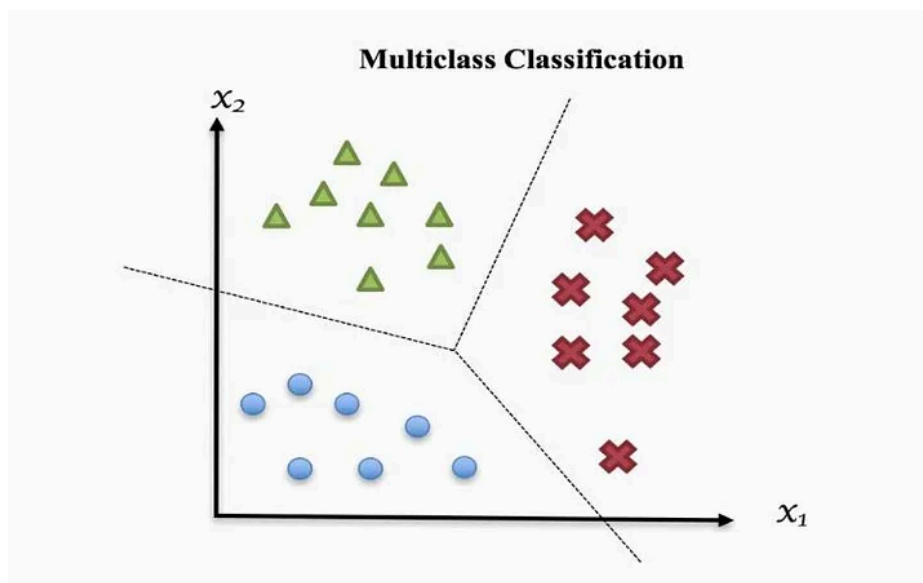
La classificació és una subcategoria de l'aprenentatge supervisat que té l'objectiu de predir les etiquetes de classe categòriques de noves instàncies, basades en observacions passades. Aquestes etiquetes de classe són discretes, valors desordenats que es poden entendre com els grups als quals pertanyen les instàncies. L'exemple de detecció de correu no desitjat és un exemple típic de tasca de **classificació binària**, en què l'algorisme d'aprenentatge automàtic aprèn una regla per distingir entre dues possibles classes: missatges que són correu desitjat o no.

No obstant això, el conjunt d'etiquetes de classe no té per què ser de naturalesa binària. El model predictiu que s'ha après mitjançant un algorisme d'aprenentatge supervisat pot assignar qualsevol etiqueta de classe que hagi aparegut en el conjunt de dades d'entrenament a una nova instància sense etiqueta. Un exemple típic d'una tasca de **classificació multiclasse** és el reconeixement de dígitos manuscrits. Aquí, podem recollir un conjunt de dades d'entrenament que consisteix en múltiples exemples manuscrits de cada lletra de l'alfabet. Ara, si un usuari dona un nou caràcter manuscrit des d'un dispositiu d'entrada, el nostre model predictiu serà capaç de predir la lletra correcta de l'alfabet amb una certa precisió.

La següent figura il·lustra el concepte d'una tasca de classificació binària; una part d'aquesta mostra estan etiquetades com a classe positiva (símbol **x**) i una altra com a classe negativa (símbol **o**). En aquest cas, el conjunt de dades és bidimensional, cosa que significa que cada mostra té dos valors com a variables d'entrada: x_1 i x_2 . Ara, podem usar un algorisme d'aprenentatge automàtic supervisat per aprendre una regla -el límit de decisió està representat amb una línia contínua- que pot separar les dues classes i classificar noves dades dins cada categoria a partir dels seus valors de x_1 i x_2 .



Els problemes de classificació en què hi ha més de dues categories reben el nom de **multiclasse**. A la figura tenim un exemple de classificació en tres categories.



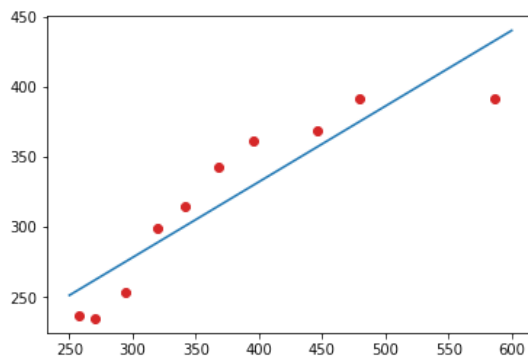
4.2. Regressió

Regressió per predir resultats continus

A la secció anterior hem vist que la tasca de classificació consisteix a assignar etiquetes categòriques i sense ordre a instàncies. Un segon tipus d'aprenentatge supervisat és la predicció de resultats continus, també conegut com a **anàlisi de regressió**. En aquest cas tenim un nombre de variables predictores (explicatives) i una variable de resposta contínua (**resultat** o **destinació**), i hem de trobar una relació entre aquestes variables que permeti predir un resultat.

El terme **regressió** fou ideat per **Francis Galton** al seu article *Regression Towards Mediocrity in Hereditary Stature* el 1886. Galton va descriure el fenomen biològic segons el qual la variació d'alçada en una població no augmenta amb el temps. Ell observà que l'alçada dels pares no passa la dels fills, però que, en canvi, l'alçada dels fills convergeix cap a la mitjana de la població.

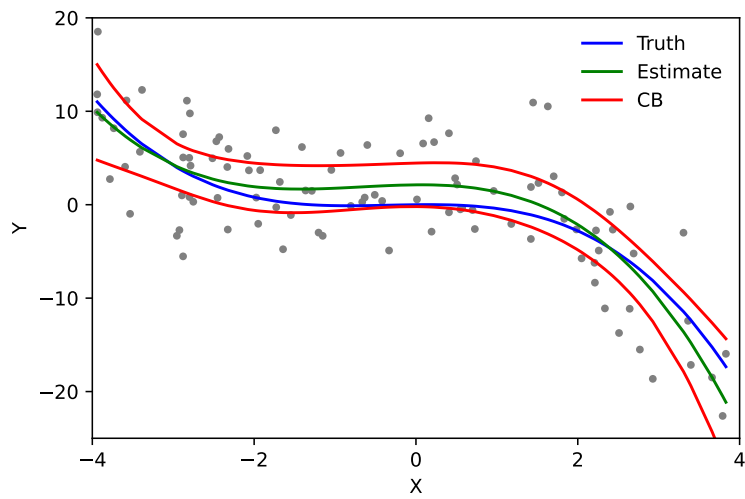
La figura següent il·lustra el concepte de regressió lineal. Donada una variable predictora x i una resposta y , ajustam una línia a aquestes dades, que minimitza la distància entre els punts de la mostra i la línia del model. Habitualment aquesta distància és la distància quadràtica mitjana. Després d'ajustar el model lineal, ja es pot utilitzar l'ordenada a l'origen i el pendent de la recta ajustada per predir la variable resultat y per a una nova dada x .



