

Inteligența artificială

Inteligența artificială desemnează capacitatea mașinărilor de a simula inteligența umană. Această tehnologie reprezintă orice activitate care ar putea fi realizată de o persoană, de exemplu dispozitive inteligente (Amazon Alexa) care interpretează comenzi vocale și oferă înapoi un răspuns sau cum este cazul unui computer care participă la un joc strategic precum Go, șah etc.

În momentul actual cantitățile uriașe de stocare de date și centrele mari de date fac lucrurile posibile, ceea ce în trecut erau considerate mult timp ca fiind idei îndepărtate. Cele două ramuri ale inteligenței artificiale, învățarea automată (machine learning) și învățarea aprofundată (deep learning) se folosesc de posibilitățile date de datele mari (*big data*) pentru a îmbunătăți procesele pentru a descoperi noi soluții și pentru a dobândi noi perspective.

*Cu ajutorul software-ului, datele adunate de la micile întreprinderi și până la marile corporații internaționale pe care le pot folosi sunt consolidate și evaluate.

Această inteligență cuprinde sisteme simple, care se bazează pe o gândire creată în jurul judecatei "dacă A, atunci B". Aceste motoare cu reguli constituie un set de reguli programate de către om și nu neapărat o formă de inteligență.

Învățarea automată (Machine learning)

Învățarea automată este un subdomeniu al informaticii și o subdiviziune a inteligenței artificiale, al cărui scop este de a dezvolta procedee care permit calculatoarelor posibilitatea de a învăța. Este o formă de inteligență artificială care implică un algoritm mult mai capabil de a interpreta și de a înțelege datele pe care acesta le obține. Mai exact, se urmărește să se creeze programe capabile de generalizare a comportamentului pe baza informațiilor furnizate în formularul de exemple. Învățarea automată poate fi considerat ca o încercare de a automatiza unele părți din procesul științific, folosind metode matematice.

Acest tip de învățare conține o serie largă de aplicații, inclusiv motoarele de cautare, diagnostice medicale, detectare de fraudă în utilizarea cardului de credit, analiză a pieței de valori, clasificare a secvențelor de ADN, recunoaștere a vorbirii și limbajului scris, jocuri și robotică.

În acest moment, metoda cea mai des folosită este recunoașterea imaginilor. Alte aplicații include roboți inteligenți sau asistenți digitali, recunoașterea vocii, recunoașterea facială, traducerea automată și transcrierea, analiza videoclipurilor și a textului și conducerea autonomă a vehiculelor.

Învățarea automată are ca rezultat un model pentru a rezolva o anumită sarcină. Aceste modele pot fi:

-Modele geometrice:

-Modele probabilistice:.....

-Modele logice:.....

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

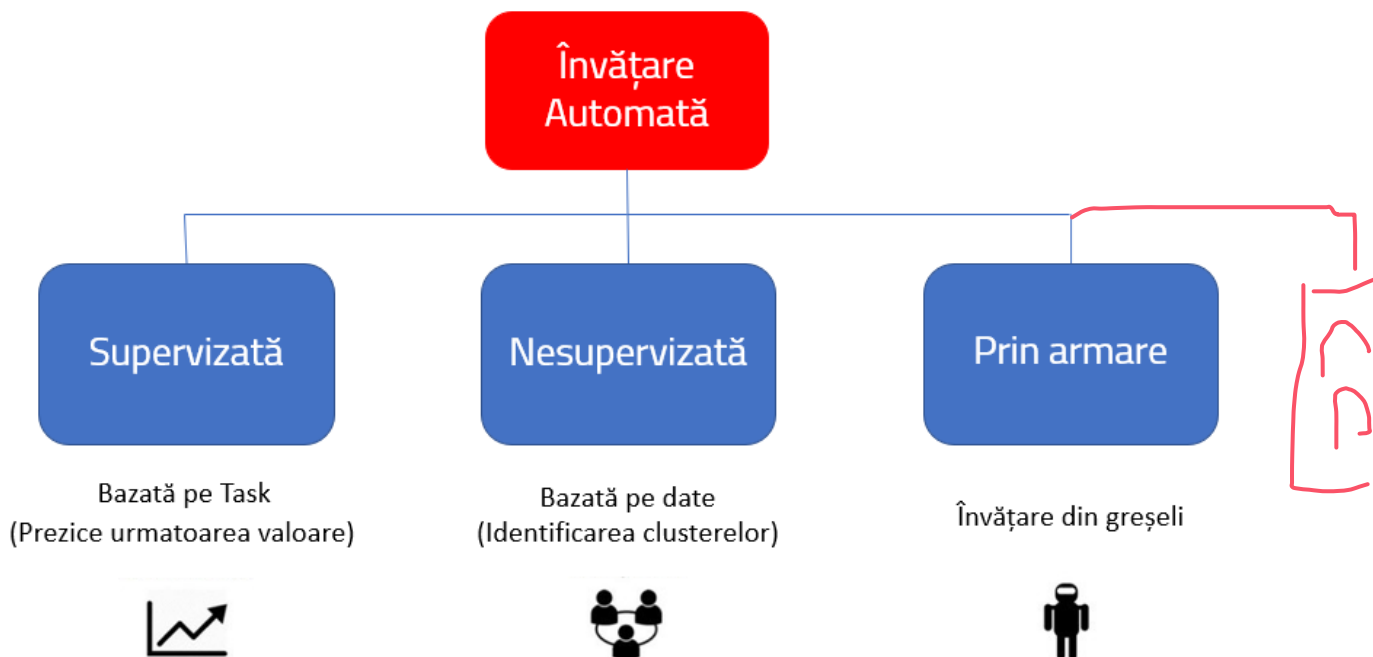
Datele cresc în fiecare zi și este imposibil să se înțeleagă toate datele cu o viteză mai mare și o precizie mai mare. Mai mult de 80% din date sunt nestructurate, adică audio, videoclipuri, fotografii, documente, grafice etc. Găsirea modelelor în datele de pe planeta noastră este imposibilă pentru creierul uman. Datele au devenit foarte masive, timpul necesar calculării crește și aici intră în acțiune Învățarea Automată, pentru a ajuta oamenii cu date semnificative în timp minim.

