Universitatea Tehnică ”Gheorghe Asachi” din Iași

Facultatea de Automatică și Calculatoare

Sisteme de Control Incorporate

Predicția Bolii Cardiace Folosind Învățare Automată

Studentă:Munteanu Maria-Daniela

Grupa: SCI_1B

Cuprins

1. Introducere

Bolile cardiovasculare (CVD) reprezintă principala cauză de deces la nivel global, fiind responsabile pentru aproximativ 17,9 milioane de vieți pierdute anual, ceea ce echivalează cu 31% din totalul deceselor mondiale. Majoritatea acestor decese (4 din 5) sunt provocate de infarct miocardic și accident vascular cerebral, iar o treime dintre ele apar prematur, afectând persoane sub 70 de ani. Aceste afecțiuni pot duce la complicații severe, precum infarct miocardic, insuficiență cardiacă și alte tulburări cardiovasculare. În acest context, prevenția și diagnosticul precoce joacă un rol esențial în îmbunătățirea calității vieții și în reducerea numărului de decese cauzate de aceste boli.

În ultimele decenii, progresele în domeniul tehnologiei și al științelor computaționale au permis dezvoltarea unor modele de învățare automată care pot analiza seturi mari de date pentru a identifica tipare relevante. Învățarea automată oferă metode puternice pentru diagnosticare, contribuind la dezvoltarea unor sisteme care pot prezice riscul de boală cardiacă pe baza caracteristicilor medicale ale fiecărui pacient. Spre deosebire de metodele tradiționale de diagnostic, care necesită interpretarea manuală a datelor de către medicii specialiști, modelele de învățare automată pot automatiza acest proces, reducând timpul de analiză și crescând precizia predicțiilor.

Acest studiu explorează utilizarea a două metode de clasificare pentru predicția bolii cardiace: K-Nearest Neighbors (K-NN) și Random Forest (RF). Vom analiza structura datasetului, metodele de preprocesare aplicate, performanțele acestor modele și concluziile care pot fi extrase din analiza comparativă a acestora.

2. Problema propusă

Diagnosticul bolilor cardiovasculare este o provocare semnificativă în domeniul medical, întrucât implicațiile unui diagnostic greșit pot fi

critice. Detectarea corectă a pacienților cu risc ridicat este esențială pentru prevenirea complicațiilor și pentru stabilirea unui plan de tratament eficient. În mod tradițional, medicii se bazează pe simptome clinice, teste de laborator și imagistică medicală pentru a diagnostica bolile cardiovasculare. Cu toate acestea, astfel de metode necesită timp, expertiză și resurse financiare considerabile, iar interpretările pot fi uneori subiective.

Insuficiența cardiacă (afecțiune în care inima nu mai poate pompa suficient sânge pentru a satisface nevoile organismului) este o consecință frecventă a CVD și poate fi anticipată pe baza unor factori de risc precum hipertensiunea arterială, diabetul, nivelul ridicat de lipide în sânge sau existența unei afecțiuni cardiovasculare preexistente. Identificarea timpurie și gestionarea adecvată a acestor riscuri sunt esențiale pentru prevenirea complicațiilor severe, iar modelele de învățare automată pot oferi un suport important în acest proces, prin analiza unor caracteristici relevante și predicția probabilității de apariție a unei boli cardiace. Datasetul analizat conține 11 indicatori care permit modelului de machine learning să realizeze astfel de predicții, contribuind la îmbunătățirea diagnosticării și tratamentului.Acestea reflectă datele raportate de organizații internaționale de sănătate. Factorii de risc menționați, precum hipertensiunea arterială, diabetul și nivelul ridicat de lipide în sânge, sunt bine documentați în literatura medicală.

Utilizarea învățării automate pentru predicția bolii cardiace oferă oportunitatea de a reduce aceste limitări, folosind algoritmi avansați care analizează în profunzime seturile de date medicale și identifică tipare relevante. Scopul principal al acestui studiu este compararea performanțelor modelelor K-NN și Random Forest în clasificarea pacienților pe baza caracteristicilor medicale.

Prin intermediul acestei analize, vom încerca să determinăm care dintre cele două metode este mai eficientă pentru clasificarea pacienților în funcție de riscul de boală cardiacă, având în vedere parametrii de acuratețe, timp de execuție și stabilitatea modelului.

3. Setul de date și analiza echilibrului

3.1. Descrierea datasetului

Pentru acest studiu, am utilizat un set de date medicale care conține informații despre 918 pacienți. Fiecare pacient este caracterizat de 11 variabile medicale, iar variabila țintă (HeartDisease) indică dacă pacientul suferă de o boală cardiacă sau nu.