La regressió lineal que hem vist a l'apartat anterior només considera relacions lineals entre les variables. Es pot ampliar utilitzant la regressió polinòmica, que modelitza la relació entre dues variables mitjançant un polinomi (una funció matemàtica amb termes elevats a potències positives enteres) com a funció predictiva.

En la regressió polinòmica, es busca ajustar un polinomi a les dades observades per predir o modelar una variable dependent en funció d'una variable independent. Aquest polinomi pot tenir diversos graus, com ara lineal, quadràtic (grau 2), cúbic (grau 3) i així successivament. A major grau del polinomi, més complexa serà la forma de la relació modelitzada.

La regressió polinòmica permet capturar patrons no lineals en les dades i adaptar-se millor a situacions en què la relació entre les variables no és estrictament lineal. Tanmateix, cal tenir precaució amb graus alts de polinomis, ja que poden conduir a sobreajustament (*overfitting*) i resultats menys interpretables. Per tant, és important seleccionar el grau òptim del polinomi per a cada conjunt de dades mitjançant tècniques com la validació creuada. Ho veurem amb més detall al lliurament 3, sobre regressió.



5. Aprenentatge no supervisat

Descobrir estructures ocultes amb l'aprenentatge no supervisat

En l'**aprenentatge supervisat**, quan entrenam el model sabem la resposta correcta anticipadament.

En l'**aprenentatge per reforçament** definim una mesura de recompensa per a accions futures mitjançant l'agent.

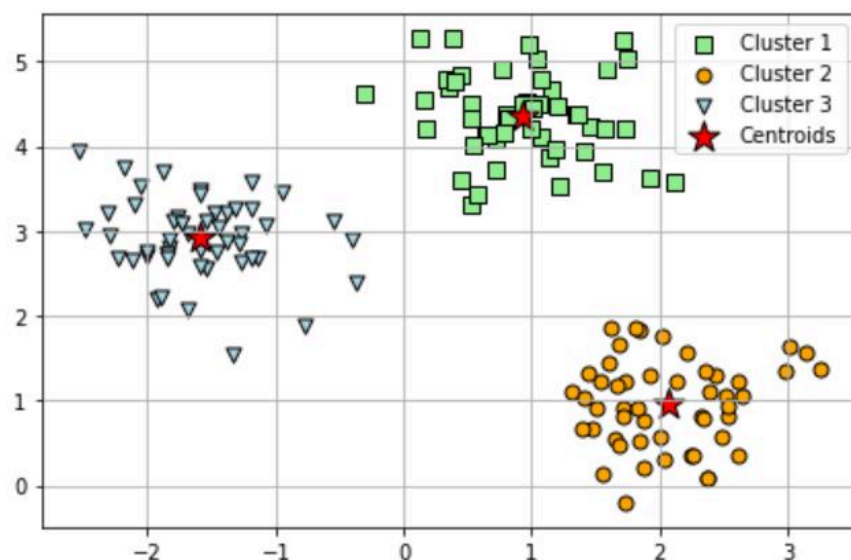
En canvi, en l'**aprenentatge no supervisat** tractam dades sense etiquetar o dades d'estructura desconeguda. Amb les tècniques d'aprenentatge no supervisat, podem explorar l'estructura de les nostres dades per extraure informació significativa sense l'ajuda d'una variable resultat coneguda ni d'una funció de recompensa.

5.1. Anàlisi de clústers

Trobar subgrups amb l'agrupament

L'**agrupament** o **clústering** és una tècnica exploratòria d'anàlisi de dades que ens permet organitzar molta informació en subgrups significatius (clústers) sense tenir cap coneixement previ dels membres del grup. Cada clúster que sorgeix durant l'anàlisi defineix un grup d'objectes que comparteixen un cert grau de semblança, però són diferents dels objectes dels clústers. Per això l'agrupament de vegades rep el nom de **classificació sense supervisió**. El clústering és una tècnica excel·lent per estructurar informació i obtenir relacions significatives entre les dades. Per exemple, permet als venedors descobrir grups de clients basats en els seus interessos, amb la finalitat de desenvolupar programes de màrqueting exclusius.

La figura següent mostra com es pot aplicar l'agrupament per organitzar dades sense etiquetar en tres grups diferents, basats en la semblança de les seves característiques x_1 i x_2 .

