

Machine Learning

• Partea II. Fluxul de date în Machine Learning

Ce ne așteaptă?

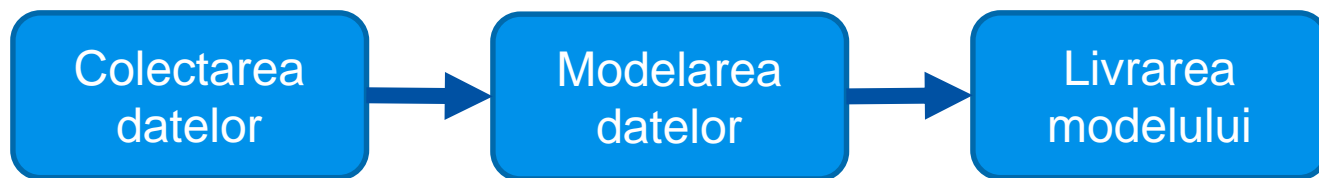
1. Etapele fluxului de date în Machine Learning
2. Definirea problemei Machine Learning
3. Studiarea datelor
4. Analiza caracteristicilor
5. Elaborarea modelului
6. Evaluarea modelului
7. Experimentarea

1. Etapele fluxului de date în Machine Learning

Precondiții ML

- **Algebra liniară** – vectori, matrice, normalizare, ortogonalitate, etc
- **Analiza matematică** – calculul diferențial, calculul integral, etc
- **Teoria probabilității** – variabile aliatore, distribuții uniforme, binomiale, Poisson, normale, etc
- **Statistica** – dispersie, testul z, testul t, testul Anova, Testul Chi-square
- **Limbajul de programarea Python** – variabile, operatori, structuri de date, ramificații și bucle, funcții, etc

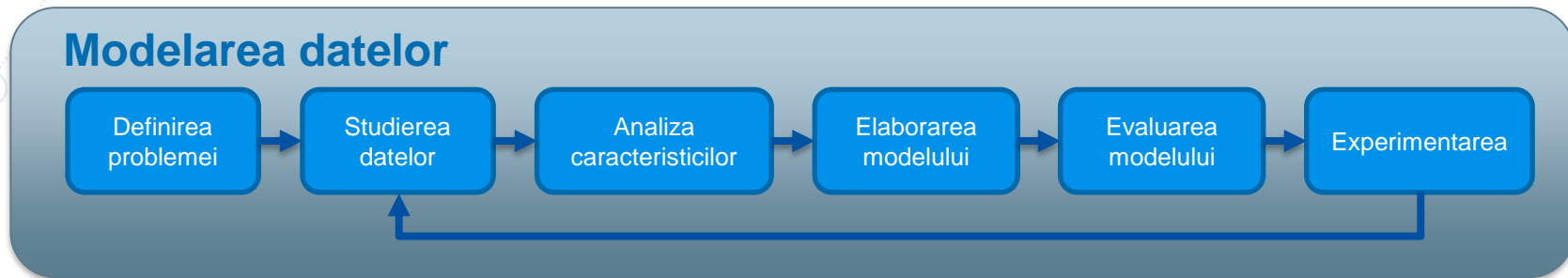
Etapile produsului ML



- **Colectarea datelor** – etapa de colectare a datelor din lumea reală (senzori fizici, sondaje, experimente, etc) și stocarea acestora
- **Modelarea datelor** – etapa de procesare a datelor, elaborare a modelului și evaluarea a performanțelor acestuia
- **Livrarea modelului** – etapa de elaborarea a produsului finit ce include modelul elaborat (aplicații mobile, servicii web, etc)

Fluxul de date în ML

- **Fluxul de date ML** – structurarea sarcinilor etapei de modelarea a datelor cu selectarea instrumentelor Data Science și ML



- Fluxul de date cuprinde 6 pași care răspund la întrebările esențiale modelării datelor:
 - Definirea problemei – Ce sarcină încerc să soluționez?
 - Studierea datelor – Ca date am la dispoziție?
 - Analiza caracteristicilor – Ce cunosc despre date?
 - Elaborarea modelului – Ce algoritm este potrivit sarcinii mele?
 - Evaluarea modelului – Sunt mulțumit de rezultatele modelului?
 - Experimentarea – Pot să îmbunătățesc modelul elaborat?