- **Variabila țintă:** HeartDisease

- **Distribuția claselor:**

- 508 pacienți bolnavi (HeartDisease = 1)
- 410 pacienți sănătoși (HeartDisease = 0)

Setul de date Heart Failure Prediction conține 11 caracteristici clinice care pot fi utilizate pentru a prezice riscul de insuficiență cardiacă. Aceste caracteristici sunt variabile medicale relevante pentru sănătatea cardiovasculară. Iată o explicație a principalelor etichete și valorilor din acest dataset:

- **Age (Vârsta)** – Vârsta pacientului în ani.
- **Sex (Sexul)** – 1 pentru bărbați, 0 pentru femei.
- **ChestPainType (Tipul durerii în piept)** – Indică tipul durerii toracice experimentate de pacient.
- **RestingECG (Electrocardiograma în repaus)**:

- ✓ **Normal** – Activitate electrică normală a inimii.
- ✓ **ST** – Anomalii ale undelor ST-T (inversiuni ale undei T sau modificări ale segmentului ST de peste 0.05 mV).
- ✓ **LVH** – Hipertrofie ventriculară stângă, indicând o posibilă îngroșare a peretelui inimii

- **Cholesterol (Colesterolul seric)** – Nivelul colesterolului în sânge.
- **FastingBS (Glicemia pe nemâncate)** – 1 dacă glicemia este peste 120 mg/dl, 0 dacă este sub această valoare.
- **RestingBP (Tensiunea arterială în repaus):** Măsoară presiunea sângelui în artere atunci când inima este în repaus. Valorile ridicate pot indica hipertensiune și un risc crescut de probleme cardiovasculare.
- **MaxHR (Frecvența cardiacă maximă atinsă)** – Măsurată în bătăi pe minut. Este un număr între 60 și 202 bpm (bătăi pe minut), indicând ritmul maxim atins de pacient în timpul testelor de efort.
- **ExerciseAngina (Angină indusă de efort)** – 1 dacă pacientul prezintă angină în timpul exercițiului, 0 dacă nu.

Cât despre angină, aceasta este o durere în piept cauzată de reducerea fluxului de sânge către inimă. Se poate manifesta ca o senzație de presiune, strângere sau arsură și poate fi un semn de boală coronariană. Există mai multe tipuri de angină:

- ✓ TA (Angină tipică) – Durerea clasică de inimă, apare de obicei la efort și dispare cu repaus sau medicamente.
- ✓ ATA (Angină atipică) – Durere în piept care nu are simptomele clasice ale anginei, poate fi cauzată de altceva.
- ✓ NAP (Durere non-anginală) – Durere care nu are legătură cu problemele cardiace, poate fi de la mușchi, stomac sau stres.

- ✓ ASY (Asimptomatic) – Persoana nu simte nicio durere în piept, dar poate avea o problemă cardiacă ascunsă.

- **Oldpeak (Depresiunea segmentului ST)** – Măsurată în mm, indică modificări ale electrocardiogramei în timpul exercițiului.
- **ST_Slope (Panta segmentului ST)** – Indică forma segmentului ST în ECG.

1. Up (Ascendentă) – Indică o creștere a segmentului ST, ceea ce poate fi un semn normal sau benefic.
2. Flat (Plat) – Segmentul ST rămâne constant, ceea ce poate fi asociat cu ischemia cardiacă.
3. Down (Descendentă) – Indică o scădere a segmentului ST, un posibil semn de boală coronariană.

Variabila țintă este **HeartDisease**, care indică prezența (1) sau absența (0) unei boli cardiace.

Aceste date au fost colectate din surse medicale și sunt utilizate pentru a antrena modele de machine learning care pot ajuta la detectarea timpurie a insuficienței cardiace

Cum a fost construit datasetul?

➡️ Au existat mai multe seturi de date independente despre boli de inimă, fiecare adunat separat, cu pacienți diferiți:

- Cleveland → 303 pacienți
- Hungarian → 294 pacienți
- Switzerland → 123 pacienți
- Long Beach VA → 200 pacienți
- Stalog (Heart) → 270 pacienți