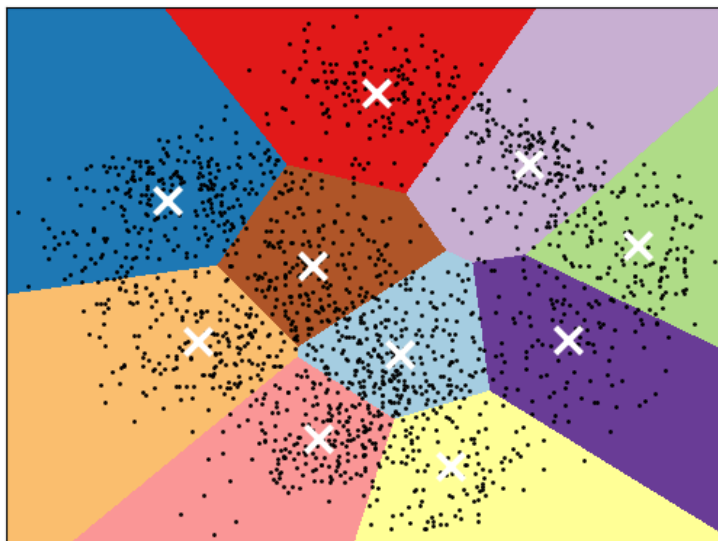


5.2. Algorismes de clústering

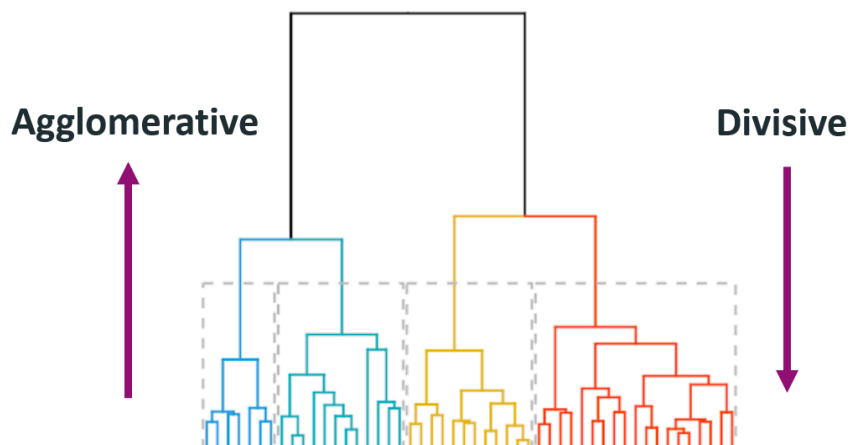
Hi ha una diversitat d'algorismes de clústering. Vegem ara una breu descripció dels tres algorismes que tractarem en el mòdul: k-means, clústering jeràrquic i DBSCAN.

K-means és un algorisme de clústering àmpliament utilitzat que té com a objectiu agrupar les dades en un nombre predeterminat de clústers. L'objectiu principal d'aquest algorisme és minimitzar la variància dins de cada clúster, és a dir, trobar els clústers que maximitzen la similitud entre les dades dins del mateix grup. L'algorisme funciona inicialment seleccionant aleatòriament els centroides inicials per a cada clúster i, després, assigna cada punt de dades al clúster més proper al seu centroide. Aquest procés es repeteix iterativament fins que es convergeix cap a una solució òptima. K-means és eficient i efectiu en la seva simplicitat, però requereix que el nombre de clústers sigui conegut prèviament, el que pot ser una limitació en algunes aplicacions.

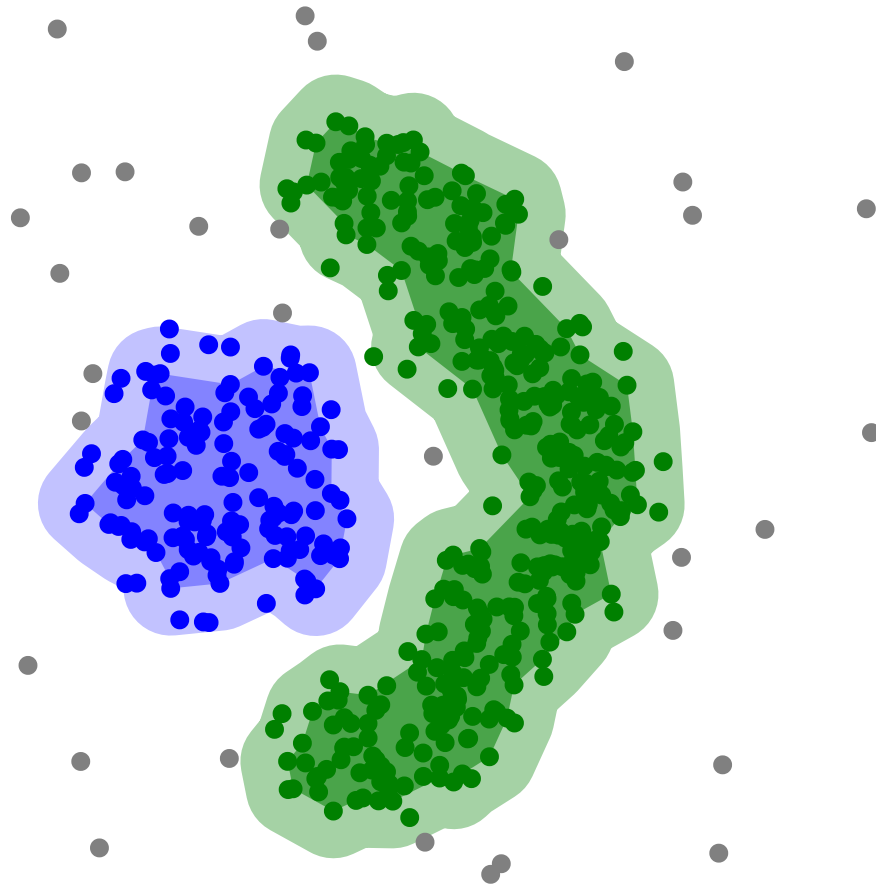
K-means clustering on the digits dataset (PCA-reduced data)
Centroids are marked with white cross



L'**agrupació jeràrquica** és un altre mètode de clústering que organitza les dades en una estructura jeràrquica d'arbre o dendrograma. Aquesta tècnica no requereix predefinir el nombre de clústers i ofereix una visió més granular de les agrupacions. Inicialment, cada punt de dades es considera un clúster individual, i després s'agrupen successivament en clústers més grans en funció de la similitud entre ells. Els dos mètodes principals de l'agrupació jeràrquica són l'agrupació aglomerativa i l'agrupació divisiva. L'agrupació aglomerativa comença amb clústers individuals i els fusiona gradualment, mentre que l'agrupació divisiva comença amb un sol clúster gran i el divideix en clústers més petits. Aquesta flexibilitat fa que l'agrupació jeràrquica sigui útil per explorar diferents nivells de granularitat en les dades.



DBSCAN (Density-Based Spatial Clustering of Applications with Noise) és un algorisme de clústering que es basa en la densitat de les dades. En lloc de predefinir el nombre de clústers, DBSCAN troba clústers identificant zones amb alta densitat de punts i separant aquestes zones d'altres regions menys densament poblades. Els punts de dades es classifiquen com a nuclis, fronteres o soroll segons la seva proximitat a altres punts. Aquest algorisme és robust enfront de soroll i pot identificar clústers de formes irregulars. No requereix que es conegui el nombre de clústers prèviament i pot identificar clústers de diferents formes i mides.