2. Definirea problemei Machine Learning

Fără ML

- În etapa de definire a problemei se va răspunde la întrebarea “Ce sarcină încerc să soluționez?”
- Nu se recomandă utilizarea ML pentru problemele care:
 - pot fi soluționate prin metodele simple tradiționale
 - necesită soluții de precizie foarte înaltă (100%)

Tipuri de ML

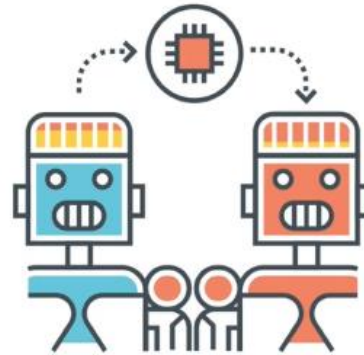
- Definirea problemei presupune selectarea tipului ML conform sarcinii propuse
- Tipurile de ML



ML supravegheat
(Supervised Learning)



ML nesupravegheat
(Unsupervised Learning)



Transfer Learning



Reinforcement Learning

ML supravegheat

- **Utilizează date istorice ce conțin caracteristici și etichete**
- **Pe baza caracteristicilor mai multor date se creează model de obținere a etichetelor prin auto-corecție**
- **Procesul de auto-corecție presupune crearea repetată a modelului și compararea rezultatului acestuia cu etichetele datelor până la obținerea celui mai bun rezultat**
- **Noțiunea de supraveghere provine de la actul de comparare a rezultatelor modelului cu etichetele datelor**
- **Modelul elaborat se va utiliza în viitor pentru predicția etichetelor altor noi date similare**

Tipuri de ML supravegheat



ML de clasificare binară



ML de clasificare multi-clase



ML de regresie

- **ML de clasificarea binară** – predicția includerii datelor în una din 2 clase, exemplu: predicția dacă o persoană este bolnavă sau nu
- **ML de clasificarea multi-clase** – predicția includerii datelor în una din mai multe clase, exemplu: predicția rasei câinelui conform pozei acestuia
- **ML de regresie** – predicția unei valori numerice corespunzătoare datelor, exemplu: predicția prețului unei case de locuit

ML nesupravegheat

- **Utilizează date istorice ce conțin doar caracteristici nu și etichete**
- **Pe baza caracteristicilor mai multor date se creează model de gruparea a datelor sau modele de reorganizare și reducere a caracteristicilor**
- **ML nesupravegheat poate fi de tip:**
 - **ML de clusterizare – gruparea datelor conform caracteristicilor acestora, exemplu: gruparea clienților în funcție de cumpărăturile făcute cu scopul de a le transmite materiale promoționale**
 - **Dimensionality Reduction - descoperirea corelațiilor între caracteristicile datelor și reorganizarea caracteristicilor în funcție de importanța lor**

Transfer Learning

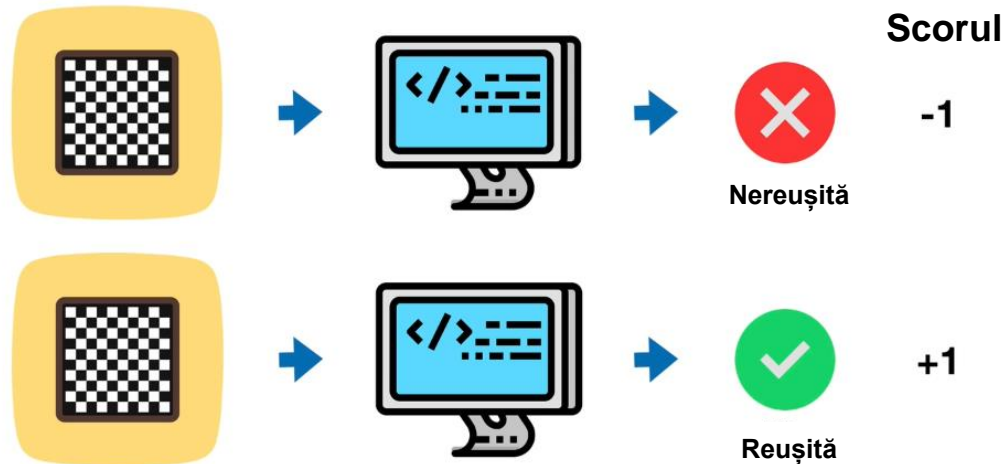
- **Transfer Learning** – utilizarea unui model ML care a fost învățat pe unele date similare și adaptarea acestuia propriilor date
- **Transfer Learning** este larg utilizat în Deep Learning în probleme de recunoaștere vizuală sau procesare a limbajului natural.
- Procedura de elaborare și training a propriei rețele neuronale în DL poate fi substituită prin utilizează unor rețele numite “state of art” învățate pe date similare mai generale și care asigură performanțe mult mai înalte.
- **Exemplu:** Pentru crearea modelului de predicție a rasei câinelui conform pozei acestuia se pot utiliza și adapta rețele neuronale EfficientNet care au fost învățate pe poze a diferitor obiecte din Imagenet