În învățarea automată, a fost dezvoltată o nouă aptitudine pentru computere. Și acum învățarea automată este prezentă în atât de multe segmente de tehnologie.

Clasificarea cea mai cunoscută a algoritmilor de machine learning face referire la tipul de învățare al sistemului. În prezent se utilizează trei modele principale de algoritmi de învățare automată: învățare automată supravegheată, nesupravegheată și prin armare. Diferența dintre primele două modele constă în modul în care fiecare învață despre date pentru a face previziuni.

și semi-sup

Tipuri de Învățare Automată



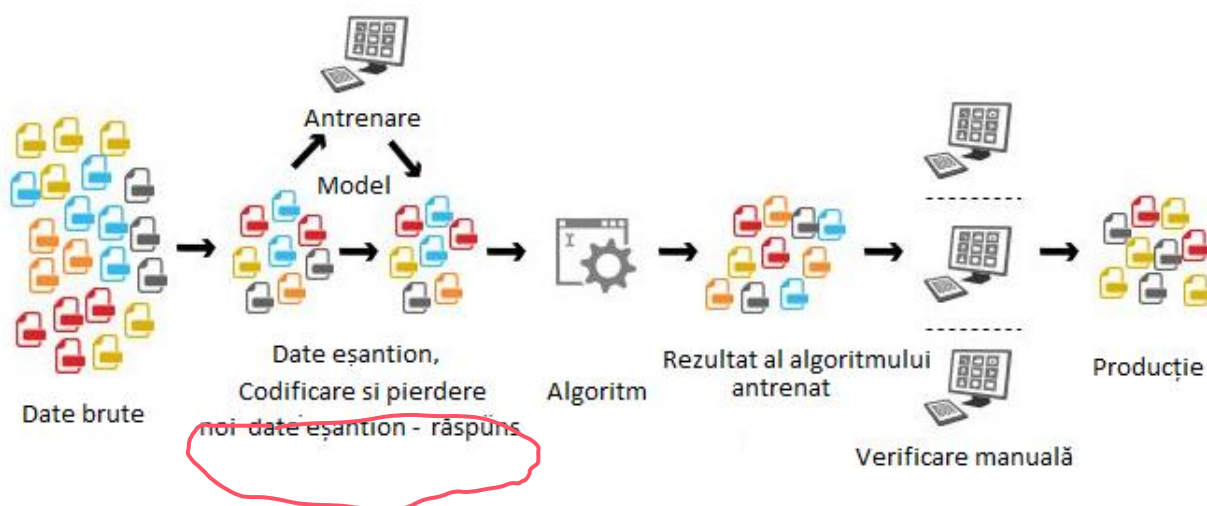
1. Învățarea automată supervizată:

Învățarea automată supervizată: algoritmi de învățare automată sunt cei mai des folosiți. Cu acest tip, un expert în date acționează ca un ghid și învață algoritmul ce concluzii ar trebui să tragă. Așa cum un copil învață să identifice animale prin memorarea lor dintr-o carte ilustrată, pentru învățarea

automată supervizată, algoritmul este antrenat de un set de date care este deja etichetat și are o ieșire predefinită, sunt cunoscute intrarea și ieșirea.

Învățarea supervizată este primul tip de învățare automată, în care datele etichetate sunt utilizate pentru antrenarea algoritmilor. În învățarea supervizată, algoritmi sunt instruiți folosind date marcate, unde intrarea și ieșirea sunt cunoscute. Se introduc datele în algoritmul de învățare ca un set de intrări, care se numesc *Caracteristici*, notate cu X împreună cu ieșirile corespunzătoare, care sunt notate cu Y , iar algoritmul învață comparând producția sa efectivă cu ieșirile corecte pentru a găsi erori. Ulterior, acesta modifică modelul în consecință. Datele brute sunt împărțite în două părți. Prima parte este pentru antrenarea algoritmului, iar cealaltă regiune este utilizată pentru testarea algoritmului antrenat.

Învățarea supervizată utilizează modelele de date pentru a prezice valorile datelor suplimentare pentru etichete. Această metodă se va utiliza în mod obișnuit în aplicații în care datele istorice prezic evenimente viitoare probabile. Ex: - Poate anticipa când tranzacțiile sunt susceptibile de a fi frauduloase sau care client de asigurare este așteptat să depună o cerere.



Învățare automată supervizată

Tipuri de învățare supervizată

Învățarea supervizată în principal este împărțită în două părți care sunt după cum urmează:



Regresia:

Regresia este tipul de învățare supervizată în care au fost utilizate datele etichetate, iar aceste date sunt utilizate pentru a face predicții într-o formă continuă. Ieșirea de intrare este întotdeauna în curs de desfășurare, iar graficul este liniar. Regresia este o formă de tehnică de modelare predictivă care investighează relația dintre o variabilă dependentă [Ieșiri] și o variabilă independentă [Intrări].