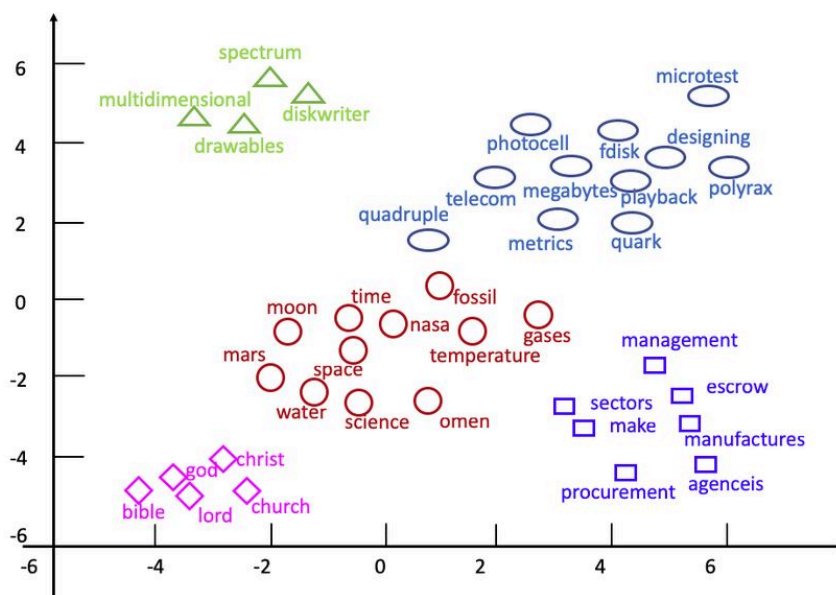
5.3. Reducció de la dimensionalitat

Reducció de dimensionalitat per comprimir dades

Un altre camp de l'aprenentatge no supervisat és la **reducció de la dimensionalitat**. Moltes vegades es treballa amb dades de dimensionalitat elevada (cada observació té un nombre gran de mesures, cada punt és descrit per moltes variables d'entrada $x_1, x_2 \dots x_N$). Això pot suposar un repte per l'espai d'emmagatzematge limitat i el rendiment computacional dels algorismes d'aprenentatge automàtic. La reducció de dimensionalitat sense supervisió és un enfocament usat amb freqüència en el preprocesament de característiques per eliminar soroll de les dades; també pot degradar el rendiment predictiu de determinats algorismes i comprimir les dades en un subespai dimensional més petit, mentre que es manté la major part de la informació important.

De vegades, la reducció de dimensionalitat pot servir per visualitzar dades; per exemple, un conjunt de dades en moltes dimensions es pot projectar en un espai de característiques d'una, dues o tres dimensions per poder veure-les amb gràfics de dispersió o histogrames 2D o 3D.

A la figura següent, com a exemple, podem veure la proximitat entre paraules segons el seu significat, després d'una projecció bidimensional de les seves característiques multidimensionals.

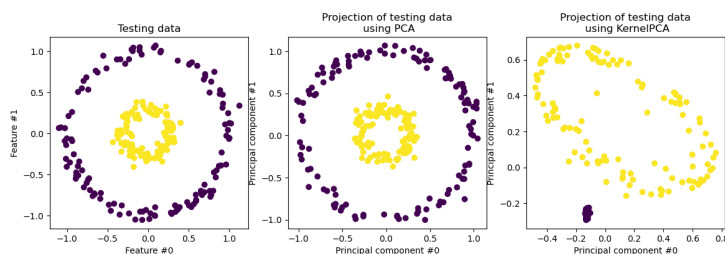


5.4. PCA i PCA amb kernel

PCA (*Principal Component Analysis*, anàlisi de components principals) és una tècnica àmpliament utilitzada per reduir la dimensionalitat de conjunts de dades grans mentre es manté la major part de la informació original. La idea principal darrere del PCA és transformar les dades originals en un nou conjunt de variables, conegudes com a components principals, que són combinacions lineals de les variables originals. Aquests nous components principals estan ordenats per la quantitat de variància que capturen, de manera que els primers components principals contenen la major part de la variància del conjunt de dades. Mitjançant la selecció d'un nombre adequat de components principals, es pot reduir significativament la dimensionalitat del conjunt de dades sense perdre molta informació. PCA és eficaç quan les dades tenen correlacions lineals significatives entre les variables originals.

D'altra banda, el PCA amb nucli (kernel PCA) és una extensió del PCA que permet manejar dades que no són separables linealment en l'espai original. En lloc de treballar amb les variables originals, el kernel PCA aplica una funció de nucli als punts de dades per projectar-los a un espai d'alta dimensionalitat on les dades poden ser separades de manera més efectiva. A continuació, s'aplica el PCA convencional en aquest espai de nucli per reduir la dimensionalitat. Aquesta tècnica és particularment útil quan les dades tenen estructures complexes i no lineals. Els nuclis més comuns utilitzats en el kernel PCA inclouen el kernel polinòmic i el kernel radial (RBF).

A la figura, podem observar com l'extensió de kernel PCA aconsegueix separar linealment unes dades que en l'espai original no s'hi podien separar.



5.5. Aprenentatge supervisat vs. no supervisat

La presentació següent exposa la diferència entre aprenentatge supervisat i no supervisat i ens ajuda a reforçar els conceptes ja exposats. Hi ha disponible la traducció automàtica dels subtítols, originalment en anglès.

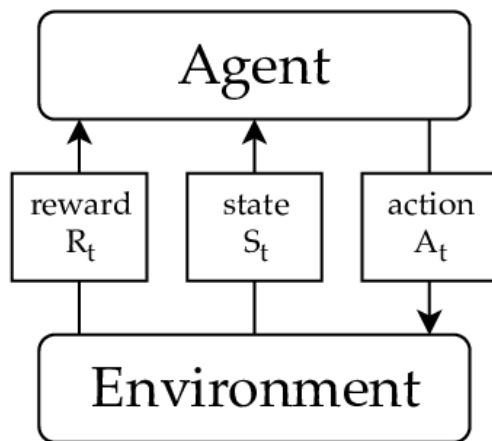


6. Aprenentatge per reforç

Resoldre problemes interactius amb aprenentatge per reforç

Un tercer tipus d'aprenentatge automàtic és l'**aprenentatge per reforç**. En aquest tipus d'aprenentatge, l'objectiu és desenvolupar un sistema (**agent**) que millori el seu rendiment basat en interaccions amb l'entorn. Com que la informació sobre l'estat actual de l'entorn sol incloure un **senyal de recompensa**, això pot fer pensar que està relacionat amb l'aprenentatge supervisat. Tanmateix, en l'aprenentatge per reforç aquesta retroacció o *feedback* no és cap valor ni etiqueta correcta, sinó una mesura d'avaluació de l'acció mitjançant una funció de recompensa. Interactuant amb l'entorn, un agent pot usar l'aprenentatge per reforç per aprendre una sèrie d'accions que maximitzin aquesta recompensa, mitjançant assaig-error o una planificació deliberada.