Reinforcement Learning

- Reinforcement Learning este un tip nou al ML care presupune învățarea unui agent în funcție de comportamentul său într-un mediu.
- Reinforcement Learning necesită prezența unui agent, un mediu în care este plasat agentul și un scop ce trebuie atins de către agent.
- Fiecare acțiune a agentului este dependentă de poziția acestuia în mediu și de scopul ce trebuie atins.
- Dacă acțiunea curentă duce la îndepărtarea de scop agentul este învățat prin pedepsire, iar dacă duce la apropierea de scop - prin încurajare.
- La baza procesului de învățare în Reinforcement Learning stau tanturile lui Markov

Exemplu Reinforcement Learning

- Exemplu:** Jocul de șah al unui calculator. Una din piese reprezintă agentul, tabla de șah - mediul și punctajul maxim - scopul. Piesa curentă este plasată pe o poziție a tablei iar mutarea acesteia va reprezenta o acțiune a agentului. O mutare pe tabla de șah poate fi reușită sau nereușită deci va avea drept efect creșterea cu +1 sau descreșterea cu -1 a punctajului și în consecință agentul este încurajat sau pedepsit



Coordonarea problemei

- În funcții de date se coordonează propria problemă cu tipul ML:

- Datele istorice sunt cu etichete categoriale din 2 clase



ML de clasificare binară

- Datele istorice sunt cu etichete categoriale din clase multiple



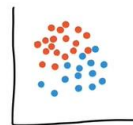
ML de clasificare multi-clase

- Datele istorice sunt cu etichete numerice



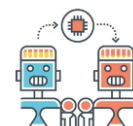
ML de regresie

- Datele istorice sunt fără etichete



ML de clusterizare

- Problema mea poate fi identică cu alte probleme



Transfer Learning

3. Studierea datelor

Clasificarea datelor

- În etapa de studiere a datelor se va răspunde la întrebarea “Ce date am la dispoziție?”
- Primul criteriu de divizarea a datelor, analizat la etapa de definire a problemei, este prezenței etichetelor: date cu etichete și date fără etichete
- Un alt criteriu de divizarea a datelor este în funcție de simplitatea definirii caracteristicilor conform căruia datele pot fi structurate și nestructurate
- În funcție de variația în timp datele pot fi statice și streaming

Date structurate

- Date structurate – caracteristicile datelor pot fi ușor definite și de obicei datele sunt structurate în tabele
- Pe liniile tabelului sunt specificate datele iar pe coloane caracteristicile acestora.
- În cazul datelor structurate cu etichetă, eticheta va fi reprezentată prin intermediul unei coloane
- Datele structurate sunt stocate în fișiere CSV, Excel, baze de date, etc.

ID	Weight	Sex	Blood Pressure	Chest pain	Heart disease?
4328	110kg	M	120/80	4	Yes
5681	64kg	F	130/90	1	No
7911	81kg	M	130/80	0	No



Date nestructurate

- **Date nestructurate – caracteristicile datelor sunt greu de definit sau chiar imposibil de definit de către om**
- **Date nestructurate sunt reprezentate prin intermediul fișierelor image, audio sau text**
- **În cazul datelor nestructurate cu etichetă, eticheta va fi specificată în denumirea fișierului sau într-un alt fișier unde se specifică relația dintre datele fișier și etichete**



Image



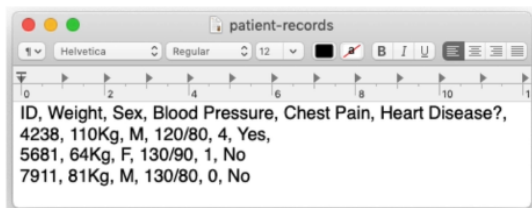
Audio

Pentru început, mersi mult pentru
că ești drăguț.
Acest curs de Machine Learning
este incredibil.

Text

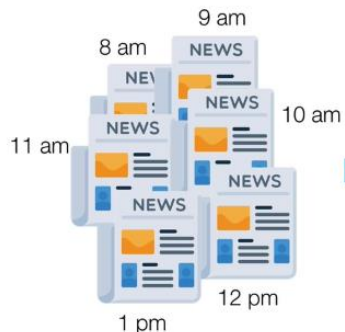
Date statice și date streaming

- Datele statice sunt datele care nu se modifică în timp

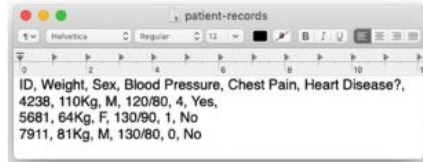


ID	Weight	Sex	Blood Pressure	Chest pain	Heart disease?
4328	110Kg	M	120/80	4	Yes
5681	64Kg	F	130/90	1	No
7911	81Kg	M	130/80	0	No