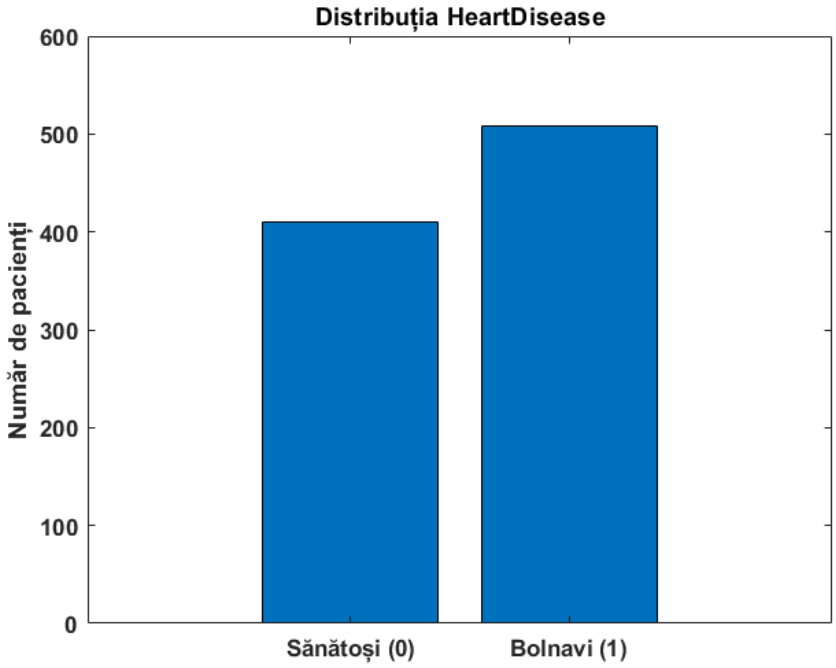
303 + 294 + 123 + 200 + 270 = 1190 observații totale (înainte de curățare).

Initial, in cadrul acestui set a fost observat că 272 rânduri erau duplicate (pacienți repetați sau date identice).
Astfel, data setul a fost modificat si au rezultat:
1190 inițial – 272 duplicate = 918 observații unice finale.

3.2. Analiza echilibrului datasetului

Un aspect important al analizei datasetului este echilibrul între clase, adică distribuția pacienților bolnavi și sănătoși. Dacă una dintre clase este dominantă, modelele de învățare automată pot fi influențate negativ, deoarece vor tinde să favorizeze predicțiile către clasa majoritară.

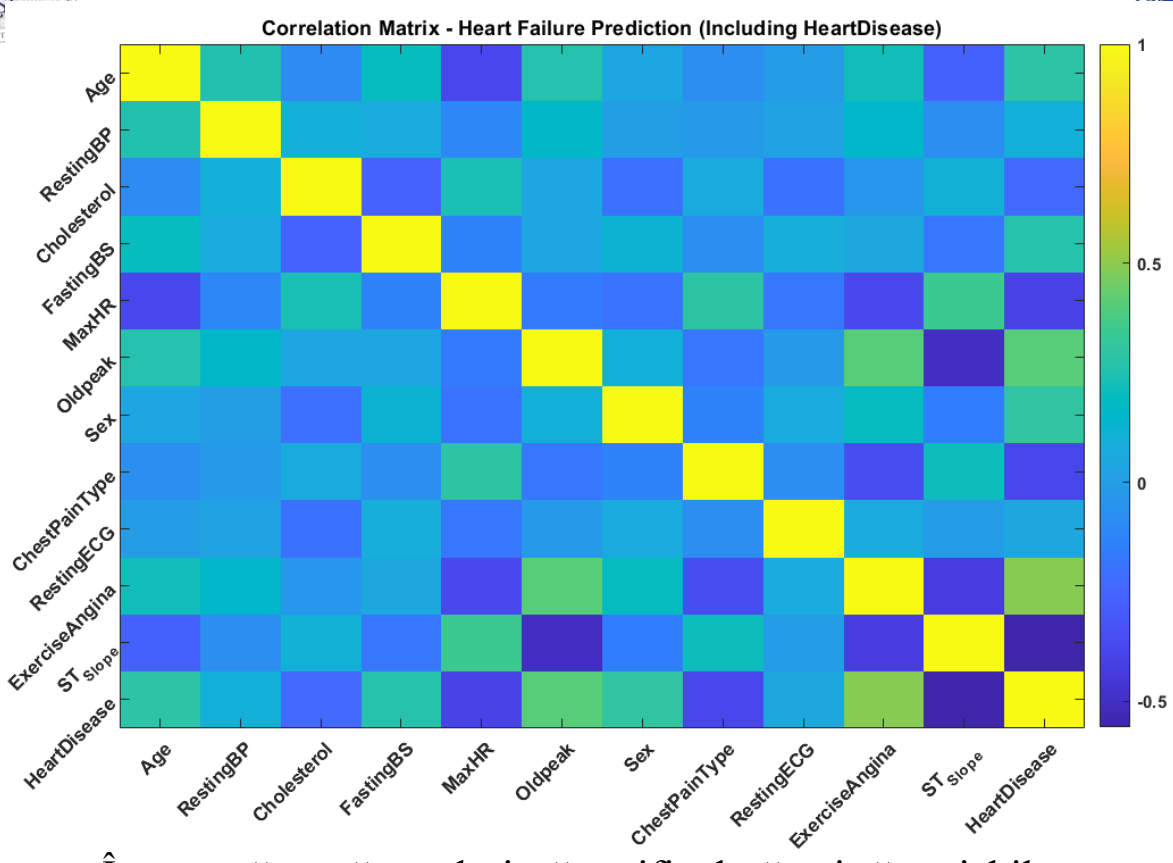
- Există un ușor dezechilibru între clase, deoarece avem mai mulți pacienți bolnavi (508) decât sănătoși (410).
- Acest dezechilibru poate afecta precizia modelului, în special dacă algoritmul învață să prezică mai frecvent boala cardiacă, ignorând pacienții sănătoși.