Un exemple conegut d'aprenentatge per reforç és el motor d'escacs. Aquí, l'agent tria entre una sèrie de moviments segons l'estat del tauler (l'entorn), i la recompensa es pot definir com **guanyar** o **perdre** al final del joc.



Hi ha diferents subtipus d'aprenentatge per reforç. Tanmateix, un esquema general és que l'agent en aprenentatge per reforç intenta maximitzar la recompensa mitjançant una sèrie d'interaccions amb l'entorn. Cada estat pot anar associat a una recompensa positiva o negativa, i una recompensa es pot definir com l'assoliment d'un objectiu general (com ara guanyar o perdre una partida d'escacs). Per exemple, en escacs, el resultat de cada moviment podria ser un estat distint de l'entorn. Per explorar una mica més l'exemple dels escacs, pensem en certes jugades del tauler associades a un esdeveniment positiu (per exemple, eliminar una peça de l'adversari o amenaçar la reina). Això no obstant, unes altres jugades estan associades a un esdeveniment negatiu (com ara perdre una peça per a l'adversari en el següent torn). Ara bé, no tots els torns donen com a resultat l'eliminació d'una peça del tauler, i l'aprenentatge per reforç se centra a aprendre les sèries de passes maximitzant una recompensa basada en la retroacció immediata i diferida.

6.1. Aprendre de les recompenses

L'**aprenentatge per reforçament** (RL, *Reinforcement Learning*) és un paradigma d'aprenentatge automàtic que implica que un agent prengui decisions seqüencials en un entorn per aconseguir un objectiu. L'agent aprèn a prendre decisions mitjançant la interacció amb l'entorn i rebent recompenses o penalitzacions per les seves accions. Hi ha dues aproximacions principals a l'aprenentatge per reforçament: l'aprenentatge basat en models (Model-based RL) i l'aprenentatge sense model (Model-free RL).

Aprenentatge Basat en Models (Model-based RL):

En l'aprenentatge basat en models, l'agent construeix un model intern de com funciona l'entorn. Aquest model intern li permet simular les conseqüències de les seves accions sense necessitat de provar-les realment en l'entorn real. Això permet a l'agent prendre decisions més informades i planificar les seves accions per aconseguir el seu objectiu. L'agent utilitza el seu model intern per simular diferents estratègies i seleccionar la millor basant-se en les previsions de recompensa. Tot i que l'aprenentatge basat en models pot ser més eficient en termes de mostres que l'aprenentatge sense model, requereix un bon model de l'entorn, el que pot ser un repte en entorns complexos.

Aprenentatge Sense Model (Model-free RL):

En l'aprenentatge sense model, l'agent no construeix un model explícit de l'entorn i aprèn a prendre decisions basant-se únicament en la seva experiència prèvia. L'agent explora l'entorn i ajusta les seves accions segons les recompenses rebudes directament. Aquest enfocament és més senzill i pot ser més adequat per entorns on és difícil construir un model precís. Els algorismes de Model-free RL, com ara Q-learning i Sarsa, aprenen funcions d'avaluació de política o funcions d'acció-valor per determinar la millor manera de procedir sense necessitat de comprendre l'entorn.

En resum, l'aprenentatge per reforçament és un paradigma d'aprenentatge que involucra agents que prenen decisions per aconseguir recompenses en un entorn. L'aprenentatge basat en models implica la construcció d'un model intern de l'entorn per a la planificació, mentre que l'aprenentatge sense model es basa en l'experiència directa per aprendre com actuar. L'elecció entre aquests dos enfocaments depèn de la naturalesa de l'entorn i els recursos disponibles.

6.2. Aprenentatge per reforç passiu

L'**aprenentatge per reforç passiu** (*Passive Reinforcement Learning*) és una branca de l'aprenentatge de reforç que se centra en l'aprenentatge a partir de dades històriques o d'experiències prèvies sense prendre decisions actives. En aquest context, hi ha tres mètodes clau utilitzats: *Direct Utility Estimation*, *Adaptive Dynamic Programming* i *Temporal-Difference Learning*.

Direct Utility Estimation

Direct Utility Estimation (DUE) és un mètode passiu que cerca estimar directament les funcions d'avaluació de política o les funcions d'acció-valor sense que calgui una política de presa de decisions activa. En lloc d'això, utilitza dades històriques, com ara registres d'interacció passada, per estimar les funcions de valor. Aquest mètode és útil quan es disposa de dades acumulades a través de l'experiència prèvia i es vol aprofitar aquesta informació per millorar les decisions futures.

Adaptive Dynamic Programming

Adaptive Dynamic Programming (ADP) és un enfocament que combina elements de l'Aprenentatge per Reforçament amb la Programació Dinàmica per adaptar les funcions de valor mitjançant l'ús d'una política fixa. Aquesta tècnica utilitza les dades històriques per ajustar gradualment les funcions de valor sense canviar la política actual. És especialment útil quan es vol mantenir una política prèviament establerta i millorar-la mitjançant les dades existents sense canviar-la radicalment.

Temporal-Difference Learning

Temporal-Difference (TD) Learning és un mètode d'aprenentatge per reforç passiu que es basa en l'ús de l'error temporal (TD error) per actualitzar les estimacions de les funcions de valor. Aquest enfocament permet aprendre de forma incremental a mesura que l'agent interactua amb l'entorn. El mètode TD utilitza l'error temporal per ajustar les estimacions de les funcions de valor i millorar la qualitat de la política de l'agent sense haver de tenir un model explícit de l'entorn.

6.3. Aprenentatge per reforç actiu

L'aprenentatge per reforç actiu se centra en l'aprenentatge mitjançant la presa activa de decisions per aconseguir recompenses.

Exploració

En l'aprenentatge de reforç, l'exploració es refereix a la pràctica d'investigar noves opcions i accions amb l'objectiu d'adquirir coneixement sobre l'entorn i millorar la política de presa de decisions de l'agent. L'exploració és essencial per evitar que l'agent quedi atrapat en polítiques subòptimes i per descobrir les millors accions per aconseguir recompenses més altes. L'ús de mètodes d'exploració, com ara ϵ -greedy o UCB (*Upper Confidence Bound*), ajuda a trobar un equilibri entre l'**exploració** i l'**explotació** de les opcions conegudes.