- Datele streaming sunt datele care se modifică constant în timp



Instrumente de studiere a datelor



patient-records

ID, Weight, Sex, Blood Pressure, Chest Pain, Heart Disease?,
 4238, 110Kg, M, 120/80, 4, Yes,
 5681, 64Kg, F, 130/90, 1, No
 7911, 81Kg, M, 130/80, 0, No

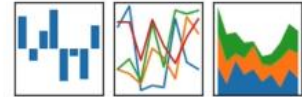


ID	Weight	Sex	Blood Pressure	Chest pain	Heart disease?
4238	110kg	M	120/80	4	Yes
5681	64kg	F	130/90	1	No
7911	81kg	M	130/80	0	No

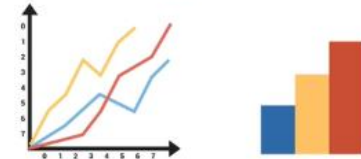
Date statice

pandas

$$y_{it} = \beta^i x_{it} + \mu_i + \epsilon_{it}$$



Analiza datelor



matplotlib



Modelul Machine Learning

4. Analiza caracteristicilor

Ce reprezintă caracteristicile

- În etapa de analiză a caracteristicilor se va răspunde la întrebarea “Ce cunosc despre date?”
- Caracteristicile, numite și variabile de intrare sunt datele despre date prin care datele se aseamnă sau se diferențiază
- În cazul datelor structurate, denumirile caracteristicilor sunt definite de denumirile coloanelor tabelului, exceptând eticheta.
- În cazul datelor nestructurate denumirile caracteristicilor nu pot fi definite de către om.
- În funcție de tipul de reprezentare, caracteristicile pot fi numerice sau categoriale

Caracteristici numerice și categoriale

- Caracteristicile numerice pot lua valori continue și sunt reprezentate prin numere
- Caracteristicile categoriale pot lua valori discrete și sunt reprezentate prin texte sau numere
- Pentru elaborarea modelului, caracteristicile categoriale se vor converti în caracteristici numerice

id	Masa (kg)	Sex	Vârstă (ani)	Înălțime (m)	Tip durere	Fumător?	Bolnav?
2357	81	M	45	1,77	2	Nu	Nu
6439	64	F	37	1,52	0	Nu	Da
4518	77	M	54	1,66	3	Da	Da

Caracteristici numerice

Caracteristici categoriale

Etichetă categorială

Ingineria caracteristicilor

- Ingineria caracteristicilor (Feature engineering) – studiază diferite caracteristici ale datelor, creează unele caracteristici noi sau șterge unele existente
- Caracteristicile noi create pe baza caracteristicilor existente se numesc caracteristic derivate
- Caracteristicile derivate apar și în rezultatul conversiei caracteristicilor categoriale în numerice

id	Masa (kg)	Sex	Vârstă (ani)	Înălțime (m)	Tip durere	Fumător?	Bolnav?	Indice (kg/m)
2357	81	M	45	1,77	2	Nu	Nu	45,76
6439	64	F	37	1,52	0	Nu	Da	42,1
4518	77	M	54	1,66	3	Da	Da	46,39

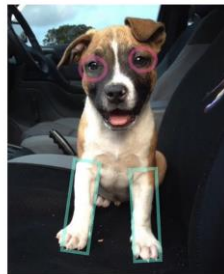
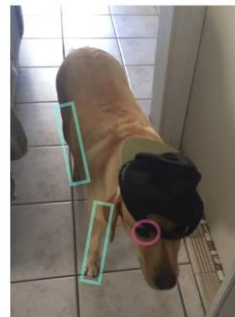
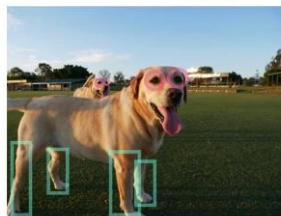
Caracteristici existente

Caracteristici derivate

Eticheta

Caracteristicile datelor nestructurate

- Caracteristicile datelor nestructurate sunt detectate de către algoritm (rețele neuronale) în procesul de training pe mai multe date
- Cu creșterea numărului datelor crește și reprezentativitatea caracteristicilor detectate
- Exemplu de posibile caracteristici ale câinilor diferitor rase



Acoperirea caracteristicilor

- **Acoperirea caracteristicilor presupune analiza lipsei datelor pentru unele caracteristici**
- **Pentru elaborarea modelului ML este necesară excluderea lipsurilor datelor prin ștergerea completă a caracteristicilor sau prin completarea lipsurilor cu anumite valori**
- **În cazul în care pentru o anumită caracteristică lipsesc majoritatea datelor (peste 90%) atunci se poate recurge la excluderea completă a caracteristicii (ștergerea coloanei)**
- **În cazul în care pentru o anumită caracteristică lipsesc unele date, în funcție de importanța și tipul acestora, lipsurile pot fi completate cu anumite valori: valoarea medie, valoarea mediană, o valoare constantă, etc.**