Această tehnică utilizată pentru prognozarea vremii, modelarea seriilor de timp, optimizarea proceselor. Ex: unul dintre exemplele tehnicii de regresie este predicția prețului casei, unde prețul casei va fi prezis din intrări, cum ar fi numărul de camere, localitatea, ușurința transportului, vârsta casei, zona casei.

Tipuri de algoritmi de regresie:

Există mulți algoritmi de regresie prezenți în învățarea automată, care sunt folosiți pentru diferite aplicații de regresie. Unii dintre principalii algoritmi de regresie sunt după cum urmează:

-Regresia Liniară Simplă:

În regresia liniară simplă, se prezice scorurile pentru o variabilă din evaluările unei a doua variabile. Variabila care se prognozează este numită variabila criteriu și este notată cu Y. Variabila pe care se bazează predicțiile se numește variabila predictor și este notată cu X.

-Regresia Liniară Multiplă:

Regresia liniară multiplă este unul dintre algoritmi tehnicii de regresie și este cea mai comună formă de analiză a regresiei liniare. Ca o analiză predictivă, regresia liniară multiplă este utilizată pentru a explica relația dintre o variabilă dependentă cu două sau mai multe variabile independente. Variabilele independente pot fi continue sau categorice.

-Regresia polinomială:

Regresia polinomială este o altă formă de regresie în care puterea maximă a variabilei independente este mai mare de 1. În această tehnică de regresie, cea mai bună linie de potrivire nu este o linie dreaptă, ci este sub forma unei curbe.

-Regresia Vector-Suport/Regresia cu Suport Vectorial/Regresia Vectorială de Suport

Regresia vectorială de suport poate fi aplicată nu numai problemelor de regresie, ci și în cazul clasificării. Aceasta conține toate caracteristicile care caracterizează algoritmul de marjă maximă.

Cartarea mașinii de învățare liniară sprijină o funcție neliniară în spațiul caracteristic indus de nucleu de înaltă dimensiune. Capacitatea sistemului a fost controlată de parametri care nu depind de dimensionalitatea spațiului caracteristic.

-Regresia Ridge (Regresia Crestei):

Regresia Ridge este unul dintre algoritmi utilizați în tehnica de regresie. Este o tehnică de analiză a datelor de regresie multiplă care suferă de multicolaritate. Prin adăugarea unui grad de distorsiune la calculele de regresie, reduce erorile standard. Efectul net va fi acela de a da calcule mai fiabile.

-Regresia Lasso:

Regresia Lasso este un tip de regresie liniară care folosește contracția. Reducerea este locul în care valorile datelor se micșorează spre un punct central, cum ar fi media. Procedura Lasso încurajează modele simple, rare (adică modele cu mai puțini parametri). Acest tip particular de regresie este potrivit pentru modelele care prezintă niveluri ridicate de multicolaritate sau când se dorește să se automatizeze anumite părți ale selecției modelului, cum ar fi selecția variabilă / eliminarea parametrilor.

-Regresia Netă Elastică (ElasticNet):

Regresia netă elastică a combinat normele L1 (LASSO) și normele L2 (regresia crestei) într-un model penalizat pentru regresia liniară generalizată și îi conferă proprietăți de raritate (L1) și robustețe (L2).

-Regresia Bayesiană:

Regresia Bayesiană permite unui mecanism rezonabil de natural să supraviețuiască datelor insuficiente sau datelor slab distribuite. Acesta permite să pună coeficienți pe prior și zgomet, astfel încât priorii să poată prelua în absența datelor.

-Regresia Arborelui decizional:

Arborele de decizie construiește o formă ca o structură de arbore din modele de regresie. Aceasta împarte datele în subseturi mai mici și, în același timp, un arbore decizional asociat s-a dezvoltat treptat/în mod incremental. Rezultatul este un arbore cu noduri de decizie și noduri de frunze.

-(Regresie forestieră aleatorie) Random Forest Regression:

Random Forest (Mulțimea aleatorie) este, de asemenea, unul dintre algoritmii utilizați în tehnica de regresie și este foarte flexibil, ușor de utilizat, algoritmul de învățare automată care produce, chiar și fără reglarea hiper-parametrilor. De asemenea, acest algoritm este utilizat pe scară largă datorită simplității sale și a faptului că poate fi folosit atât pentru sarcini de regresie, cât și pentru clasificare. Mulțimea (The forest) pe care o construiește este un ansamblu de arbori de decizie, de cele mai multe ori instruiți cu metoda „împachetării”.

Clasificarea:

Clasificarea este tipul de învățare supravegheată în care pot fi utilizate datele etichetate, iar aceste date sunt utilizate pentru a face predicții într-o formă non-continuă. Leșirea informațiilor nu este întotdeauna continuă, iar graficul este neliniar. În tehnica de clasificare, algoritmul învață din datele introduse și apoi folosește această învățare pentru a clasifica o nouă observație. Acest set de date poate fi doar bi-clasă sau poate fi și multi-clasă. Ex: - Unul dintre exemplele de probleme de clasificare este de a verifica dacă e-mailul este spam sau nu spam prin antrenarea algoritmului pentru diferite cuvinte spam sau e-mailuri.