Matrice de corelație este o reprezentare vizuală a relațiilor liniare dintre variabilele din dataset, unde fiecare celulă indică coeficientul de corelație între două trasături. Prin analizarea acestei matrici se poate observa dacă anumite variabile sunt foarte strâns corelate (suprapuse sau redundante) sau dacă relațiile dintre ele sunt echilibrate. A fost calculată pentru un set de date destinat predicției insuficienței cardiace, inclusiv prezența bolii cardiace. Fiecare celulă din matrice reprezintă coeficientul de corelație, măsurând relația liniară dintre cele două variabile corespunzătoare.

Culorile și semnificația lor:

- Culorile calde (tentă spre galben) indică o corelație pozitivă puternică, adică atunci când una dintre variabile tinde să crească, cealaltă tinde, de asemenea, să crească.
- Culorile reci (tentă spre albastru) reflectă o corelație negativă, ceea ce înseamnă că o creștere a unei variabile este asociată cu o scădere a celeilalte.
- Nuanțele neutre (verde sau alte tonuri intermediare) semnifică relații slabe sau inexistentă între cele două variabile.

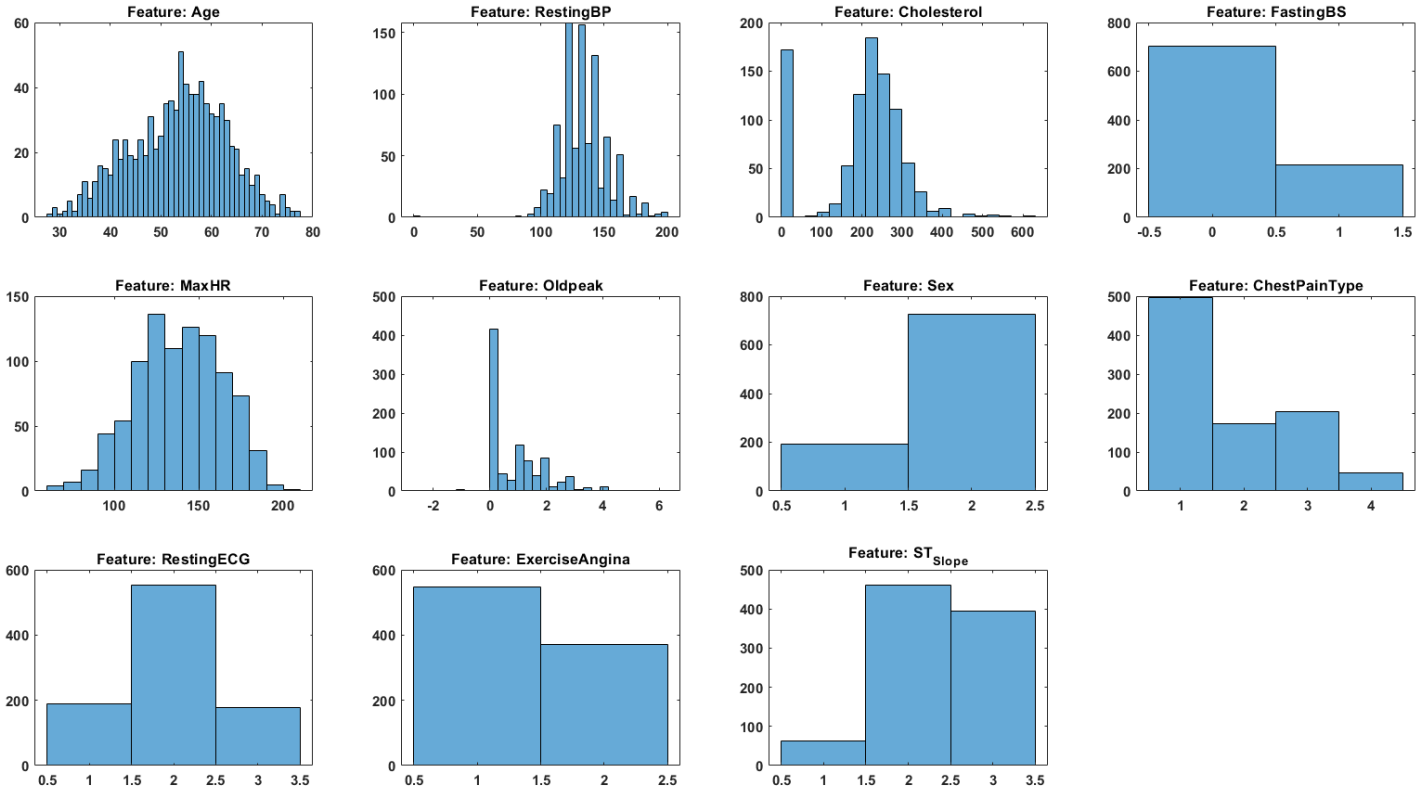


În această etapă am dorit să verific dacă există variabile predictoare care au o corelație extrem de ridicată (peste pragul de 0.9), ceea ce ar putea indica redundanță și ar putea afecta performanța modelului. Pentru aceasta am calculat matricea de corelație a tuturor predictorilor din matricea X și am căutat, prin folosirea funcției find pe partea superioară a matricei (cu triu), perechile de variabile care îndeplineau condiția de corelație puternică. De asemenea, am calculat 2 matrice de corelatie, una incluzand targetul, si alta doar cu predictorii pentru a observa modul în care fiecare caracteristică se leagă de prezența bolii cardiace , utilă pentru a înțelege importanța predictorilor si prin excluderea variabilei țintă, ne concentrăm pe relațiile dintre caracteristicile explicative. Astfel, putem identifica coliniaritatea între predictori, ceea ce poate determina eliminarea variabilelor redundante.

Rezultatele au arătat că niciun coeficient absolut de corelație între predictorii analizați nu a depășit pragul de 0.9. Aceasta înseamnă că variabilele din setul nostru sunt relativ independente și nu prezintă o

multicoliniaritate atât de puternică încât să necesite eliminarea vreuneia dintre ele.

Si am ilustrat reprezentarea trasaturilor in urmatoarea figura:



Motivarea alegerii acestor algoritmi