5. Elaborarea modelului

Divizarea datelor

- În etapa de elaborarea a modelului se va răspunde la întrebarea “Ce algoritm este potrivit sarcinii mele?”
- Procedura de divizarea a datelor în seturi este valabilă doar în cazul proiectelor ML supravegheate
- Prima etapă de divizare a datelor constă în specificarea caracteristicilor (setul X) și a etichetei (setul y)
- A doua etapă de divizarea a datelor constă în specificarea atât în setul X cât și setul y a datelor de training, de validare și de test

Setul de date X și setul y

- Setul de date X în cazul datelor structurate include toate coloanele tabelului corespunzătoare caracteristicilor datelor după prelucrarea acestora (caracteristici existente și derivate)
- Setul de date X în cazul datelor nestructurate include toate fișierele de date corespunzătoare
- Setul de date y cuprinde toate etichetele datelor și poate reprezenta coloana corespunzătoare în cazul datelor structurate sau lista denumirilor fișierelor în cazul datelor nestructurate

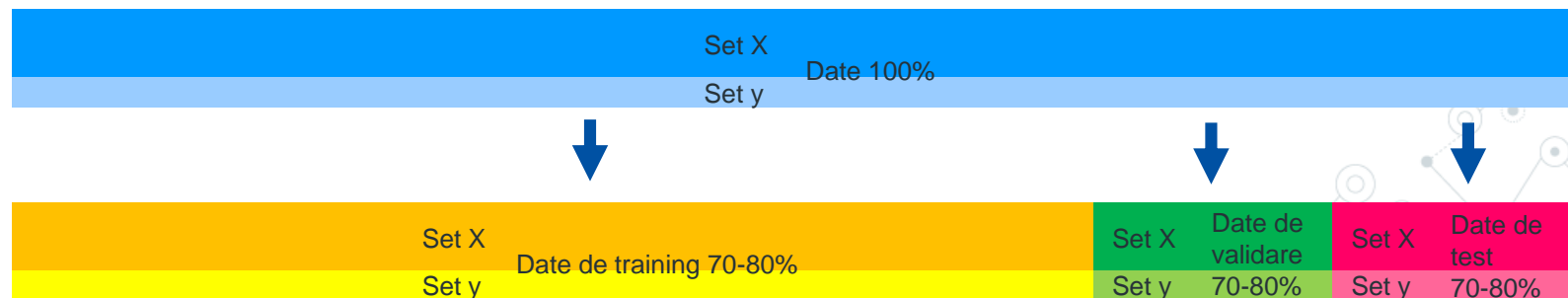
id	Masa (kg)	Sex	Vârstă (ani)	Înălțime (m)	Tip durere	Fumător?	Bolnav?
2357	81	M	45	1,77	2	Nu	Nu
6439	64	F	37	1,52	0	Nu	Da
4518	77	M	54	1,66	3	Da	Da

Setul X

Setul y

Date de training, de validare și de test

- În caz general datele se pot diviza dor în date de training și date de test
- Datele de training (70-80%) asigură învățarea modelului, adică detectarea legității de corelare dintre caracteristici și etichete
- Datele de test (10-15%) permit evaluarea finală a performanței modelului
- Datele de validare (10-15%) permit evaluarea intermediară a performanței modelului cu scopul îmbunătățirii acestuia



Necesitatea datelor de validare

- Datele de validare permit evaluarea versiunilor intermediare ale modelului
- În funcție de rezultatul acestor evaluări se decide reglarea hiperparametrilor modelului întru îmbunătățirea acestuia
- Datele de test permit evaluarea finală a modelului după îmbunătățirile acestuia
- Procedurile de evaluare trebuie să aibă loc pe date neîntâlnite de model



Studierea materialului
(date de training)



Autoevaluare
(date de validare)