Exploració segura

Amb l'exploració segura es pretén que l'agent explorador no prengui accions perilloses o desconegudes que puguin causar danys o violar restriccions de seguretat. Aquesta és una preocupació important en aplicacions pràctiques com la robòtica o la conducció autònoma, on les decisions errònies poden tenir conseqüències greus. L'exploració segura implica el desenvolupament de polítiques d'exploració que considerin la seguretat com a prioritat i l'ús de tècniques específiques com per exemple la planificació segura.

Temporal-Difference Q-learning

Temporal-Difference Q-learning és un mètode d'aprenentatge de reforçament actiu que s'utilitza per aprendre una funció d'acció-valor (Q-funció) que estima la recompensa esperada d'executar una acció en un estat i seguir una política determinada. Aquest mètode utilitza errors temporals (TD errors) per actualitzar les estimacions de la Q-funció i millorar progressivament la política de presa de decisions de l'agent. Q-learning és conegut per la seva capacitat per gestionar l'exploració i l'explotació eficaçment, ja que aprèn a partir de l'experiència prèvia mentre explora noves opcions.

6.4. Aplicacions

Les aplicacions de l'aprenentatge per reforç són diverses i abasten una àmplia gamma de dominis. Dos dels àmbits més destacats en els quals aquesta tecnologia ha demostrat un gran èxit són els jocs i el control robòtic.

Jocs

L'aprenentatge de reforç ha assolit un gran èxit en el món dels jocs, especialment en jocs d'estratègia i videojocs.

1. **Jocs d'Atari**, com ara Breakout i Space Invaders.
2. **Jocs de taula**: Els algorismes d'aprenentatge per reforç s'han utilitzat per aprendre a jugar a jocs com el pòquer, les dames, els escacs o el Go, i fins i tot han derrotat a campions mundials en molts d'aquests.

Control robòtic

L'ús de l'aprenentatge de reforç en el control robòtic ofereix moltes oportunitats per millorar la capacitat de manipulació i navegació dels robots. Algunes aplicacions notables inclouen:

1. **Robòtica industrial**: En entorns de fabricació i producció, els robots poden aprendre a dur a terme tasques de muntatge, soldadura o embalatge de manera més eficient utilitzant l'aprenentatge per reforçament per optimitzar els seus moviments.
2. **Manipulació d'objectes**: Els robots poden utilitzar l'aprenentatge de reforç per aprendre com agafar i manipular objectes de manera eficaç, adaptant-se a canvis en la forma o la ubicació dels objectes.
3. **Navegació autònoma**: En aplicacions de robòtica mòbil, com els robots de neteja o les entregues amb drons, l'aprenentatge de reforç ajuda els robots a navegar de manera segura i eficient en entorns desconeguts.
4. **Exoesquelets i pròtesis**: L'aprenentatge per reforç s'utilitza per millorar la funcionalitat de les pròtesis i els exoesquelets, i això permet als usuaris restaurar o augmentar la seva mobilitat i control.

7. IA generativa

La **intel·ligència artificial generativa** representa un dels avenços més espectaculars i transformadors en el camp de la tecnologia moderna. Aquesta branca de la IA ha captat l'atenció tant d'experts com del públic general per la seva capacitat de crear contingut nou i original, imitant i sovint superant la creativitat humana en diversos àmbits.

La IA generativa comprèn un conjunt de models que creen contingut a partir d'una entrada d'usuari o *prompt*. Per exemple, la IA generativa pot crear imatges noves, composicions musicals o acudits; pot resumir articles, explicar com fer una tasca o editar una fotografia.

La IA generativa pot prendre una varietat d'entrades i crear una diversitat de sortides, com ara text, imatges, àudio i vídeo. També pot prendre com a entrada i generar com a sortida una combinació d'aquestes modalitats. Per exemple, un model pot prendre una imatge com a entrada i crear una imatge i text com a sortida, o prendre una imatge i text com a entrada i crear un vídeo com a sortida.

Es poden classificar els models generatius segons les seves entrades i sortides; això s'escriu habitualment com a "tipus d'entrada" a "tipus de sortida". Per exemple, a continuació tenim una llista parcial d'algunes entrades i sortides de models generatius.

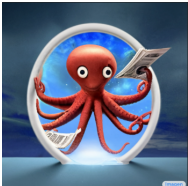

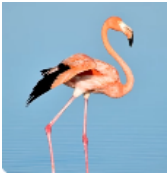
- Text a text
- Text a imatge
- Text a vídeo
- Text a codi
- Text a parla
- Imatge i text a imatge

La IA generativa es diferencia d'altres tipus d'IA en diversos aspectes clau:

1. **Objectiu principal:** Mentre que molts sistemes d'IA estan dissenyats per analitzar, classificar o predir basant-se en dades existents, la IA generativa té com a objectiu principal crear nou contingut.
2. **Creativitat:** La IA generativa pot produir resultats originals i creatius, a diferència dels sistemes d'IA més tradicionals que solen seguir regles predefinides.
3. **Complexitat:** Els models generatius solen ser més complexos i requereixen més potència de càlcul que altres tipus de sistemes d'IA, ja que han d'aprendre i replicar patrons molt sofisticats.
4. **Versatilitat:** La IA generativa pot aplicar-se a una àmplia gamma de dominis creatius, des de l'escriptura fins a la composició musical, passant per la generació d'imatges i el disseny.
5. **Interacció:** Molts sistemes de IA generativa, com els xatbots avançats, poden mantenir converses interactives i adaptar-se a les entrades de l'usuari en temps real.

7.1. Exemples

La taula següent mostra diversos exemples de models generatius, la seva entrada i un exemple de la seva possible sortida.