Alegerea K-Nearest Neighbors (K-NN) și Random Forest pentru analiza bolii cardiace si rezolvarea problemei de clasificare a depins de mai mulți factori precum complexitatea datelor, interpretabilitatea și performanța algoritmului:

K-NN

- ✓ **Implementare Rapidă:** Ușor de implementat și interpretat, fără setări sau antrenamente complexe, ideal ca model de baza – baseline

- ✓ **Analogii în Medicină:** K-NN compară un pacient cu cazuri similare din trecut, așa cum fac medicii pentru a recunoaște tipare clinice.
- ✓ **Non-Parametric:** Algoritmul nu impune ipoteze despre relații liniare sau distribuții normale, fiind flexibil pentru diverse tipuri de date.
- ✓ **Eficiență pe Seturi de Date Mici:** Funcționează foarte bine chiar și cu un volum redus de date, fiind ideal ca model de bază.
- ✓ **Flexibilitate în Distanțe:** Permite alegerea unor măsuri diferite de similaritate (ex.: Euclidean, Manhattan, Cosine,etc.) pentru a adapta analiza la specificul problemei.
- ✓ **Detectare de Tipare Locale:** Identifică pacienți cu caracteristici similare, ajutând la evidențierea unor posibile diagnostice pe baza datelor anterioare.

Random Forest

- ✓ **Captarea Relațiilor Complexe**: Random Forest poate detecta relații complexe și neliniare între trasături, esențiale pentru analiza datelor medicale variate.
- ✓ **Robustețe la Zgomot și Outlieri**: Modelul este robust în prezența datelor zgomotoase sau a outlier-urilor, asigurând fiabilitatea deciziilor.
- ✓ **Explicabilitate Parțială**: Permite identificarea trasăturilor care influențează cel mai mult decizia, facilitând interpretarea rezultatelor din punct de vedere clinic.
- ✓ **Reducerea Riscului de Overfitting**: Combinarea mai multor arbori de decizie ajută la generalizarea modelului și la evitarea suprapotrivirii.
- ✓ **Gestionarea Valorilor Lipsă**: Funcționează bine și cu date incomplete, o caracteristică importantă în aplicațiile medicale.

✓ **Performanță Ridicată în Clasificare**: În probleme precum diagnosticarea bolilor cardiace (0 = sănătos, 1 = bolnav), Random Forest poate oferi o acuratețe superioară, fiind o alegere eficientă pentru clasificare binară.

4. Metodele utilizate

4.1. K-Nearest Neighbors (K-NN)

Algoritmul **K-Nearest Neighbors (K-NN)** este unul dintre cele mai utilizate modele de clasificare bazate pe similaritate. Principiul său de funcționare este simplu: un nou punct de test este clasificat în funcție de cei mai apropiați K vecini din setul de antrenare.

Datele:

- Fie $x=(x1,x2,\ldots,x_n)\in R^n$ punctul necunoscut (exemplul de test).
- Setul de antrenare este $\{(x_i,y_i)\}_{i=1}^{N}$, unde $x_i\in R^n$ sunt vectorii de caracteristici și $y_i \in C$ etichetele de clasă, iar C este mulțimea tuturor claselor.
- Toate datele sunt prealabil normalizate, astfel încât diferențele de scară nu distorsionează calculul distanței.

Calculul distanței:

Se folosește distanța euclidiană pentru a măsura apropierea dintre x_i și fiecare y_i:

$$d(x,\ y)\ =\ \sqrt{\sum_{i=1}^{n}(y_i - x_i)^2}$$

Selectarea celor k vecini: Se alege un k impar pentru a evita egalitățile. Pentru fiecare exemplu necunoscut x, se calculează distanța față de toate exemplele de antrenare și se formează mulțimea

$N_k(x)=\{i\in\{1,2,\ldots,N\}$: d(x,y) este printre cele mai mici k valori$\}$.

Decizia Finală**:**

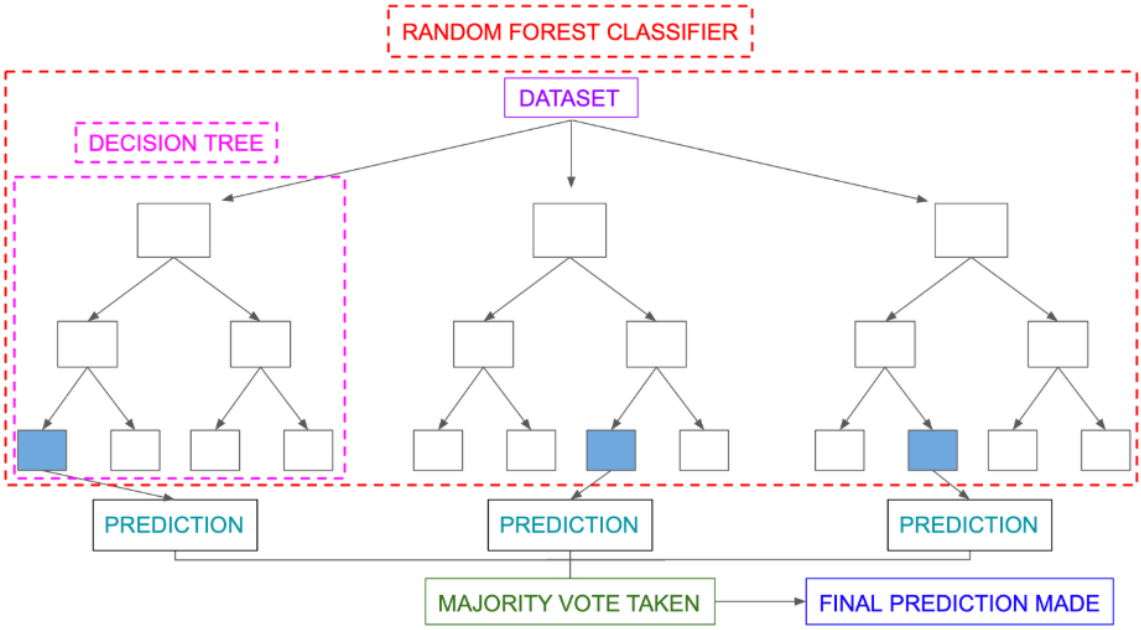
$f(x)= \sum_{i\in Nk(x)} 1\{y_i=c\}$

- C reprezintă mulțimea tuturor claselor posibile;
- $N_k(x)$ este mulțimea indicilor celor k vecini cei mai apropiați de x;
- y_i este eticheta asociată cu exemplarului x_i din setul de antrenare;
- Funcția indicator $1\{y_i=c\}$ -> daca $y_i=c$ functia returneaza 1 , altfel 0.

Se atribuie lui x clasa majoritară dintre etichetele celor k vecini- pentru fiecare clasă C, se numără câți dintre cei k vecini au eticheta C. Clasa pentru care această sumă este maximă devine predicția finală pentru punctul x.

4.2. Random Forest

Random Forest reprezintă un algoritm de învățare automată ce se bazează pe un ansamblu de arbori de decizie. Ideea fundamentală a algoritmului este agregarea predicțiilor realizate de mai mulți arbori de decizie, construiți pe diverse subseturi ale datelor de antrenament și cu selecția aleatorie a caracteristicilor, pentru a obține o performanță globală robustă și precisă. Prin combinarea diverselor „opinie” (predicții) ale arborilor individuali, Random Forest reduce riscul de supra-antrenare (overfitting) și îmbunătățește stabilitatea modelului.



Algoritmul Random Forest – Pași Principali

1. **Bootstrap Sampling**
    - Se selectează aleatoriu, cu înlocuire (bootstrap sampling- la selectarea unui element, acesta este "păstrat" în setul de date, așa că poate fi extras de mai multe ori în subsetul final), un subset de date de antrenament pentru fiecare arbore de decizie. Astfel, fiecare arbore este antrenat pe un set de date ușor diferit, crescând diversitatea și reducând riscul de overfitting.
2. **Selecția Aleatorie a Caracteristicilor**

    Pentru fiecare arbore și pentru fiecare nod intern al acestuia, în loc să se considere toate caracteristicile p disponibile pentru găsirea split-ului optim, se extrage un subset aleatoriu din caracteristici, de obicei de dimensiunea m (cu $m < p$). În problemele de clasificare, o alegere comună este $m=\sqrt{p}$, iar pentru regresie $m=p/3$ (sau alți parametri bazați pe validări empirice).

Construirea Arborilor fără Pruning

- Fiecare arbore este construit complet, fără aplicarea unei proceduri de „pruning” (tăierea unor ramuri), astfel încât să maximizeze învățarea datelor.

3. **Repetarea Procesului**
   - Pașii 1-2 se repetă pentru **n arbori** (unde „n” este un parametru stabilit de utilizator). În general, mai mulți arbori duc la o performanță mai bună și la o stabilitate crescută a modelului.
4. **Predicția Finală – Vot Majoritar**
   - Fiecare arbore de decizie realizează o predicție independentă. În cazul problemelor de clasificare, fiecare arbore votează, iar clasa cu cele mai multe voturi devine rezultatul final. În cazul regresiei, predicția finală este obținută prin medierea valorilor estimate de arbori.
   - **(Clasificare):** Pentru un exemplu de test necunoscut x, fiecare arbore din ansamblu produce o predicție $y^{(i)}$ (unde i=1,2,…,n). Predicția finală f(x) este obținută prin votul majoritar:

$$f(x) = \sum_{i=1}^{n} 1\{y^{(i)} = c\}$$

     unde C este mulțimea claselor și $1\{y^{(i)}=c\}$ este funcția indicator.

5. **Criteriul de Împărțire – Gini Impurity**

- La fiecare nod, pentru a determina split-ul optim, se utilizează criteriul de impuritate Gini. La fiecare nod, algoritmul selectează caracteristica care minimizează **impuritatea Gini**. Această impuritate măsoară cât de „amestecate” sunt clasele într-un anumit nod. Formula este:

$$Gini(t) = 1 - \sum_{i=1}^{C} p_i^2$$

unde p_i este proporția observațiilor din clasa i în nodul t, iar C este numărul total de clase.

7. **Alegerea Caracteristicii Optime**
   - Algoritmul caută trasătura care minimizează **Gini Gain** – diferența dintre impuritatea Gini înainte și după împărțire – pentru a asigura că fiecare împărțire maximizează separarea corectă a claselor. Minimarea impurității Gini la fiecare split asigură o separare cât mai clară a claselor, contribuind la performanța și precizia modelului. Alegerea acestei măsuri are avantajul de a fi computațional eficientă și interpretabilă.

5. Compararea modelelor și analiza performanței

În etapa de evaluare a modelelor, se urmărește nu doar antrenarea și obținerea predicțiilor, ci și analiza aprofundată a modului în care diferite configurații de algoritmi influențează performanța pe datele de test. Algoritmii de tip K-Nearest Neighbors (K-NN) și Random Forest au

fost implementați cu diverse configurații pentru a determina care este abordarea cea mai eficientă pentru problema clasificării bolii cardiace.

Pentru K-NN, s-au antrenat două modele distincte: unul care folosește 5 vecini și altul cu 10 vecini. Alegerea unui număr diferit de vecini afectează direct compromisurile între sensibilitate (recall) și precizie: un număr mai mic de vecini poate conduce la o reacție mai sensibilă, dar și la o instabilitate sporită în prezența zgomotului, în timp ce un număr mai mare de vecini tinde să netezească predicțiile, oferind o acuratețe globală ușor îmbunătățită.

În paralel, pentru Random Forest, s-au evaluat două configurații: o implementare ce utilizează 50 de arbori și alta cu 200 de arbori. Pe lângă îmbunătățirea stabilității prin medierea predicțiilor individuale, numărul de arbori influențează complexitatea și capacitatea modelului de a capta relațiile subtile din date. Rezultatele indică faptul că, odată cu creșterea numărului de arbori, performanța se stabilizează, iar diferențele între modelul cu 50 și cel cu 200 de arbori devin minore, sugerând că metoda ajunge la un prag de saturare din punct de vedere al performanței.

După antrenarea fiecărui model pe setul de antrenare și generarea predicțiilor pe setul de test, s-au construit matrici de confuzie care au permis calculul unor metrice esențiale: precizia, recall-ul, scorul F1 și acuratețea. Aceste metrici oferă o imagine completă asupra modului în care fiecare model reușește să discrimineze între clasele de pacienți, evidențiind astfel compromisurile între ratele de eroare de tip fals pozitiv și fals negativ.

◆ Acuratețea

Proporția de predicții corecte din totalul predicțiilor. Exprimă cât de des modelul face predicții corecte.

$$\text{Accuracy} = \frac{TP+TN}{\text{TP+TN+FP+FN}}$$

Precizia pentru clasa 1

Proporția de instanțe prezise ca fiind pozitive (clasa 1) care sunt cu adevarat pozitive. Oferă detalii despre cât de sigur este modelul atunci când prezice o anumită clasă.

$$\text{Precision} = \frac{TP}{TP+FP}$$

Recall (Sensibilitate / Rata de detecție) – pentru clasa 1

Proporția de instanțe pozitive reale (clasa 1) care au fost corect identificate de model. Evidențiază cât detectează modelul clasa 1.

$$\text{Recall} = \frac{TP}{TP+FN}$$

F1-Score – pentru clasa 1

Media armonică dintre precizie și recall. Este un compromis între ele și este utilă mai ales când ai date dezechilibrate.

$$F1 = 2 \cdot \frac{Precision * Recall}{Precision+Recall}$$

Mai jos, în tabelul de mai jos, sunt prezentate valorile obținute pentru fiecare model:

Model	Precision	Recall	F1 Score	Accuracy
K-NN (5 vecini)	0.83654	0.89691	0.86567	0.85246
K-NN (10 vecini)	0.88542	0.87629	0.88083	0.87432
Random Forest (50 arbori)	0.84615	0.90722	0.87562	0.86339
Random Forest (200 arbori)	0.85294	0.89691	0.87437	0.86339

- K-NN (5 vecini): Prezintă un echilibru bun între precizie și recall, însă acuratețea este ușor inferioară comparativ cu versiunea cu 10 vecini.
- K-NN (10 vecini): Atinge o valoare mai ridicată a preciziei și acurateții, sugerând că incluzând mai mulți vecini se obține o estimare mai stabilă și mai robustă.
- Random Forest (50 arbori): Oferă o performanță remarcabilă, cu un recall foarte ridicat, ceea ce sugerează că majoritatea cazurilor pozitive sunt identificate corect.
- Random Forest (200 arbori): Valorile sunt foarte similare cu cele ale modelului cu 50 de arbori, indicând stabilizarea performanței pe măsură ce numărul de arbori crește

6. Concluzii și direcții de îmbunătățire

Comparând aceste rezultate, se evidențiază faptul că, în cazul problemei studiate, atât metoda K-NN, cu o ajustare adecvată a numărului de vecini, cât și Random Forest, cu un număr rezonabil de arbori, oferă performanțe competitive. Totuși, configurația K-NN cu 10

vecini pare să aibă un mic avantaj în acuratețe, în timp ce Random Forest demonstrează o capacitate foarte bună de a capta cazurile pozitive datorită strategiilor de bootstrap sampling și selecție aleatorie a caracteristicilor.

Compararea acestor metode evidențiază faptul că, în contextul problemei studiate, se poate atinge un echilibru optim între precizie și recall atât prin reglarea parametrilor pentru K-NN, cât și prin alegerea unui număr corespunzător de arbori în Random Forest. Rezultatele obținute indică o performanță solidă pentru ambele tipuri de algoritmi, fiecare având puncte forte care pot fi valorificate în funcție de cerințele specifice ale aplicației.

Direcții de Îmbunătățire:

Optimizarea Parametrilor: Folosirea validare încrucișată pentru a identifica valoarea optimă a lui k la K-NN și numărul ideal de arbori la Random Forest.

Preprocesare Avansată: Experimentarea unor metode mai sofisticate pentru completarea valorilor lipsă și transformarea variabilelor cu distribuții asimetrice

Combinarea Modelelor: combinarea predicțiilor mai multor modele prin tehnici ensemble (de exemplu, stacking) pentru a obține predicții mai robuste.

Bibliografie

https://www.mdpi.com/1999-4893/17/2/78

https://link.springer.com/article/10.1007/s42979-023-02529-y

https://revista.cardioportal.ro/arhiva/characteristics-of-patients-with-heart-failure-from-romania-enrolled-in-esc-hf-long-term-esc-hf-lt-registry/

https://ino-med.ro/docs/document-de-pozitie-insuficienta-cardiaca.pdf

https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction

Material de curs: Tehnici de învățare automată. Moodle - https://edu.tuiasi.ro/