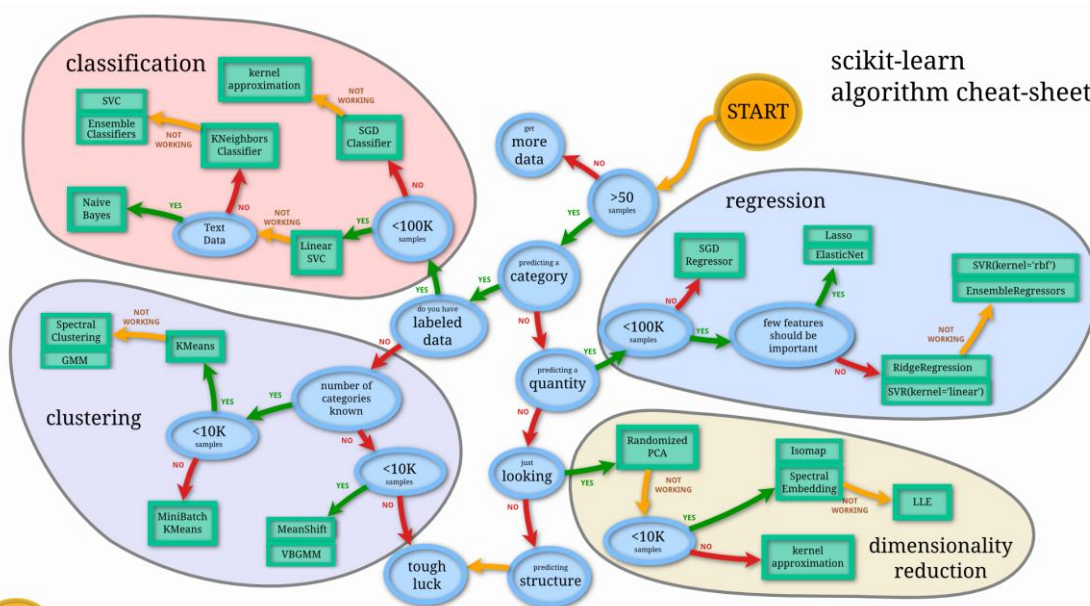
Examen final
(date de test)

Criterii de selectare a algoritmului

- **La baza oricărui proiect ML stă un algoritm statistic care în rezultatul trainingului pe datele de training formează modelul**
- **Diferite algoritme funcționează diferit în funcție de tipul datelor și tipul sarcinii**
- **Pentru selectarea algoritmului se va ține cont de:**
 - **Tipul ML - supravegheat sau nesupravegheat**
 - **Tipul datelor – structurate sau nestructurate**
 - **Tipul sarcinii – de clasificare, de regresie sau de clusterizare**
 - **Indicatorii de performanță propuși**
 - **Timpul de training**
 - **Volumul de date istorice**
 - **etc**

Schema de selectarea a algoritmului

- Producătorii instrumentului Scikit-Learn au elaborat o scheme de selectare a algoritmului în funcție de mai multe criterii



Algoritmi ai ML supravegheat

- **Algoritmi pentru sarcini de regresie**
 - Linear regression
 - Ridge regression
 - Lasso regression
 - Elastic Net
 - Suport Vector Regressor
 - Random Forest Regressor
- **Algoritmi pentru sarcini de clasificare**
 - Logistic regression
 - K Nearest Neighbors
 - Suport Vector Classifier
 - Decision Tree
 - Random Forest Classifier
 - AdaBoost
 - Gradient Boosting
 - Naive Bayes

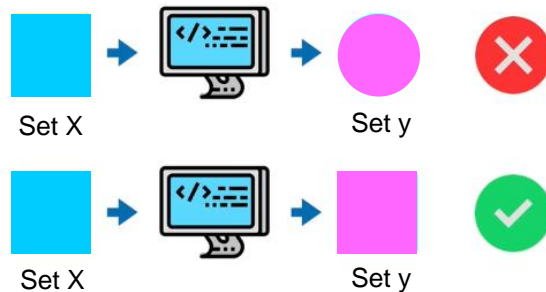
Algoritmi ai ML nesupravegheat și date nestructurate

- **Algoritmi pentru sarcini de clusterizare**
 - **K-Means Clustering**
 - **Hierarchical Clustering**
 - **Support Vector Regressor**
 - **DBSCAN**
- **Algoritmi pentru sarcini dimensionality reduction**
 - **Principal Component analysis**
- **Algoritmi pentru date nestructurate**
 - **Deep Learning (rețele neuronale)**
 - **Transfer Learning**

Procedura de training a modelului

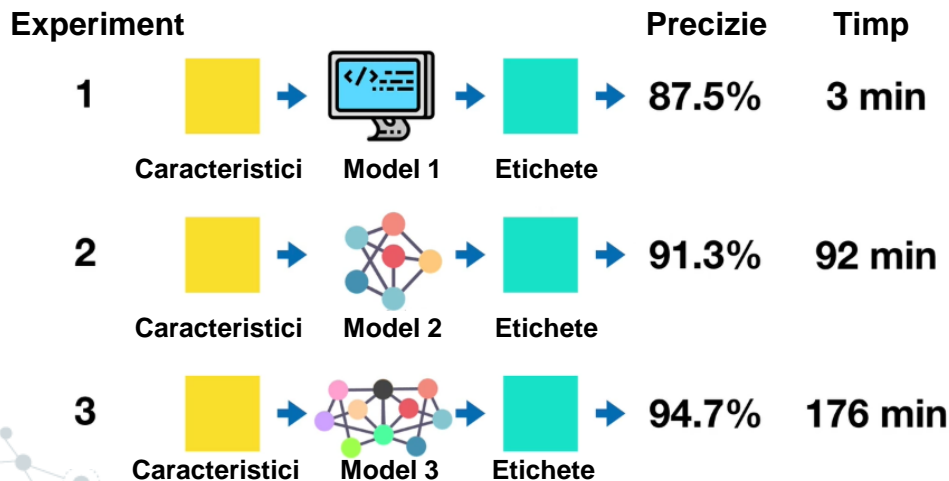
- Procedura de training se realizează pe datele de training și include:
 - Determinarea unei legități în interiorul caracteristicilor de training
 - Compararea rezultatului legității cu etichetele de training
 - Adaptarea automată a legității în funcție de rezultatul comparației
 - Repetarea procedurilor până la obținerea celui mai bun rezultat