Tipuri de Algoritmi de Clasificare:

Există mulți algoritmi de clasificare care sunt prezenți în învățarea automată, care sunt folosiți pentru diferite aplicații de clasificare. Unii dintre principalii algoritmi de clasificare sunt după cum urmează:

— Regresia/Clasificarea Logistică:

Regresia logistică se încadrează în categoria învățării supervizate; măsoară relația dintre variabila dependentă care este categorică cu una sau mai multe variabile independente prin estimarea

probabilităților folosind o funcție logistică / sigmoidă. Regresia logistică se poate utiliza în general în cazul în care variabila dependentă este binară sau dicotomă. Înseamnă că variabila dependentă poate lua doar două valori posibile, cum ar fi „Da sau Nu”, „Viu sau Mort”.

— K-Nearest Neighbours (Vecinii K cei mai apropiați):

Algoritmul KNN este unul dintre cei mai simpli algoritmi din clasificare și este unul dintre cei mai utilizați algoritmi de învățare. Un vot majoritar al unui obiect este clasificat de vecinii săi, scopul fiind atribuit clasei cele mai comune dintre cei mai apropiați k vecini ai săi. Se poate utiliza și pentru regresie - ieșirea este valoarea obiectului (prezice valori continue). Această valoare este media (sau mediana) beneficiilor celor mai apropiați k vecini ai săi.

Mașinile Vector Suport/Mașini Vectoriale de Suport (Support Vector Machines):

O mașină vectorială de suport este un tip de clasificator, în care un clasificator discriminativ definit formal de un hiperplan de separare. Algoritmul generează un hiperplan optim care clasifică noi exemple. În spațiul bidimensional, acest hiperplan este o linie care împarte un plan în două părți în care fiecare clasă se află pe ambele părți.

Mașinile Vector Suport Kernel/Mașini Vectoriale de Suport Kernel (Kernel Support Vector Machines):

Algoritmul Kernel-SVM este unul dintre algoritmii utilizați în tehnica de clasificare și este setul de funcții matematice care este definit ca (kernel)nucleu. Scopul nucleului este de a lua date ca intrare și de a le transforma în forma necesară. Diferiți algoritmi SVM folosesc diferite tipuri de funcții (kernel) de nucleu. Aceste funcții pot fi de diferite tipuri. De exemplu, funcții liniare și neliniare, funcții polinomiale, funcții de bază radială și funcții sigmoide.

Naive Bayes:

Naive Bayes este un tip de tehnică de clasificare, care se bazează pe teorema lui Bayes cu o presupunere de independență între predictorii. În termeni simpli, un clasificator Naive Bayes presupune că prezența unei anumite caracteristici într-o clasă nu are legătură cu prezența oricărei alte funcții. Modelul Naive Bayes este accesibil de construit și este deosebit de util pentru seturi de date extinse.

Clasificarea Arborelui Decizional:

Arborele decizional face modele de clasificare sub forma unei structuri arborescente. Un arbore de decizie asociat dezvoltat în mod incremental și, în același timp, împarte un set mare de date în subseturi mai mici. Rezultatul final este un arbore cu noduri de decizie și noduri de frunze. Un nod de

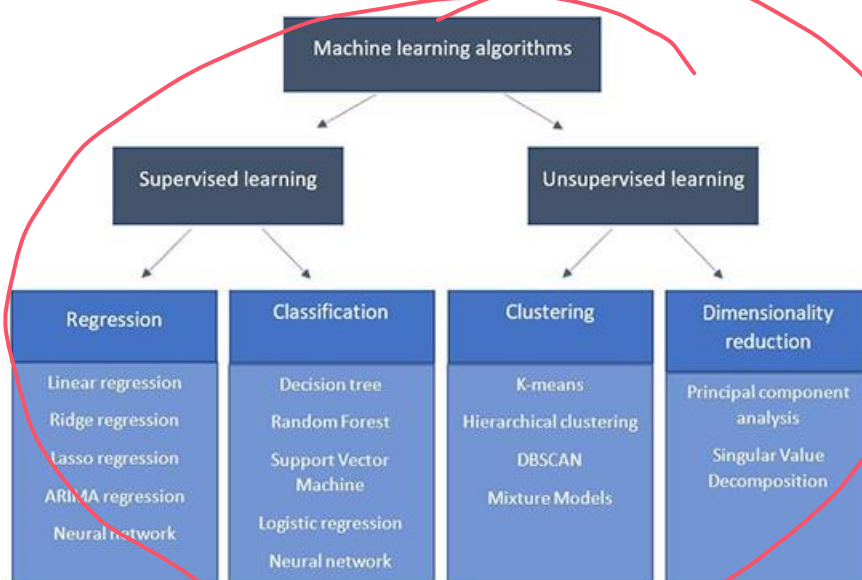
decizie (de exemplu, Root(radacină)) are două sau mai multe ramuri. Nodul frunzei reprezintă o clasificare sau o decizie. Primul nod de decizie dintr-un copac care corespunde celui mai bun predictor si este numit nod rădăcină. Arborii de decizie pot gestiona atât date categorice, cât și date numerice.

Random Forest Classification (Clasificarea aleatorie a multimilor):

Random Forest (Mulțimea aleatorie) este un algoritm de învățare supravegheat. Creează o pădure și o face cumva casuală. Lemnul pe care îl construiește este un ansamblu de arbori de decizie, de cele mai multe ori algoritmul arborelui decizional instruit cu metoda „punțire”, care este o combinație de modele de învățare crește rezultatul general.