Model	Entrada	Sortida d'exemple
Text a text (Bard)	Who invented the Le Mans race?	The 24 Hours of Le Mans was founded by the Automobile Club de l'Ouest (ACO), a motoring organization founded in 1906. The first race was held on 26 and 27 May 1923 on the Sarthe Circuit, a 13.426-kilometre (8.345-mile) road circuit located near the town of Le Mans in France. The race was originally conceived as a way to promote the ACO and the Sarthe region.
Text a imatge (Imagen)	An alien octopus floats through a portal reading a newspaper	
Text a vídeo (Phenaki)	A photorealistic teddy bear is swimming in the ocean at San Francisco. The teddy bear goes under water. The teddy bear keeps swimming under the water with colorful fishes. A panda bear is swimming under water.	
Text a codi	Write a Python loop that loops over a list of numbers and prints the prime numbers.	<pre>for number in numbers: # Check if the number is prime. is_prime = True for i in range(2, number): if number % i == 0: is_prime = False break # If the number is prime, print it. if is_prime: print(number)</pre>
Imatge a text (Google DeepMind)		This is a flamingo. They are found in the Caribbean.

Taula: Exemples d'IA generativa de diferents modalitats

Imatges i ampliació: https://developers.google.com/machine-learning/intro-to-ml/what-is-ml#generative_ai

7.2. Sistemes de diferents empreses

Tots els exemples de l'apartat anterior es basaven en sistemes de Google. Hi ha, però, un grapat d'empreses competitives en aquesta tecnologia. Aquí n'oferim un quadre com a resum.

Empresa	Sistema	Modalitat
OpenAI	GPT-4	Text i imatges
OpenAI	DALL-E	Text a imatge
Google	Gemini	Text i imatges
Google	Imagen	Text a imatge
Anthropic	Claude	Text
Stability AI	Stable Diffusion	Text a imatge
Meta	LLaMA	Text
Microsoft	VALL-E	Text a veu
DeepMind	AlphaFold	Estructura de proteïnes
Midjourney	Midjourney	Text a imatge
Runway	Gen-2	Text a vídeo
ElevenLabs	ElevenLabs	Text a veu
GitHub	Copilot	Generació de codi
Amazon	CodeWhisperer	Generació de codi
NVIDIA	Audio2Face	Animació facial des d'àudio

Taula: Sistemes d'IA generativa

Aquesta taula mostra una varietat de sistemes d'IA generativa desenvolupats per diferents empreses, abastant diverses modalitats com text, imatge, veu, vídeo i codi. Hem de tenir en compte que el camp de la IA generativa evoluciona ràpidament, amb noves eines i actualitzacions que apareixen contínuament.

7.3. Com funciona la IA generativa?

A alt nivell, els models generatius aprenen patrons a les dades amb l'objectiu de produir dades noves però semblants. Els models generatius es poden comparar amb les següents activitats humanes.

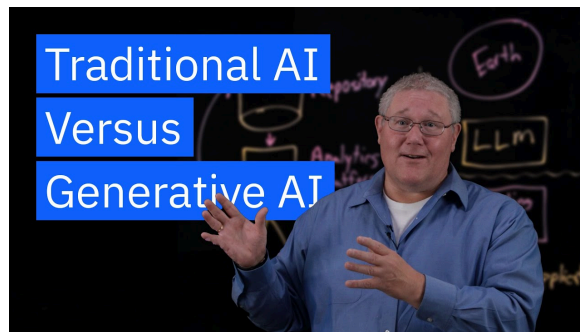
- Còmics que aprenen a imitar altres persones a través de l'observació de la conducta i la manera de parlar
- Artistes que aprenen a pintar en un estil determinat estudiant moltes obres d'aquell estil
- Grups musicals de versions que aprenen a sonar com un grup determinat escoltant molta de música d'aquell grup

Per produir sortides úniques i creatives, els models generatius s'entrenen inicialment usant una aproximació no supervisada, en què el model aprèn a imitar les dades sobre les quals s'entrena. De vegades, el model se segueix entrenant usant entrenament supervisat o per reforçament sobre dades concretes relacionades amb tasques que es poden demanar al model, per exemple, resumir un article o editar una fotografia.

La IA generativa és una tecnologia que evoluciona ràpidament, i contínuament s'hi descobreixen nous casos d'ús. Per exemple, els models generatius estan ajudant les empreses a refinar les seves imatges de productes de comerç electrònic eliminant automàticament els fons de les imatges o bé millorant la qualitat de les imatges de baixa resolució.

7.4. IA tradicional vs. IA generativa

Shad Griffin, enginyer d'intel·ligència artificial a IBM, ens fa una comparació entre la IA tradicional amb la IA generativa.



7.5. Text

Els sistemes d'IA generativa de text són eines de software que utilitzen intel·ligència artificial per generar contingut textual de forma automàtica. Alguns dels principals sistemes d'aquest tipus són:

Models de llenguatge extensos (MLE)

Els MLE (an anglès, LLM, Large Language Models) són els sistemes més avançats i versàtils per a la generació de text. Alguns exemples destacats són:

- **GPT-4 d'OpenAI:** El model més potent i capaç actualment, pot generar text de gran qualitat en múltiples idiomes i formats.
- **ChatGPT d'OpenAI:** Versió conversacional de GPT, molt popular per la seva facilitat d'ús.
- **LLaMA de Meta:** Model de codi obert amb versions de diferents mides.
- **Claude d'Anthropic:** Competidor de ChatGPT amb capacitats similars.

Eines especialitzades

Hi ha també sistemes més enfocats a tasques específiques:

- **Jasper AI:** Orientat a la generació de contingut per a màrqueting i publicitat
- **Copy.ai:** Especialitzat en la creació de textos publicitaris i comercials.
- **Rytr:** Genera diversos tipus de contingut com articles, correus electrònics, etc
- **Frase AI:** Enfocada en la creació de contingut optimitzat per SEO.

Característiques comunes

La majoria d'aquests sistemes comparteixen algunes característiques:

- Utilitzen models d'IA entrenats amb grans quantitats de dades textuais
- Permeten generar text a partir d'instruccions o prompts de l'usuari
- Ofereixen opcions per personalitzar l'estil, to i format del text generat
- Poden produir diversos tipus de contingut com articles, correus, descripcions, etc.

En resum, els sistemes d'IA generativa de text ofereixen una eina potent per automatitzar i agilitzar la creació de contingut escrit, tot i que encara requereixen supervisió humana per garantir-ne la qualitat i precisió.

7.6. Text a imatge

Els sistemes d'IA generativa de text a imatge són eines que utilitzen intel·ligència artificial per crear imatges a partir de descripcions textuais. Aquests sistemes han experimentat un ràpid desenvolupament en els darrers anys i ofereixen noves possibilitats creatives en diversos camps. Alguns dels principals sistemes d'aquest tipus són:

Models destacats

Imagen de Google

Aquest model de difusió destaca per la seva capacitat de generar imatges fotorealistes amb un alt nivell de comprensió del llenguatge.