id	Masa (kg)	Sex	Vârstă (ani)	Înălțime (m)	Tip durere	Fumător?	Bolnav?
2357	81	M	45	1,77	2	Nu	Nu
6439	64	F	37	1,52	0	Nu	Da
4518	77	M	54	1,66	3	Da	Da



Compromisul timp – performanță

- Procedura de training poate dura mult timp în funcție de algoritm și de dimensiunile datelor de training
- Pentru reducerea timpului într-un proiect ML se experimentează inițial modele mai simple pe mai puține date și treptat se crește în complexitate până la atingerea performanței dorite
- În funcție de compromisul timp-performanță se selectează algoritmul dorit



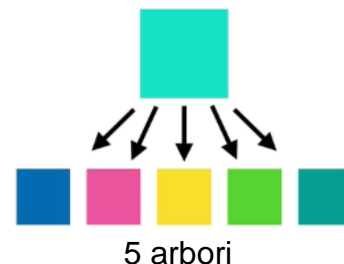
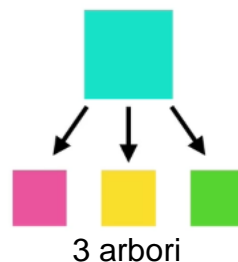
Procedura de validare

- După procedura de training pe datele de training modelul se supune procedurii de validare
- Procedura de validare presupune utilizarea modelului pentru predicția rezultatelor la aplicarea la intrare a caracteristicilor datelor de validare
- Rezultatul predicției pe datele de validare se compară cu etichetele de validare
- În funcție de rezultatul comparării se decide necesitatea ajustării modelului prin setarea hiper-parametrilor algoritmului
- După ajustarea modelului se repetă procedurile de training și de validare până la atingerea performanței dorite

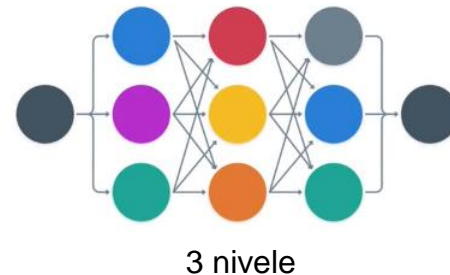
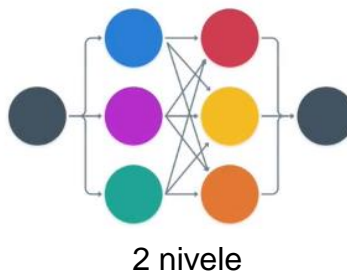
Ajustarea modelului

- Ajustarea modelului presupune setarea de către programator a unor parametri reglabili ai algoritmului numiți hiper-parametri
- Hiper-parametrii diferă de la algoritm la algoritm
- Exemplu de hiper-parametri:

Random Forest: numărul arborilor



Rețele neuronale: numărul nivelelor



6. Evaluarea modelului

Procedura de evaluare

- În etapa de evaluare a modelului se va răspunde la întrebarea “Sunt mulțumit de rezultatele modelului?”
- Odată cu definirea problemei se stabilește și precizia dorită pentru ca modelul să fie valoros
- Precizie modelului depinde de domeniul în care acesta va fi utilizat, de exemplu în medicină precizia va fi de 99%
- Procedura de evaluare se realizează pe datele de test, adică pe date ce nu au fost implicate în procedura de training sau de validare
- Evaluarea finală presupune utilizarea modelului pentru predicția rezultatelor pe caracteristicile de test și comparația acestor rezultate cu etichetele de test

Mărimi de evaluarea

- Evaluarea modelului presupune comparația rezultatelor predicție pe datele de test cu etichetele de test prin intermediul unor mărimi de evaluare
- Tipul mărimilor de evaluare depinde de tipul sarcinii ML

Clasificare	Regresie
Acuratețea (accuracy)	Eroarea medie absolută (Mean Absolute Error - MAE)
Precizie (precision)	Eroarea medie pătratică (Mean Squared Error - MSE)
Sensibilitatea (recall)	Rădăcina erorii medii pătratice (Root Mean Squared Error - RMSE)

Evaluarea de training și evaluarea de test

- Evaluarea de training este realizată în mod automat de către model în procesul de training și reflectă comparația dintre rezultatele predicție pe datele de training și etichetele de training
- Evaluarea de test este realizată de către programator și reflectă comparația dintre rezultatele predicție pe datele de teste și etichetele de test
- Un model reușit va genera rezultate relativ similare în cazul ambelor tipuri de evaluare

Set de date	Performanță
Training	98%
Test	96%



Generalizarea modelului

- Generalizarea este capacitatea modelului de a obține aceleași rezultate de performanță pe date similare neîntâlnite anterior
- Lipsa generalizării modelului se reflectă în două efecte de evaluare:
 - Underfitting – învățare nereușită

Set de date	Performanță
Training	64%
Test	47%



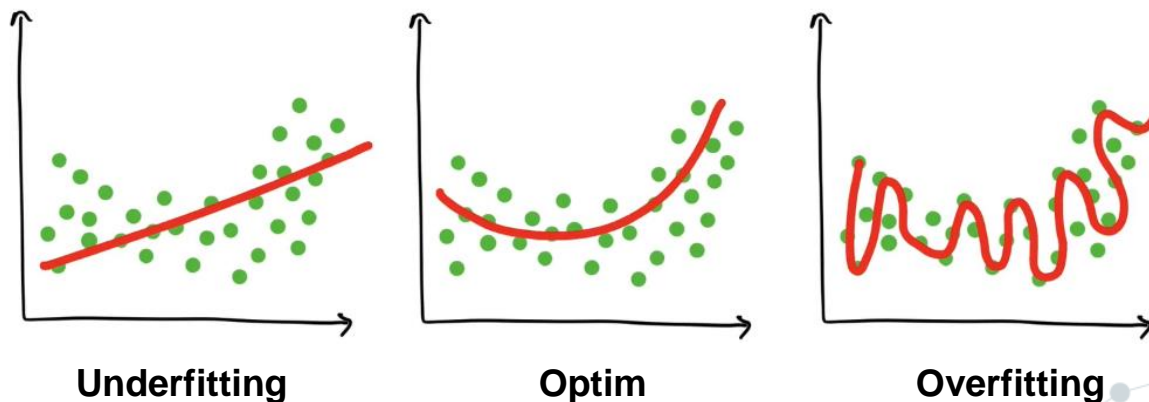
- Overfitting – supraînvățare

Set de date	Performanță
Training	99%
Test	67%



Underfitting vs overfitting

- Underfitting apare atunci când nu a reușit procesul de training, adică modelul nu a stabilit corect legitatea caracteristicilor și evaluarea de training a generat o performanță joasă
- Overfitting apare atunci când procesul de training a a fost realizat “prea” reușit pe datele de training, iar legitatea caracteristicilor nu poate fi generalizată pe datele neîntâlnite anterior



Excluderea underfitting și overfitting

- **Pentru excluderea underfitting este necesar:**
 - **Utilizarea unui algoritm mai avansat**
 - **Setarea hiper-parametrilor modelului**
 - **Reducerea numărului de caracteristici ale datelor**
 - **Creșterea timpului procedurii de training**
- **Pentru excluderea overfitting este necesar:**
 - **Creșterea volumului de date**
 - **Utilizarea unui algoritm mai puțin performant**

7. Experimentarea

Esența experimentării

- În etapa de experimentare se va răspunde la întrebarea “Pot să îmbunătățesc modelul elaborat?”
- Un proiect ML presupune elaborarea mai multor modele și selectarea celui optim reieșind din cerințele inițiale
- Experimentarea presupune elaborarea unui model simplu numit model de bază și creșterea treptată în complexitate a acestuia până la atingerea performanțelor dorite.
- Pentru atingerea performanțelor dorite uneori în cadrul experimentării se vor realiza toate etapele cadrului ML inclusiv revizuirea datelor și selectarea altui algoritm.
- La selectarea modelului optim în rezultatul experimentării se va ține cont și de compromisul performanță - timp

Mediul de lucru



Calculatorul
tău



matplotlib



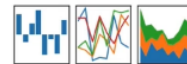
seaborn



NumPy

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



scikit
learn



dmlc
XGBoost



Instrumente de lucru

Data Science

Data Analysis

Machine Learning

Definirea
problemei

Studierea
datelor

Analiza
caracteristicilor

Elaborarea
modelului

Evaluarea
modelului

Experimentarea



pandas



dmlc
XGBoost