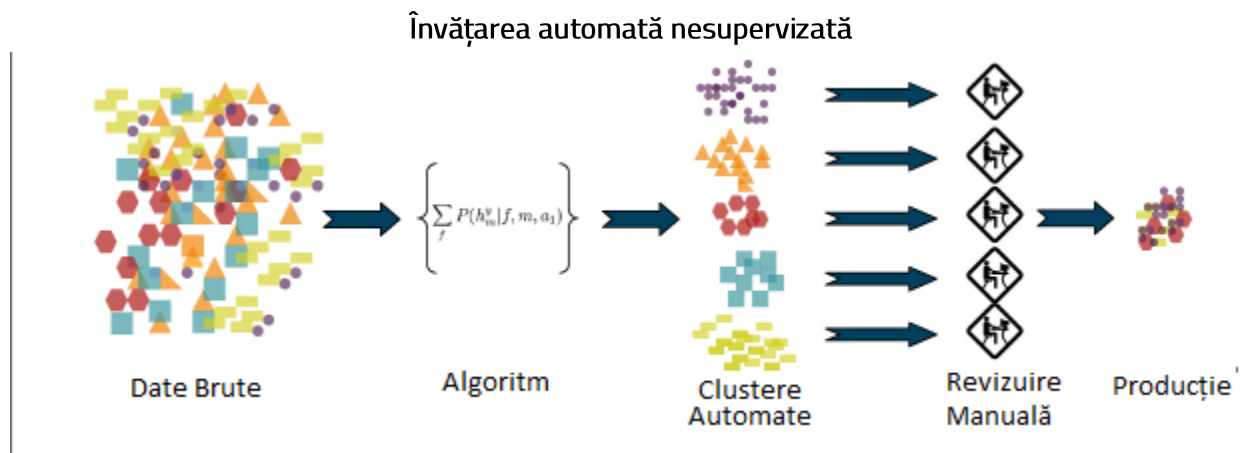
2.Învățarea nesupervizată:

Învățarea automată nesupervizată: acest tip de învățare folosește o abordare mai independentă, în care un computer învață să recunoască procese și modele complexe, fără ajutorul unui om care să ofere îndeaproape orientări în mod frecvent. Învățarea automată nesupervizată implică pregătirea bazată pe date care nu au etichete sau o ieșire specifică, definită. Pentru a continua legătura cu sistemul din copilărie, învățarea automată nesupervizată este asemănătoare cu un copil care învață să recunoască animalele prin observarea culorilor și modelelor, în loc să memoreze numele cu ajutorul unui profesor. Copilul va încerca să găsească asemănări între imagini și le va separa în grupuri, atribuind fiecărui grup propria etichetă nouă. Exemple de algoritmi pentru învățarea automată nesupervizată cuprinde gruparea în k clustere, analiza principală și independentă a



Clasificarea imaginilor folosește algoritmi de învățare automată pentru a atribui oricărei imagini de intrare o etichetă dintr-un set fix de categorii. Aceștia permit organizațiilor să modeleze planuri de construcție 3D bazate pe modele 2D, să ușureze marcarea fotografiilor în mediile de socializare, să ofere diagnostice medicale și multe altele.

Învățarea nesupervizată este al doilea tip de învățare automată, în care datele neetichetate sunt folosite pentru a antrena algoritmul, ceea ce înseamnă că este utilizat împotriva datelor care nu au etichete istorice. Ceea ce se arată trebuie să se afle prin algoritm. Scopul este de a explora datele și de a găsi o structură în interior. În învățarea nesupervizată, datele nu sunt etichetate, iar introducerea de informații brute direct în algoritm fără prelucrarea prealabilă a datelor și fără a cunoaște ieșirea datelor și datele nu pot fi împărțite într-o antrenare sau testare de date. Algoritmul calculează datele și, în funcție de segmentele de date, creează grupuri de date cu etichete noi.



Această tehnică de învățare funcționează bine pe datele tranzacționale. De exemplu, poate identifica segmente de clienți cu atribute similare care pot fi apoi tratați în mod similar în campaniile de marketing. Sau poate găsi calitățile primare care separă segmentele de clienți unul de celălalt. Acești algoritmi sunt, de asemenea, utilizați pentru a segmenta subiecte de text, pentru a recomanda elemente și pentru a identifica valori anormale ale datelor.

Tipuri de Învățare Nesupravegheată:

Învățarea nesupravegheată este împărțită în principal în două părți, care sunt după cum urmează:

Grupare (Clustering):

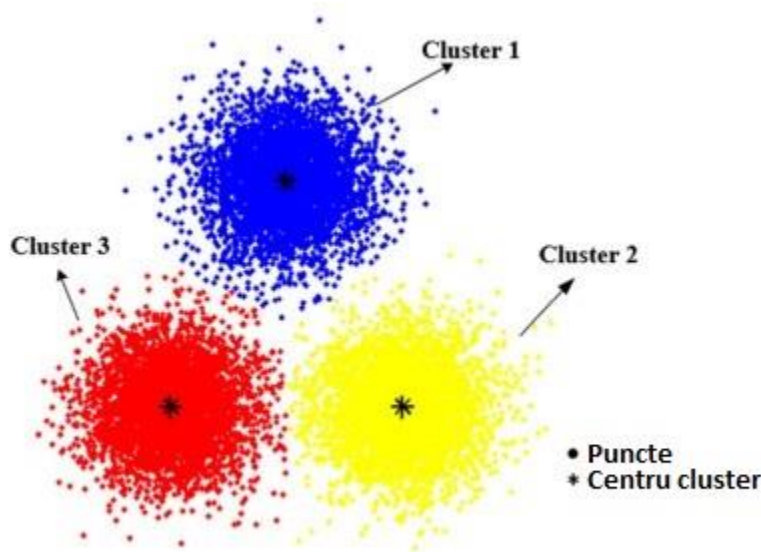
Clusterizarea este tipul de învățare nesupervizată în care s-au folosit date neetichetate și este procesul de grupare a entităților similare împreună, iar apoi datele grupate sunt utilizate pentru a face clustere (grupuri/grupări). Scopul acestei tehnici de învățare automată nesupervizată este de a găsi similitudini în punctul de date și de a grupa puncte de date similare împreună și de a-și da seama că noile date ar trebui să aparțină unui cluster.