DALL-E d'OpenAI

DALL-E 3 és una versió avançada que treballa sobre ChatGPT 4, permetent generar imatges més detallades. Per a cada indicació, genera quatre imatges úniques i permet refinar els resultats amb indicacions de seguiment.

Midjourney

Conegut per les seves imatges fotorealistes d'alta qualitat, Midjourney funciona com un bot dins Discord. Tot i que la seva interfície pot resultar inicialment confusa, ofereix resultats impressionants.

Adobe Firefly

Integrat a les eines d'edició d'Adobe, Firefly destaca per oferir un alt nivell de control granular en la generació d'imatges. La seva funció "Generative Fill" permet reemplaçar àrees específiques d'una imatge mitjançant indicacions de text.

Característiques comunes

La majoria d'aquests sistemes comparteixen algunes característiques:

- Utilitzen models d'IA entrenats amb grans conjunts de dades d'imatges i text
- Permeten generar imatges a partir d'indicacions textuais detallades
- Ofereixen opcions per personalitzar l'estil, composició i elements de les imatges generades
- Poden produir diversos tipus d'imatges, des de fotorealistes fins a il·lustracions artístiques

Aplicacions

Els generadors de text a imatge tenen múltiples aplicacions, com ara:

- Creació d'esbossos i prototips en disseny
- Generació d'il·lustracions per a contingut educatiu
- Producció de contingut visual per a màrqueting i xarxes socials
- Assistència en la conceptualització d'idees visuals per a projectes creatius
- Generació de dades de prova per a desenvolupament de software

Aquests sistemes continuen evolucionant ràpidament, oferint cada vegada més possibilitats creatives i aplicacions pràctiques en diversos camps professionals i artístics.

7.7. Text a codi

Els sistemes d'IA generativa de text a codi són una aplicació específica de la intel·ligència artificial generativa que permet generar codi de programació a partir de descripcions en llenguatge natural. Aquests sistemes ofereixen diverses funcionalitats i avantatges per als desenvolupadors:

Funcionament

Els models d'IA generativa de text a codi es basen en grans models de llenguatge entrenats amb grans quantitats de codi font disponible públicament. Aquests models aprenen els patrons i l'estructura dels diferents llenguatges de programació, permetent-los generar nou codi a partir de descripcions en llenguatge humà.

Capacitats principals

- **Generació de codi:** Poden crear fragments de codi, funcions completes o fins i tot programes sencers a partir d'una descripció en llenguatge natural del que es vol aconseguir.
- **Completació de codi:** Ofereixen suggeriments per completar línies o blocs de codi mentre s'està programant.
- **Depuració i explicació:** Alguns models poden ajudar a identificar errors en el codi i explicar el seu funcionament.
- **Suport multilingüe:** Els sistemes més avançats poden treballar amb múltiples llenguatges de programació com Python, JavaScript, Java, C++, etc.

Exemples de sistemes

Alguns dels principals sistemes d'IA generativa de text a codi són:

- **Gemini Code Assist:** Desenvolupat per Google, ofereix generació i completació de codi, així com assistència via xat.
- **Vertex AI Codey:** Conjunt d'APIs de Google Cloud per generació, completació i consultes sobre codi.
- **GitHub Copilot:** Basat en models d'OpenAI, ofereix suggeriments de codi dins de l'IDE.

Avantatges

- Augmenta la productivitat dels desenvolupadors
- Facilita l'aprenentatge de nous llenguatges
- Permet prototipatge ràpid d'idees
- Ajuda a resoldre problemes complexos

Tot i els seus avantatges, és important tenir en compte que aquests sistemes són eines d'assistència i no substitueixen el coneixement i criteri del programador. El codi generat ha de ser revisat i adaptat segons les necessitats específiques de cada projecte.

7.8. Text a vídeo

Els sistemes d'IA generativa de vídeo són una tecnologia emergent que permet crear contingut audiovisual de forma automàtica a partir de text o altres entrades. Aquests sistemes ofereixen diverses funcionalitats i avantatges:

Funcionament

Els models d'IA generativa de vídeo es basen en xarxes neuronals entrenades amb grans quantitats de vídeos i imatges. Aquests models aprenen a generar nou contingut visual a partir de descripcions textuais o altres indicacions.

Capacitats principals

- **Generació de vídeos a partir de text:** Poden crear vídeos complets basats en una descripció textual del contingut desitjat.
- **Conversió de documents a vídeo:** Alguns sistemes poden generar vídeos a partir de documents com PDFs o presentacions.
- **Creació d'avatars virtuals:** Utilitzen avatars d'IA realistes que poden parlar i moure's com actors reals.
- **Edició i personalització:** Permeten editar i personalitzar els vídeos generats.
- **Generació de música i àudio:** Alguns sistemes poden afegir música de fons i efectes sonors als vídeos.

Exemples de sistemes

Alguns dels principals sistemes d'IA generativa de vídeo són:

- **Synthesia:** Permet crear vídeos amb avatars virtuals a partir de text .
- **Runway:** Se centra en la creació de contingut creatiu i efectes visuals.
- **DeepBrain AI:** Ofereix generació de vídeos amb avatars hiperrealistes.
- **CapCut:** Generador de vídeos amb IA gratuït que crea vídeos a partir de text o imatges.

Avantatges

- Estalvi de temps i recursos en la producció de vídeos
- Facilita la creació de contingut en múltiples idiomes
- Permet generar vídeos a gran escala de forma ràpida
- No requereix equips de filmació ni actors reals

Tot i els seus avantatges, és important tenir en compte que aquests sistemes són eines d'assistència i no substitueixen completament la creativitat humana. El contingut generat ha de ser revisat i adaptat segons les necessitats específiques de cada projecte.

8. Bibliografia

Per elaborar aquest capítol s'han emprat els recursos següents.

Articles

Ramon López de Mántaras, [Cap a la intel·ligència artificial](#)

Llibres

Artificial Intelligence, a Modern Approach, de Peter Norvig i Stuart Russell. Especialment els capítols d'aprenentatge a partir d'exemples

Python Machine Learning, de Sebastian Raschka

Python Deep Learning, de Jordi Torres

Pàgines web

<https://developers.google.com/machine-learning/intro-to-ml>