Tipuri de algoritmi de clusterizare/grupare

Există mulți algoritmi de clusterizare care sunt prezenți în învățarea automată, care este utilizat pentru diferite aplicații de clusterizare. Unii dintre principalii algoritmi de clusterizare sunt după cum urmează:

Algoritmul de grupare a datelor K-Means/K-Means Clustering:

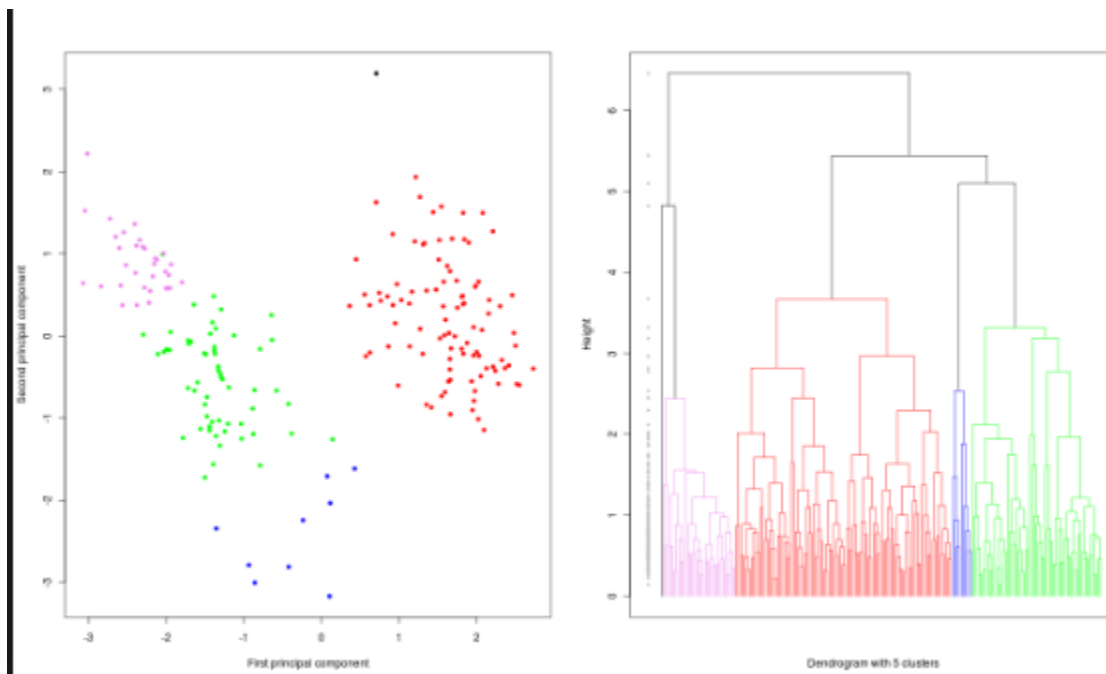
K-Means clustering este unul dintre algoritmi tehnicii Clustering, în care datele similare sunt grupate într-un cluster. K-means este un algoritm de clusterizare iterativ care își propune să găsească maxime locale în fiecare iterație. Începe cu K ca intrare, adică câte grupuri se dorește să se vadă. Se introduceți k centroizi în locații aleatorii din spațiul (dumneavoastră) ales. Acum, cu ajutorul metodei distanței euclidiene, se calculează distanța dintre punctele de date și centroizi și se atribuie punctele de date clusterului care este aproape de acesta. Se recalculază centrele clusterului ca mijloc de puncte de date atașate acestuia. Se repetă până când nu apar alte modificări.



K-Means Clustering afișează 3 clustere

Clusterizarea ierarhică:

Clusterizarea ierarhică este unul dintre algoritmi tehnicii Clustering, în care datele similare s-au grupat într-un cluster. Este un algoritm care construiește ierarhia clusterelor. Acest algoritm începe cu toate punctele de date atribuite unui grup propriu. Apoi, cele mai apropiate două grupuri sunt îmbinate în același cluster. În cele din urmă, acest algoritm se termină atunci când rămâne doar un singur cluster. Se începe prin atribuirea fiecărui punct de date grupului său. Apoi trebuie găsită cea mai apropiată pereche a grupului folosind distanța euclidiană și trebuie combinate într-un singur cluster. Apoi se calculează distanța dintre cele mai apropiate cluster și trebuie combinat până când toate elementele sunt grupate într-un singur cluster.



Reducerea dimensiunii:

Reducerea dimensiunii este tipul de învățare nesupervizată, în care dimensiunile datelor sunt reduse pentru a elimina datele nedorite din intrare. Această tehnică este utilizată pentru a elimina caracteristicile nedorite ale datelor. Se referă la procesul de conversie a unui set de date cu dimensiuni mari în date cu date de același tip și dimensiuni mici. Aceste tehnici sunt utilizate la rezolvarea problemelor de învățare automată pentru a obține caracteristici mai bune.

Tipuri de algoritmi de reducere a dimensiunii:

Există mulți algoritmi de reducere a dimensiunii care sunt prezenți în învățarea automată, sunt utilizați pentru diferite aplicații de reducere a dimensionalității. Unii dintre principalii algoritmi de reducere a dimensionalității sunt după cum urmează:

Analiza Componentelor Principale:

Analiza componentelor principale este unul dintre algoritmii reducerii dimensiunii, în această tehnică, s-a transformat într-un nou set de variabile din vechile variabile, care sunt combinația liniară a variabilelor reale. Un nou set specific de variabile sunt cunoscute ca componente principale. Ca rezultat al transformării, prima componentă primară are varianța cea mai semnificativă posibilă și fiecare element care urmează are cea mai mare diferență de potențial sub constrângerea că este ortogonală cu principiile de mai sus. Păstrarea numai a primelor $m < n$ componente reduce dimensiunea datelor, păstrând în același timp majoritatea informațiilor,

Analiza Discriminantă Liniară:

Analiza discriminantă liniară este unul dintre algoritmii de reducere a dimensiunii în care creează, de asemenea, combinații liniare ale caracteristicilor originale. Cu toate acestea, spre deosebire de ACP (Analiza Componentelor Principale), ADL (Analiza Discriminantă Liniară) nu maximizează varianța explicată. În schimb, optimizează separabilitatea între clase. ADL poate îmbunătăți performanța predictivă a caracteristicilor extrase. Mai mult, ADL oferă variante pentru a face față obstacolelor specifice.

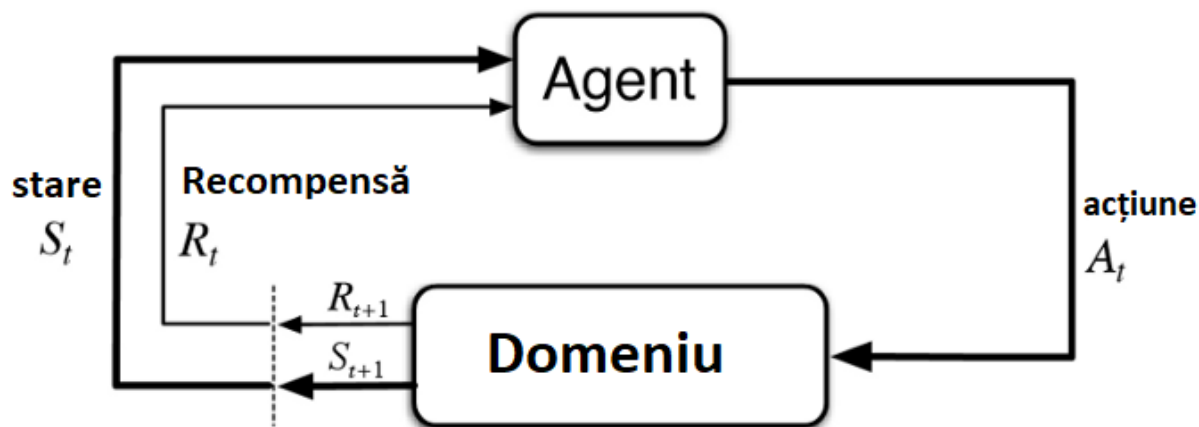
Analiza Componentelor Principale Kernel:

Analiza componentelor principale Kernel este unul dintre algoritmii de reducere a dimensiunii, iar variabilele care sunt transformate în variabile ale noului set, care sunt combinația neliniară a variabilelor originale înseamnă versiunea neliniară a ACP, numită Analiza Componentelor Principale Kernel (ACPK). Este capabil să capteze o parte din statisticile de înaltă ordine, oferind astfel mai multe informații din setul de date original.

3.Învățarea prin armare:

Învățarea prin armare este al treilea tip de învățare automată în care nu se furnizează date brute ca intrare, în schimb algoritmul de învățare prin armare trebuie să descopere situația pe cont propriu. Învățarea prin armare este utilizată frecvent pentru robotică, jocuri și navigație. Odată cu învățarea prin armare, algoritmul descoperă prin încercări și erori care acțiuni produc cele mai semnificative recompense. Acest tip de antrenare are trei componente principale care sunt:

- agentul care poate fi descris ca cursant/învățăcel sau factor de decizie,
- mediul care a descris tot ceea ce interacționează cu agentul,
- acțiunile care sunt reprezentate de ceea ce poate face agentul.



Învățare prin armare

Obiectivul este ca agentul să întreprindă acțiuni care maximizează recompensa estimată pe o anumită măsură de timp. Agentul va atinge obiectivul mult mai repede urmând o metodă bună. Deci, scopul învățării prin armare este de a învăța cel mai bun plan.

Tipuri de învățare prin armare:

Există mulți algoritmi de învățare prin armare care sunt prezenți în învățarea automată, care s-au aplicat pentru diferite aplicații de învățare prin armare. Unii dintre principalii algoritmi sunt după cum urmează:

Q-Learning (Învățarea-Q):

Învățarea-Q este unul dintre algoritmii învățării prin armare, în care un agent încearcă să învețe strategia optimă din experiența sa trecută de comunicare cu domeniul. O înregistrare a unui agent este o succesiune de stări-acțiune-recompense. Învățarea-Q învață o metodă optimă, indiferent

de procedura pe care o urmează agentul, atâta timp cât nu există nicio restricție cu privire la numeroasele încercări ale unei acțiuni în orice stare. Deoarece învață o metodă optimă, indiferent de strategia pe care o desfășoară, se numește o metodă în afara politicii.

SARSA [State Action Reward State Action]: *S-T-A-P-E A-C-T R-E-C-S*

SARSA este unul dintre algoritmi învătăării prin armare, în care se determină să fie actualizat la valorile acțiunii. Este o diferență minoră între implementările SARSA și Învațarea-Q(Q-learning), dar provoacă un efect puternic. Metoda SARSA ia un alt parametru, acțiunea2, care este acțiunea efectuată de agent din a doua stare. Permite agentului să găsească în mod explicit valoarea recompensei viitoare. Apoi.....

Rețeaua-Q profundă (Deep Q-Network):

Rețeaua-Q profundă este unul dintre algoritmi învătăării prin armare, deși învațarea Q este un algoritm foarte robust, principalul său defect este lipsa generalității. Dacă vedeți Q-learning ca reînnoirea numerelor într-o matrice bidimensională (Action Space * State Space), acesta, de fapt, urmează programarea dinamică. Aceasta indică faptul că pentru stările pe care agentul Q-learning (învațării-Q) nu le-a văzut până acum, nu are nici o idee despre ce acțiune să ia. Cu alte cuvinte, un agent Q-learning nu poate estima valoarea pentru stările nevăzute. Pentru a rezolva această problemă, DQN scapă de matricea bidimensională prin introducerea rețelei neuronale.

Procese decizionale Markov:

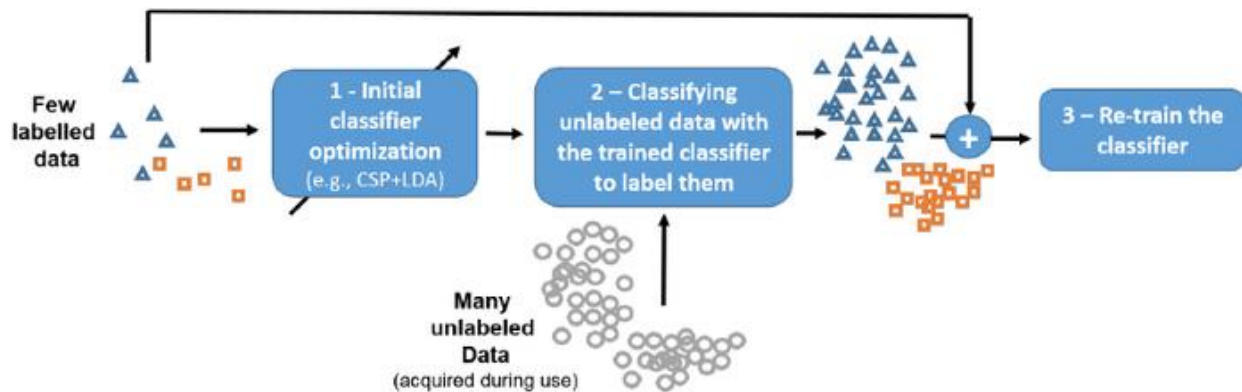
Procesul de decizie Markov este unul dintre algoritmi învătăării prin armare, în care conține * Un set de stări lume posibile S. * Un set de modele. * Un set de acțiuni posibile A. * O funcție de recompensă cu valoare reală R (s, a). * O metodă soluția procesului decizional Markov. Pentru a atinge un obiectiv, se folosește procesul decizional Markov, este un cadru direct al problemei învătăării din interacțiune. Agentul selecta acțiuni și mediul care răspundea la aceste acțiuni, iar agentul și mediul interacționează continuu și prezintă noi situații agentului.

DDPG [Gradient profund de metodă deterministă]:

Gradientul profund de politică deterministă este unul dintre algoritmi învătăării prin armare, în care se bazează pe designul actor-critic cu două componente omonime, actor și critic. Un actor este utilizat pentru a regla parametrul θ pentru funcția de politică, adică pentru a decide cea mai bună acțiune pentru o anumită stare. Ideile de rețea țintă separată și reluarea experienței sunt, de asemenea, împrumutate de la DQN. Explorarea rareori pentru operațiuni este o altă problemă pentru DDPG. O soluție pentru aceasta este adăugarea de zgomot la spațiul parametrilor sau spațiul de acțiune.

4.Învățare semi-supervizată:

Învățarea semi-supervizată este al patrulea tip de învățare automată, în care se utilizează ambele tipuri de date brute. Învățarea semi-supravegheată este un hibrid de învățare automată supervizată și nesupervizată. Învățarea semi-supervizată este utilizată în aceleași scopuri ca și învățarea supervizată, unde folosește atât date etichetate, cât și neetichetate pentru antrenare, de obicei o cantitate mică de date etichetate cu o cantitate semnificativă de date neetichetate. Acest tip de învățare poate fi utilizat cu metode precum clasificarea, regresia și predicția.



Această tehnică este utilă din câteva motive. În primul rând, procesul de etichetare a unor cantități masive de date pentru învățarea supervizată este deseori prohibitiv și costisitor. Mai mult, o etichetare prea mare poate impune părtiniri umane asupra modelului. Aceasta înseamnă că includerea multor date neetichetate în timpul procesului de antrenare tinde să îmbunătățească acuratețea modelului final, reducând în același timp timpul și costurile cheltuite pentru construirea acestuia.

Aplicații ale învățării automate:

Există multe utilizări ale învățării automate în diferite domenii, unele dintre domenii sunt medicale, de apărare, tehnologie, finanțe, securitate, etc. Unele dintre domeniile în care se utilizează acești algoritmi ML sunt următoarele:

