Universitatea Transilvania din Brașov

Universitatea Transilvania din Brașov
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

ȘCOALA DOCTORALĂ INTERDISCIPLINARĂ

Facultatea de Matematică și Informatică

Árpád KERESTÉLY

Modele de Machine Learning în Predicția Cancerului

REZUMAT

Conducător științific

Prof. Dr. Marius-Sabin TĂBÎRCĂ

BRAȘOV, 2022

Tema tezei de doctorat și domeniile în care se încadrează

Teza prezintă cercetarea și rezultatul unui subiect interdisciplinar, utilizarea informaticii în domeniul medical, și anume tema predicției cancerului folosind învățarea automată, atingând astfel domenii din ambele. Din perspectiva informaticii, domeniul învățării automate este piesa centrală, care aduce cu sine multe dintre domeniile strâns conectate, cum ar fi știința datelor, extragerea datelor, calculul de înaltă performanță sau volume mari de date. Din punct de vedere al domeniului medical, elementul central este domeniul predicției cancerului, cu accent pe sarcina molară și cancerul de sân, dar domeniul părinte, cel al domeniului medical este, de asemenea, revizuit în acest proces. Astfel, tema tezei se integrează bine în domenii atât din informatică, cât și din domeniul medical.

Focusând pe problemele și soluțiile tratate din perspectiva informaticii, merită menționat faptul că seturile de date utilizate au fost tabelare și au fost abordate probleme precum clasificarea și calculul seriilor de timp, astfel algoritmi precum clasificatorul de regresie logistică, pădurea aleatorie, rețelele neuronale artificiale, rețelele neuronale recurente sau metoda celor mai mici pătrate au fost principalele surse de interes în găsirea soluțiilor. Figura 1 prezintă un exemplu de măsurători în contextul sarcinii molare, pentru care problema constă în găsirea unei curbe de regresie care se potrivește cel mai bine cu datele, folosind doar un sub-eșantion al primelor date de intrare.

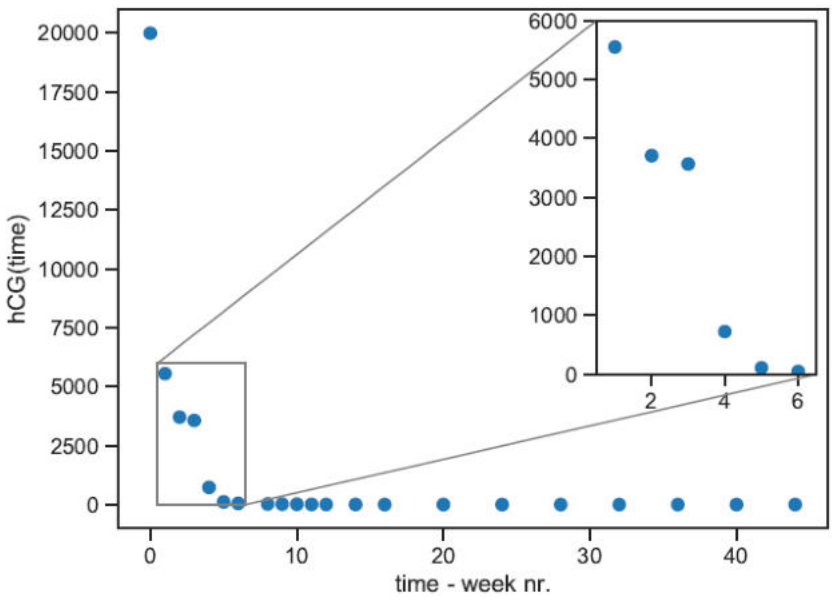


Figura 1: Un eșantion din măsurătorile hCG privind sarcina molară.

Obiectivele cercetării

Cercetarea în contextul acestei teze a fost inițiată de necesitatea unei soluții precise și automatizate pentru problema determinării devoluției cancerului la pacienții diagnosticați și

tratați cu sarcină molară. Un exemplu de astfel de problemă și posibilă soluție poate fi văzut în Figura 2, unde măsurătorile hCG din primele cinci săptămâni (puncte albastre) ale pacientului au fost utilizate pentru a prognoza devoluția cancerului (reprezentată de curba albastră) și, după cum se poate observa, curba se potrivește cu datele reale (puncte portocalii). Astfel, obiectivul principal al tezei a fost găsirea de soluții la problema menționată mai sus. Ca obiective secundare, teza și-a propus să găsească și să acopere nișe în literatura de informatică care nu au fost încă acoperite, precum și să adune un set de instrumente și procese care pot fi utilizate de cercetători atunci când lucrează cu predicția cancerului, dar și cu alte tipuri de probleme care implică clasificarea sau calculul seriilor de timp folosind date tabelare.

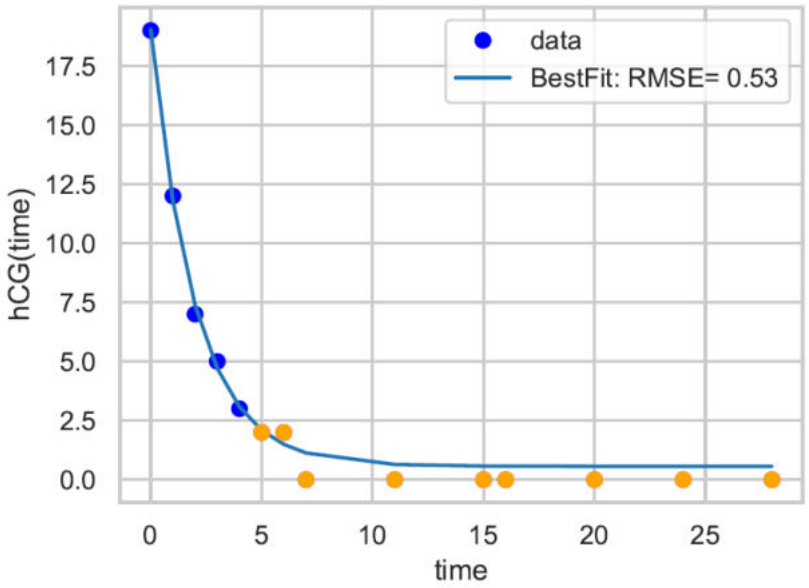


Figura 2: Prognozarea devoluției cancerului folosind primele cinci valori de intrare.

Structura tezei

Teza este alcătuită din șase capitole.

Primul capitol este "Introducere" care conține o descriere detaliată a: motivațiilor care au stat la baza începerii acestei teze, obiectivelor, rezultatelor cercetării împreună cu lucrările publicate în domeniul cercetării și structurii tezei.

Al doilea capitol se intitulează "Învățarea automată în domeniul medical și al predicției de cancer" și analizează modele și tehnici de învățare automată de ultimă generație utilizate în diverse probleme din domeniul medical și două domenii specifice de predicție a cancerului, și anume cancerul de sân și sarcina molară.

Al treilea capitol, numit "Date, preprocesarea datelor și gestionarea trăsăturilor", prezintă dificultățile în obținerea accesului la datele medicale, dar prezintă, de asemenea, diferite surse în care pot fi găsite date publice privind cancerul, care sunt procedurile de lucru tipice de explorare a datelor, ce metode sunt utilizate pentru tratarea sau calcularea valorilor lipsă,

ponturi pentru îmbunătățirea unui set de date cu trăsături suplimentare, precum și eliminarea trăsăturilor care nu sunt necesare pentru a reduce complexitatea seturilor de date, toate în contextul predicției cancerului de sân și al sarcinii molare. Pentru a susține teoriile, experimentale au fost realizate pe date reale, în cazul ambelor variante de cancer.

Contribuția principală a acestei cercetări se regăsește în "Serii de timp și pronosticuri în predicția cancerului", al patrulea capitol al tezei, unde un număr substanțial de rezultate sunt prezentate pe setul de date privind sarcina molară. Acest capitol prezintă o nouă metodă pentru potrivirea unei curbe pe puncte de date care descresc cu o tendință exponențială, această metodă este comparată și evaluată cu alte metode existente și sunt prezentate aplicații ale metodei, cum ar fi crearea de prognoze sau detectarea anomaliilor. În a doua parte a acestui capitol, rețelele neuronale recurente (RNR) sunt utilizate pentru a prezice devoluția bolii unui pacient pe baza datelor altor pacienți, împreună cu un set de proceduri care pot ajuta RNR-urile să funcționeze mai bine atunci când datele sunt într-un număr limitat.

Capitolul cinci, "Calculul de înaltă performanță în contextul volumelor mari de date și al învățării automate", analizează metodele de calcul de înaltă performanță existente disponibile pentru învățarea automată și Big Data, în timp ce în a doua parte, două biblioteci populare din domeniul învățării automate sunt comparate între ele, iar performanțele lor sunt evaluate. Acest capitol își propune să prognozeze timpul și complexitatea resurselor necesare pentru rularea unei clasificări pe un set de date mai mare privind cancerul de sân, dar deoarece la scrierea acestei teze datele privind cancerul de sân nu au fost disponibile în cantități mari, a fost adoptată o soluție de avarie, în care algoritmii au fost executați pe un set de date cu format similar, dar de dimensiuni mai mari, din domeniul detectării defectelor la rulmenți.

Ultimul capitol al tezei, "Concluzii", rezumă contribuțiile pe care această teză le-a adus comunității de cercetare și prezintă direcțiile de cercetare viitoare.

Metodologia de cercetare

Cercetările existente pentru tema principală a tezei, și anume învățarea automată în predicția devoluției bolii la pacienții tratați cu sarcină molară, sunt destul de rare, astfel încât cercetarea tezei a început cu revizuirea stadiului actual al domeniul medical, continuând cu un tip răspândit de cancer la femei, cancerul de sân. S-a pornit de la ideea de a găsi probleme și soluții care sunt similare în natură cu cea privind sarcina molară. Capitolul 2 descrie toate constatările din toate cele trei domenii și colectează o suită de instrumente și procese care pot fi aplicate de obicei în domeniul predicției cancerului.

Capitolul 3 continuă cu studiul atât al cancerului de sân, cât și al sarcinii molare, dar de data aceasta, prin aplicarea cunoștințelor colectate anterior cu privire la seturi de date concrete. În primul rând, se aduc noi contribuții la un set de date privind cancerul de sân disponibil public, efectuând o analiză exploratorie profundă, aplicând selectarea manuală și automată de

trăsături, și prezentând rezultatele ca urmare a rulării clasificatorilor de regresie logistică și pădure aleatorie. În continuare, se aduc contribuții la un set privat de date privind sarcina molară. Ca și în cazul anterior, s-a efectuat o analiză exploratorie profundă, precum și o selecție manuală a trăsăturilor. În plus, au fost discutate metode de tratare a valorilor lipsă și au fost propuse transformări necesare pentru a aduce setul de date într-un format compatibil cu învățarea de secvențe. Aceste experimente au permis concentrarea exclusiv pe partea de învățare automată în capitolele următoare.

Capitolul 4 este o continuare a secțiunii capitolului 3 privind sarcina molară. În acest capitol accentul principal este pe găsirea de soluții pentru prezicerea devoluției cancerului la pacienții tratați de sarcina molară. Acest lucru se realizează, în primul rând prin aplicarea cunoștințelor de domeniu, i.e., că măsurătorile hCG la pacienții tratați cu sarcină molară scad pe baza unei curbe exponențiale deplasate vertical. Această curbă este unică pentru fiecare pacient, astfel încât soluțiile implică utilizarea primelor trei măsurători ale unui pacient în determinarea parametrilor curbei. Curba rezultată permițând prognozarea devoluției bolii. În continuare, a fost investigată o abordare folosind rețele neuronale recurente, în care scopul a fost de a învăța natura descrescătoare a măsurătorilor folosind date de la toți pacienții. Acest lucru a permis evitarea încorporării cunoștințelor de domeniu în soluție, permițând astfel învățarea și predicția cazurilor în care boala a recidivat.

Capitolul 5 este o perpetuare a secțiunii capitolului 3 referitoare la cancerul de sân. Ideea din spatele acestui capitol a fost că, la un moment dat, cantități masive de date vor fi disponibile pentru cercetare în domeniul cancerului de sân, fiind unul dintre cele mai studiate tipuri de cancer. În pregătirea pentru această schimbare, a fost evident faptul că soluția utilizată în capitolul 3 nu va fi capabilă să gestioneze volumul mare de date, așa că a fost necesară o schimbare la o soluție de calcul mai scalabilă și de înaltă performanță. Având această viziune, a fost efectuată o revizuire a soluțiilor de calcul de înaltă performanță existente care implică învățarea automată. Spark a fost aleas ca soluție, deoarece poate gestiona date tabulare prin definiție, iar API-ul este foarte similar cu API-ul scikit-learn, cadrul utilizat în capitolul 3 pentru studiul setului de date privind cancerul de sân. Astfel, a doua parte a capitolului se axează pe analiza comparativă a acestor două biblioteci, în scopul de a evalua în primul rând performanța Spark în raport cu performanța scikit-learn, și în al doilea rând, pentru a pregăti terenul pentru momentul în care datele de cancer de sân vor fi disponibile în cantități mari. Pentru a putea efectua comparația, un set mare de date tabelar a fost folosit ca și înlocuitor, din domeniul detectării defectelor la rulmenți. Acest set este foarte similar ca structură cu setul de date privind cancerul de sân și este etichetat și pentru probleme de clasificare.

Rezultatele originale, concluziile, contribuții la domeniul științific și relevanța

Capitolul 2: Învățarea automată în domeniul medical și al predicției de cancer

2.1 Introducere

Acest capitol își propune să prezinte, să analizeze și să discute unele dintre cele mai recente progrese în învățarea automată din punct de vedere al domeniului medical și al predicției cancerului. Aspecte importante pe care le acoperă acest capitol sunt algoritmii de învățare automată utilizați recent, datele disponibile în scopuri de cercetare și domeniile în care se extind asistența medicală și predicția cancerului. Se trage o concluzie bazată pe aceste aspecte, dacă există o nevoie și o posibilitate de dezvoltare potențială în continuare a algoritmilor de învățare automată în domeniul medical și al predicției cancerului.

Metodologia aplicată în analizarea literaturii este compusă din metode bine cunoscute. În primul rând, șirurile de căutare au fost compuse din următoarele cuvinte cheie: domeniul medical, prevenirea și predicția cancerului, cancerul de sân, sarcina molară, boala trofoblastică gestațională, învățarea automată, extragerea datelor, seriile de timp, prognoza. Apoi, șirul de căutare a fost folosit în următoarele baze de date pentru a găsi studii relevante: Google Scholar, Elsevier Scopus, Elsevier Science Direct și Web of Science. Aceste baze de date au fost alese datorită numărului mare de discipline în care sunt indexate. Lista studiilor de cercetare adunate în acest fel, a fost filtrată, pentru a le obține pe cele mai relevante din punct de vedere al științei datelor și al învățării automate. Mai întâi acestea au fost filtrate după titlu. Dacă titlul părea suficient de bun sau dacă era cel puțin ambiguu, rezumatul acelei cercetări a fost analizat. Lucrările care s-au dovedit a fi relevante după verificarea rezumatului au fost trecute printr-o altă rundă de filtrare bazată pe introducere și concluzii. În cele din urmă, acele lucrări de cercetare care încă păreau relevante, au fost citite de la început până la sfârșit. Cele care au prezentat idei relevante, interesante, inovatoare și diverse au fost raportate în cele ce urmează.

2.2 Domeniul medical

Este interesant de observat că cercetătorii din întreaga lume sunt interesați de tema învățării automate legate de domeniul medical. Lucrările studiate sunt scrise în diferite părți ale lumii, dar împărtășesc același scop, de a aduce îmbunătățiri domeniului medical. De exemplu: [82] a fost realizat în Brazilia, [26] în California, [23] în China. Lucrările menționate mai sus și-au implementat, de asemenea, cercetările asupra datelor în regiunile în care au fost realizate (este

important de menționat că unele boli se pot manifesta în moduri diferite, în funcție și de locul în care au fost observate [23]).

Pentru majoritatea algoritmilor de învățare automată este importantă cantitatea de date cu care se lucrează. De obicei, cu cât datele sunt mai mari, cu atât soluția va fi mai exactă, după cum afirmă multe lucrări de cercetare, inclusiv cele din domeniul predicției cancerului [64]. Datele pot proveni din diverse surse: date mobile, fișe medicale, rețele sociale, utilizarea internetului, date genomice sau date de mediu.

Cert este că sunt necesare rezultate și soluții mai multe și mai bune în domeniul medical. Volumul datelor a crescut recent și probabil va crește și mai mult în viitor. Algoritmii de învățare automată au fost cercetați pentru a rezolva unele probleme specifice, dar există încă mult loc pentru îmbunătățiri, deoarece foarte puțini dintre aceștia abordează problemele cu o acuratețe ridicată și chiar mai puțini într-un mod generic. Niciunul dintre algoritmii de învățare automată utilizați nu este potrivit pentru mai mult de câteva probleme, în plus, după cum se vede în studiile menționate, niciunul dintre aceștia nu a apărut mai mult de două ori, ceea ce este un semn că probabil algoritmii mai buni sunt încă nedescoperiți. Având în vedere aceste fapte, viitorul domeniului medical pare promițător cu ajutorul învățării automate.

2.3 Cancerul de sân

Figura 3, Figura 2.1c din teză) iar, împreună cu celelalte tipuri de cancer, formează a doua cea mai letală boală din lume în rândul oamenilor [89]. S-a depus multe eforturi de cercetare pentru prezicerea și chiar prevenirea acesteia. În timp ce domeniul predicției cancerului a beneficiat foarte mult de învățarea automată în ultimele decenii deoarece multe lucrări de cercetare au abordat problema predicției cancerului, există încă loc de îmbunătățiri. Deoarece această teză se concentrează pe predicția cancerului, studiul cancerului de sân poate oferi o perspectivă valoroasă și puncte de plecare pentru dezvoltarea de soluții pentru problema menționată în introducere. Această secțiune își propune să revizuiască unele dintre cercetările existente din domeniul cancerului de sân, să adune un set de metode și procese utile și să le aplice în capitolul 3 pe un set de date privind cancerul de sân disponibil public, cu accent pe înțelegerea modului în care diferite caracteristici sau absența lor pot influența rezultatul unei predicții.

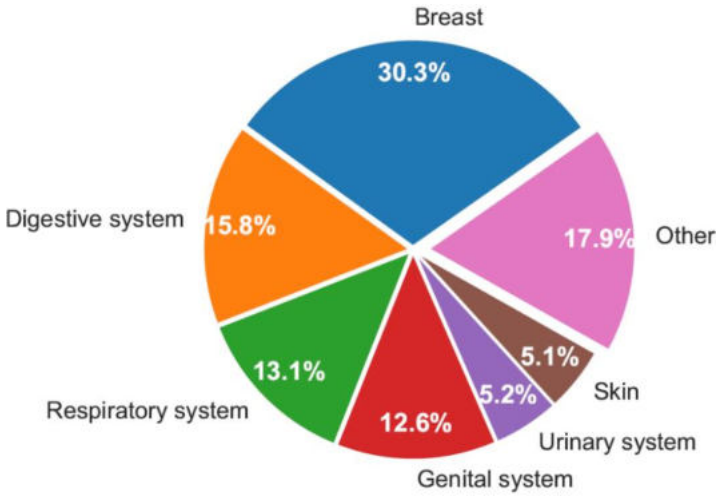


Figura 3: Estimarea noilor cazuri de cancer la femeile din SUA în funcție de tip, pe baza datelor din [89]

Cancerul de sân este unul dintre cele mai invazive și periculoase tipuri de cancer în rândul femeilor, după cum se poate observa în [89, 47], astfel încât atrage atenția multor cercetători. În ultimele decenii, s-au înregistrat progrese datorită algoritmilor mai noi de învățare automată. Totuși, există loc de îmbunătățit și mai mult. Întrucât învățarea automată necesită date din care să învețe, viteza cu care progresează această îmbunătățire depinde în mare măsură de cantitatea de seturi de date disponibile public. Din păcate și mai ales din cauza preocupărilor recente legate de confidențialitatea datelor, prea puține date sunt disponibile în spațiul public și cu atât mai puține sunt noi, astfel încât un cercetător fie folosește ceea ce este disponibil public, fie formează o echipă cu un institut medical. În cazul acestei teze, accentul se va pune pe cercetarea care se face pe datele disponibile public, astfel încât rezultatele raportate să poată fi verificate și îmbunătățite.

Un studiu recent al cancerului de sân realizat de autorii [4] raportează că pe Wisconsin Diagnostic Breast Cancer (WDBC) au obținut o acuratețe aproape perfectă, cu modelele cei mai apropriați K vecini, pădure aleatorie și perceptron multistrat. Pe lângă acuratețe, autorii au măsurat și scorul F. Deși nu a fost menționat de autori, după ce s-a încercat reproducerea rezultatelor, s-a descoperit că în experimentele lor, validarea rezultatelor a fost cel mai probabil făcută pe același set de date ca și instruirea, de unde și acuratețea foarte mare.

În [3] autorul a studiat mai mulți algoritmi de învățare automată: o combinație de unitate recurentă cu porți și mașină de suport vectorială, clasificator de regresie liniară, perceptron multistrat, căutare folosind metoda celor mai apropiați K vecini, regresie Softmax și mașină de suport vectorială pe același set de date WDBC. Înainte de a rula algoritmii de învățare automată, a fost aplicat un pas de preprocesare pentru a standardiza setul de date, și anume StandardScaler de la scikit-learn [79]. Setul de date inițial a fost împărțit în set de antrenare

de 70% și set de testare de 30%, iar fiecare algoritm testat a fost rulat cu o acuratețe de peste 90% pe setul de testare, perceptronul multistrat ajungând la o precizie de 99,04%.

Autorii articolului [72] au avut o abordare oarecum diferită de lucrările deja menționate. Ei au folosit analiza topologică a datelor pe setul de date privind cancerul de sân al Netherlands Cancer Institute (NKI), pentru a colecta informații din peste 1500 de caracteristici de expresie genetică. Cercetările lor au oferit perspective interesante asupra datelor prin inspectarea vizuală a graficelor rezultate. Ei au raportat găsirea unora dintre cele mai importante caracteristici care ar putea determina șansele de supraviețuire ale unui pacient, respectiv gena ESR1.

Având în vedere lucrările de cercetare revizuite, următoarele aspecte ar trebui să fie luate în considerare atunci când se face angajarea într-o problemă de predicție a cancerului. De multe ori, vrem să cunoaștem cazurile maligne și benigne, sau reapariția cancerului, deci, în general vorbind, de cele mai multe ori vom avea o clasificare binară. Acuratețea ar trebui utilizată ca măsurătoare numai dacă setul de date este echilibrat. În plus, acuratețea poate însemna două lucruri: pacientul este fie depistat ca având cancer, astfel încât un tratament să poată fi administrat cât mai curând posibil, fie pacientul este sănătos, ceea ce înseamnă că nu sunt necesare investigații suplimentare sau administrarea vreunui tratament. Dacă setul de date este dezechilibrat așa cum este de obicei în seturile de date de predicție a cancerului, ar trebui utilizat în schimb scorul F1. Cel mai adesea este încurajată reglarea algoritmului pentru a evita pe deplin rezultatele fals negative, deoarece, dacă nu se face acest lucru, poate duce la decesul pacientul. Pe de altă parte, rezultatele fals pozitive ar trebui reduse la minim pentru a reduce costurile investigațiilor și tratamentelor medicale. În cazul predicției cancerului, datele conțin adesea zgomot, din cauza înregistrărilor care au fost introduse eronat, astfel încât acestea trebuie să fie detectate și, uneori, chiar eliminate manual. În cele din urmă, algoritmii care nu pot fi ușor interpretați sau explicați unui medic ar putea fi respinși, astfel încât, dacă este posibil, ar trebui aleși algoritmi mai simpli.

2.4 Sarcina molară

Principala problemă abordată în această teză este din domeniul sarcinii molare, mai exact reapariția acestui tip de cancer (creșterea anormală a celulelor) la pacienții tratați, prognozarea recuperării pacientului și detectarea anomaliilor în perioada de vindecare a pacientului. Această secțiune va descrie sarcina molară în general, va prezenta studiile aferente care utilizează învățarea automată ca soluție și va descrie problemele particulare pentru care această teză caută soluții.

Boala trofoblastică gestațională (GTD) este un termen folosit pentru a descrie o serie de boli rare în care celulele trofoblastice anormale cresc în interiorul uterului după concepție [90, 5, 96]. Cel mai frecvent tip de GTD este sarcina molară, cunoscută și sub denumirea de aluniță

hidatiformă [69]. GTD face ca nivelul hormonului gonadotropină corionică umană (hCG) să fie foarte mare, deoarece celulele trofoblastice produc hCG. Mai multe măsurători ridicate de hCG sunt un indicator puternic al prezenței unei aluniță hidatiforme complete [17].

Mai multe studii de cercetare utilizează învățarea automată pentru detectarea și clasificarea aluniței hidatiforme. Autorii articolului [76] folosesc procesare și segmentare clasică de imagini cu îndrumarea patologilor experți pentru a analiza imagini cu aluniță hidatiforme. Mai târziu, aceiași autori folosesc o rețea multineuronală pentru a analiza imaginile și a recunoaște tipare fie ale aluniței hidatiforme parțiale, fie ale celor hidatiforme complete din [77]. Ei susțin că au depășit cu această metodă performanța multor experți umani.

Odată cu îndepărtarea aluniței hidatiforme, s-a demonstrat că nivelul hormonilor hCG scade exponențial la femeile diagnosticate cu GTD [81, 86, 97, 105]. În cele mai multe cazuri, nivelurile hCG vor reveni la normal, fără tratament suplimentar, dar s-a demonstrat că în Marea Britanie 15% dintre femeile care au avut aluniță hidatiformă completă, au necesitat chimioterapie [88].

După îndepărtarea aluniței hidatiforme, există riscuri ca un pacient să redezvolte GTD. Astfel, supravegherea continuă este obligatorie, deși riscurile de a dezvolta GTD sunt puțin mai mici, 5%, pentru pacienții care au avut anterior aluniță parțială hidatiformă, comparativ cu 20% până la 25% pentru cei care au avut aluniță hidatiformă completă [69]. Pe scurt, un pacient este monitorizat în general timp de un an, deși redezvoltarea GTD are loc de obicei în primele șase luni. Revenirea la niveluri nedetectabile de hCG poate dura până la 24 de săptămâni.

Cercetările existente privind GTD și cancerul molar sunt limitate. Deși este o formă rară de cancer, este una care afectează mulți pacienți. Literatura de specialitate se concentrează în principal pe detectarea tipurilor de aluniță din datele de imagine și foarte puține studii se preocupă cu faza post-tratament a bolii, ceea ce înseamnă că cercetările ulterioare sunt necesare și binevenite.

2.5 Concluzii

Acest capitol a acoperit treptat domeniul medical și cel al predicției cancerului, în care învățarea automată a fost utilizată ca răspuns la anumite probleme. S-au adunat cercetări cu privire la două tipuri de cancer: cancerul de sân și sarcina molară. Ambele fiind cancere specifice la femei, primul fiind unul dintre cele mai răspândite, cu o mulțime de cercetări în spate, iar al doilea fiind un tip de cancer mai puțin frecvent, mai puțin cercetat, revizuirea literaturii de specialitate s-a axat pe găsirea de informații generale care ar putea ajuta la rezolvarea problemelor referitoare la cancerul molar.

S-a constatat că, în general, algoritmii de învățare automată sunt aleși pe baza:

- tipului de date, adică structurate (tabelare) și nestructurate (texte, imagini, videoclipuri)

- tipului de problemă, adică, clasificare, regresie, grupare, prognoză sau chiar mecanisme defensive contra atacurilor adverse

Mai mult decât atât, cercetările au arătat că, deși pe baza tendinței, ar trebui să fie disponibile din ce în ce mai multe date pentru cercetători. Acest lucru nu este întotdeauna cazul, iar pentru cei care se ocupă de seturi de date mici, metode precum validarea încrucișată cu k pliuri sau îmbunătățirea datelor există.

Cercetările au arătat, de asemenea, că pentru datele istorice, cum ar fi în problema recurenței sarcinii molare, pot fi utilizate rețele neuronale recurente, deoarece în anumite cazuri pot depăși chiar și experții medicali la a face prognoze. Astfel, dovedindu-se a fi un aliat de nădejde al personalului medical, în prezicerea și prevenirea viitoarelor îmbolnăviri.

În sprijinul acestui capitol au fost publicate următoarele articole de cercetare:

- *Machine Learning in Healthcare: An Overview*, de Árpád Kerestély, Lucian Mircea Sasu și Marius-Sabin Tăbîrcă, în Buletinul Universității Transilvania din Brașov, 2018. Acest articol stă la baza secțiunii 2.2 și acoperă algoritmii de învățare automată utilizați recent în domeniul medical, datele disponibile în scopuri de cercetare și domeniile la care se extinde domeniul medical.
- *Feature Inspection and Elimination in the Context of Breast Cancer Prediction*, de Árpád Kerestély, în Proceedings of the 36th International Business Information Management Association (IBIMA), 2020. Această lucrare își propune să revizuiască unele dintre cercetările existente din domeniul cancerului de sân, fiind astfel baza secțiunii 2.3. Adună un set de metode și procese utile și, în cele din urmă, le aplică pe un set de date referitor la cancerul de sân disponibil public, cu accent pe înțelegerea modului în care diferitele trăsături sau absența lor pot influența rezultatul unei predicții.

Capitolul 3: Date, preprocesarea datelor și gestionarea trăsăturilor

Datele reprezintă o parte importantă a fiecărei cercetări de învățare automată [58]. Un cercetător o problemă în absența datelor, dar are mai multe probleme în prezența acestora. Este greu realizarea cercetărilor pe un set de date necunoscut sau pe unul care nu a fost pe deplin înțeles. Prin urmare, este important cunoașterea câtorva tehnici de preprocesare a datelor, care vor ajuta la analiza exploratorie a datelor, reducând astfel numărul de potențiale probleme viitoare. Acest capitol își propune să treacă în revistă analiza exploratorie, vizualizarea datelor, selecția și ingineria trăsăturilor, precum și unele probleme frecvente pe care un cercetător le-ar putea întâmpina, pe două seturi de date concrete: pe datele NKI privind cancerul de sân disponibile public și pe datele private privind sarcina molară.

3.1 Provocări în colectarea datelor clinice

Accesul la datele clinice, care face obiectul principal al fiecărei cercetări medicale, a fost redus considerabil de recentul Regulament general al UE privind protecția datelor (GDPR), care a intrat în vigoare la 25 mai 2018. Deși UE dispunea de un cadru juridic anterior care datează din 1995, acest nou regulament, păstrând în același timp abordarea generală de reglementare a predecesorului său, introduce, de asemenea, o serie de noi obligații de conformitate și, împreună cu aceasta, sancțiuni mai mari. Acest lucru încurajează instituțiile medicale să fie mai puțin deschise la partajarea de date clinice cu cercetătorii, ca să nu mai vorbim de faptul că îi restricționează în întregime de la publicarea datelor în spațiul public, astfel că este un de îngrijorare semnificativ pentru cei implicați în cercetarea clinică.

EU-GDPR prevede în mod clar că prelucrarea datelor genetice, biometrice sau de sănătate cu caracter personal (adică identificabile, ne anonimizate) este interzisă [30]. Cu toate acestea, articolul 89 din EU- GDPR prevede că prelucrarea este permisă în scopuri de arhivare în interes public, în scopuri de cercetare științifică sau istorică sau în scopuri statistice, dacă se iau garanții adecvate, cum ar fi minimizarea datelor, pseudonimizarea și anonimizarea, acolo unde este posibil. Această scutire permite statelor membre libertatea de a legifera la nivel național în anumite domenii, una dintre acestea fiind prelucrarea datelor cu caracter personal în scopuri științifice și de cercetare.

Cu aceste reglementări în vigoare, deși datele sunt generate în cantități enorme de fiecare instituție medicală, devine o resursă prețioasă pentru cercetători. Analizând imaginea de ansamblu, un cercetător poate avea acces fie la date publice anonimizate, eventual colectate și publicate înainte de GDPR, fie prin solicitarea unei aprobări etice, care este un proces lung și consumator de timp, care se termină adesea cu respingerea solicitării. Restul capitolului va prezenta un set de date pentru studiul cancerului de sân, care poate fi obținut din surse publice, și un set de date pentru studiul sarcinii molare, care a fost obținut de la o instituție medicală situată în Irlanda înainte de intrarea în vigoare a GDPR. Ambele seturi de date sunt relativ mici, consolidând deficitul și prețiozitatea datelor clinice. În cazul setului de date privind sarcina molară, s-au făcut inițiative pentru a obține acces la mai multe date, dar din cauza reglementărilor și a necesității unei aprobări etice, acestea au eșuat.

3.2 Cancerul de sân

Această secțiune își propune să experimenteze și să descopere cunoștințe interesante într-un set de date privind cancerul de sân disponibil public, utilizând cunoștințele colectate din capitolul 2. Aceste cunoștințe constau în principal în analiza exploratorie, vizualizarea datelor, gruparea, preprocesarea, selectarea trăsăturilor, partiționarea, clasificarea, evaluarea modelelor de învățare și validarea încrucișată.

Ghidat de lucrările studiate și de căutarea atentă pe diverse depozite publice de seturi de date [20, 35, 75, 93] s-a descoperit că există puține seturi de date disponibile public referitor la cancerul de sân. Dintre acestea, setul de date Wisconsin Diagnostic Breast Cancer (WDBC) este cel mai frecvent utilizat, oferind 32 de trăsături ale 569 de pacienți. Este un set de date destul de mic, publicat în 1995, care a făcut obiectul atâtor studii de cercetare, încât este greu aducerea de noi contribuții la acesta. Din păcate, nici unele dintre celelalte seturi de date nu sunt prea mari, iar unele au, de asemenea, un număr mare de valori lipsă, motiv pentru care au coborât pe lista de priorități.

Un anumit set de date, compilat de Netherlands Cancer Institute (NKI), deși conține doar 272 de înregistrări ale pacienților cu cancer de sân, are o cantitate impresionantă de 1570 de trăsături. Acest set de date este relativ nou în forma sa actuală, deși sunt indicii că este mai mult sau mai puțin același set de date care a fost studiat de [98]. Setul de date nu a fost pe deplin explorat de literatura de specialitate, deci este un candidat bun pentru a aduce noi contribuții. În plus, faptul că are un număr mare de caracteristici, îl face un candidat bun pentru toți algoritmii de preprocesarea a datelor și de selecție a trăsăturilor.

Datele privind cancerul de sân de la Netherlands Cancer Institute (NKI) au fost descărcate de la [84]. Acesta conține date demografice și clinice despre pacienți și date genetice din tumorile mamare, care însumează aproximativ 1570 de trăsături. Trăsătura "eventdeath" oferă informații despre cazurile de cancer care au fost fatale. Distribuția valorilor trăsăturii este dezechilibrată, 77 (28,3%) cazuri de "deceased" și 195 (71,7%) "survived" cazurilor (Figura 3.1 din teză). După cum se vede din capitolul 2, distribuția valorilor este un aspect important, deoarece va influența alegerea metricii de performanță mai târziu.

Ca punct de plecare, trăsăturile au fost analizate manual și sa dovedit că primele 16 reprezintă date clinice, în timp ce restul de 1554 reprezintă date genetice. Dintre aceste 16, trei trăsături au fost eliminate manual, și anume "Patient", "ID" și "barcode", care reprezentau în mod clar informații irelevante pentru un algoritm de învățare. Trăsături "eventdeath" a fost, de asemenea, extrasă ca variabilă dependentă. Trăsăturile setului de date au fost analizate numeric și s-a realizat o histogramă a celor 12 trăsături clinice rămase (a se vedea Figura 3.2 din teză). Nu s-au găsit valori lipsă.

Pe un set de date care conține restul de 13 trăsături de date clinice, s-a făcut o verificare pentru a vedea modul în care cele 12 variabile independente se corelează între ele și cu singura variabilă dependentă. Scorul de corelație Pearson a fost calculat între perechi de trăsături, iar rezultatele au fost codificate folosind culori pentru a putea vizualiza mai bine care două caracteristici se corelează cel mai mult, așa cum se poate observa înFigura 4 (Figura 3.3 din teză). S-a descoperit că "eventdeath" este puternic corelat cu trăsăturile "survival" și "timerecurrence", precum și că "survival" și "timerecurrence" se corelează puternic între ele.

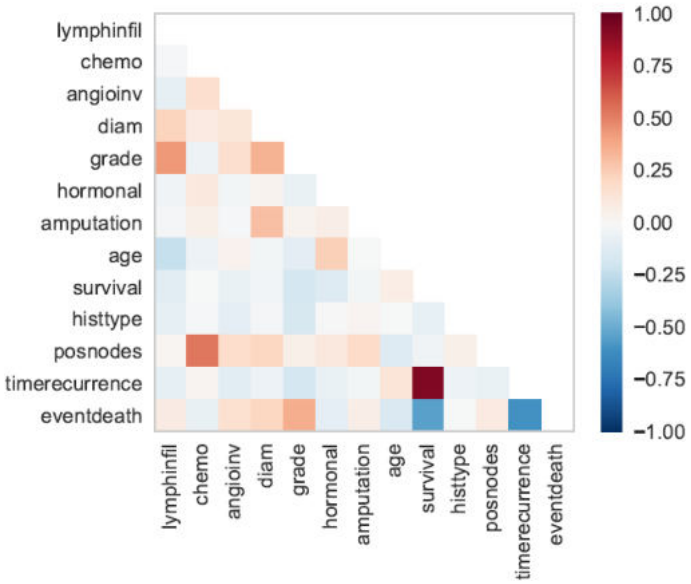


Figura 4: Corelația Pearson între 13 trăsături, inclusiv variabila dependentă.

Separabilitatea clasei a fost analizată folosind cele 12 variabile clinice independente, cu metoda de vizualizare RadViz. Rezultatele pot fi văzute în Figura 3.4 din teză, de la care s-ar putea concluziona, că, deși există o anumită separare între cele două clase, aceasta nu este una foarte clară.

Clasificările preliminare efectuate pe setul de date care conținea la început doar cele 12 trăsături clinice, apoi toate cele 1566 de trăsături au arătat că scorul de clasificare este puternic influențat de trăsăturile selectate, așa cum se vede în primele două rânduri din Tabelul 1 (Tabelul 3.1 din teză). După o revizuire atentă a rezultatelor, concluzia a fost că o selecție automată a trăsăturilor ar trebui să fie rulată pe setul de date. Rularea algoritmului Recursive Feature Elimination with Cross-Validation (RFECV) folosind clasificatorul de regresia logistică a arătat că cele mai bune scoruri pot fi obținute folosind doar 32 de trăsături atunci când se utilizează acuratețea ca indicator de performanță, și 71 caracteristici atunci când se utilizează scorul F1. Timpul necesar pentru antrenarea clasificatorului a fost, de asemenea, îmbunătățit. Trăsăturile selectate pot fi găsite în Tabelul 3.2 și 3.3 din teză. Evoluția scorurilor, eliminând trăsături una-câte-una, poate fi văzută înFigura 5 (Figura 3.7b din teză).

Tabelul 1: Rezumatul rulării clasificatorului de regresie logistică cu un număr diferit de trăsături.

	Accuracy (%)			F1 score		
	score	fit_time*	score_time*	score	fit_time*	score_time*
Clinical features (12)	88.20 ± 4.35	0.181 ± 0.036	0.008 ± 0.002	0.78 ± 0.10	0.183 ± 0.030	0.011 ± 0.002
All features (1566)	84.15 ± 7.46	1.424 ± 0.419	0.075 ± 0.030	0.68 ± 0.18	1.530 ± 0.457	0.070 ± 0.020
Best features (32/71)	95.60 ± 4.30	0.053 ± 0.017	0.010 ± 0.005	0.96 ± 0.03	0.057 ± 0.014	0.015 ± 0.009

(*) Tested on a laptop having an Intel i7-4720HQ CPU, with 12 GB RAM

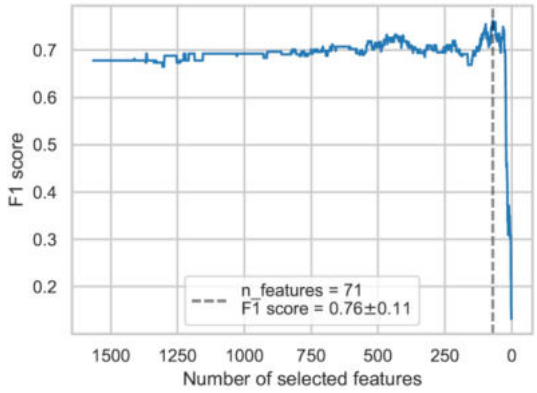


Figura 5: Socrul F1 al clasificatorului de regresie logistică în raport cu numărul de trăsături selectate.

Pentru a consolida utilitatea selecției automate de trăsături pe acest set de date, testele au fost repetate folosind un clasificator diferit, clasificatorul pădure aleatorie. După cum se poate vede în Figura 6 (Figura 3.8b din teză), evoluția scorurilor este și mai vizibilă în cazul clasificatorului pădure aleatorie. Comparația rezultatelor de dinaintea selecției de trăsături și după selecția de trăsături poate fi văzută în Tabelul 3.4 din teză, iar trăsăturile selectate pot fi văzute în Tabelul 3.5 din teză. De data aceasta, 23 de trăsături au fost selectate atunci când s-a utilizat acuratețea ca și indicator de performanță și 9 trăsături atunci când s-a utilizat scorul F1. Acuratețea medie s-a îmbunătățit de la 79,03% la 91,19%, în timp ce scorul F1 mediu de la 0,48 la 0,84.

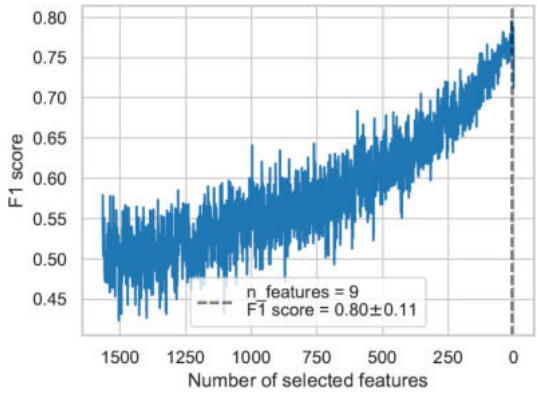


Figura 6: Socrul F1 al clasificatorului pădure aleatorie în raport cu numărul de trăsături selectate.

3.3 Sarcina molară

Aceasta secțiune acoperă setul de date folosit pentru principala problemă abordată de această teză, cea a sarcinii molare. Cunoștințele sunt extrase din trăsăturile care există în setul de date, precum și din relațiile dintre acestea. Ingineria trăsăturilor și tehnicile de gestionare a valorilor lipsă sunt aplicate pentru a pregăti setul de date pentru viitorii algoritmi de învățare. De-a lungul secțiunii, vizualizarea datelor este utilizată cât mai des posibil, pentru a ajuta la

perceperea în profunzime a setului de date, deoarece aceste cunoștințe sunt esențiale pentru înțelegerea algoritmilor prezentați în capitolul 4.

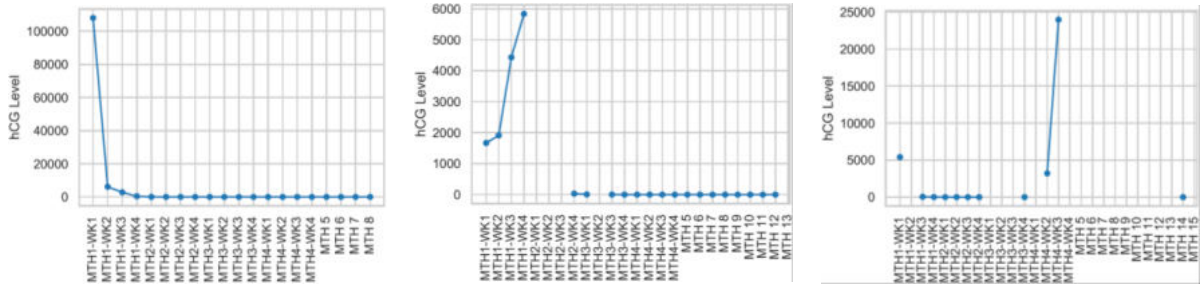
Setul de date studiat în această secțiune este unul privat, de la Cork University Maternity Hospital, unde profesorul John Coulter a colectat date între 2008 și 2013 despre pacienții care au avut sarcină molară. Setul de date conține măsurători post-tratament, într-un tabel. După tratament, pacienții care au avut sarcină molară sunt supuși unei perioade de supraveghere pe parcursul căreia nivelurile de hCG sunt măsurate periodic. Aceste măsurători sunt colectate în tabelul menționat mai sus, împreună cu alte informații cum ar fi vârsta pacienților sau tipul aluniței hidatiforme pe care pacienții le-au dezvoltat.

Setul de date are următoarele trăsături: MRN - numărul de înregistrare medical, I - identificator pacient, DX - diagnostic, care poate fi utilizat pentru a determina tipul aluniței, AGE - vârsta pacientului la momentul diagnosticului, DATA DDX - data la care a fost înregistrat diagnosticul. Următorul set de trăsături sunt măsurătorile hCG, o trăsătură pe săptămână pentru un total de 4 luni (4 săptămâni pe lună), apoi o trăsătură pe lună până la un total de 2 ani (24 luni). În total, 41 de trăsături, dintre care 5 reprezintă date clinice, în timp ce restul de 36 reprezintă măsurători hCG, dintre care 16 săptămânale și 20 lunare. Este interesant de observat că prelevarea de probe hCG din acest set de date se corelează puternic cu eșantionarea sugerată de literatura de specialitate cu privire la supravegherea pacienților post cancer molar. Cealaltă dimensiune a setului de date este de 57, ceea ce înseamnă că există date despre 57 de pacienți.

În urma analizei setului de date, se pot face mai multe observații:

- Din trăsătura DATA DDX, se poate stabili că acest set de date conține date despre pacienți din 2008 până în 2013, majoritatea pacienților fiind diagnosticați cu cancer molar în 2009 și 2010.
- Analizând statisticile de vârstă, se poate concluziona că sarcina molară poate apărea la orice vârstă, dar în acest set de date majoritatea pacienților au avut: 39, 31 sau 32 de ani atunci când au fost diagnosticați cu sarcină molară. A se vedea Figura 3.10 din teză pentru detalii.
- Din punct de vedere al tipului de aluniță hidatiformă, trăsătura DX, se poate spune că setul de date este în mare parte echilibrat. De asemenea, s-ar putea spune că șansele de a dezvolta fie un mol parțial (PM), fie un mol complet (CM) sunt aproape aceleași, 54,4%, respectiv 45,6%. A se vedea Figura 3.11 din teză pentru detalii.
- Folosind trăsătura de vârstă, după filtrarea pacienților în funcție de tipul molei pe care le-au dezvoltat, se poate concluziona că ambele tipuri de moli apar la orice vârstă. A se vedea Figura 3.12 din teză pentru detalii.
- Nivelurile hCG sunt eșantionate săptămânal sau lunar, pe care anumiți algoritmi de învățare nu le vor putea gestiona în forma lor actuală.

- Măsurătorile hCG au multe valori lipsă, probabil pentru că fie pacientul nu s-a prezentat la test, fie medicul a decis că pacientul nu trebuie testat.
- În medie, fiecare pacient are 13 măsurători, dar distribuția nu este normală, mai exact, 13 pacienți au mai puțin de 5 măsurători, în timp ce 4 pacienți au mai mult sau egal cu 27 de măsurători. Făcând o ruptură la mijloc, se poate spune că 24 de pacienți au mai mult de 15 măsurători, în timp ce restul de 33 de pacienți au mai puțin de 15. Imaginea de ansamblu poate fi observată în Figura 3.13 din teză.
- Toți pacienții au făcut primul test, în luna 1 - săptămâna 1. Apoi, se poate observa o scădere constantă în testarea pacienților, în Figura 3.14 din teză, până în lunile 5-6, care este probabil un test cheie pentru a determina dacă măsurătorile hCG descresc conform așteptărilor, adică la niveluri aproape nedetectabile, astfel încât mai mulți pacienți apar pentru test. Apoi, o altă scădere constantă a prezenței la test poate fi observată până în luna 17, după care foarte puțini pacienți sunt testați din nou până în luna 24, inclusiv.
- Figura 7 (Figura 3.15 din teză) arată câteva dintre măsurătorile hCG ale pacienților și, aruncând o privire mai atentă, se poate observa că boala se poate manifesta în moduri diferite pentru diferiți pacienți. În Figura 7a se poate observa o evoluție normală a bolii, care urmează o tendință de scădere exponențială. Apoi, în Figura 7b, măsurătorile sugerează că pacientul a avut complicații după extracția molei, dar după terapie, nivelurile de hCG au revenit la valorile așteptate în mod normal, descrescând exponențial. Al treilea caz, care poate fi văzut în Figura 7c, arată un exemplu în care boala a recidivat probabil după o perioadă mai lungă de timp, dar după intervenția medicilor, nivelurile au revenit la normal, după cum se poate observa din măsurătorile cu 10 luni mai târziu.



(a) Evoluția normală a bolii (b) Anomalie timpurie (c) Reapariția posibilă a bolii

Figura 7: Eșantioane de diferite tipuri de măsurători hCG.

După efectuarea unei analize exploratorie adecvate și aprofundate a setului de date, selecția manuală a trăsăturilor devine mult mai ușoară. Analizând din nou trăsăturile setului de date, dar de data aceasta având o înțelegere a ceea ce reprezintă de fapt, este ușor de determinat care dintre ele ar putea fi utile în viitor. Pentru început, numărul de înregistrare medical (MRN) și identificatorul pacientului (I) vor trebui cu siguranță eliminate, deoarece nu conțin informații

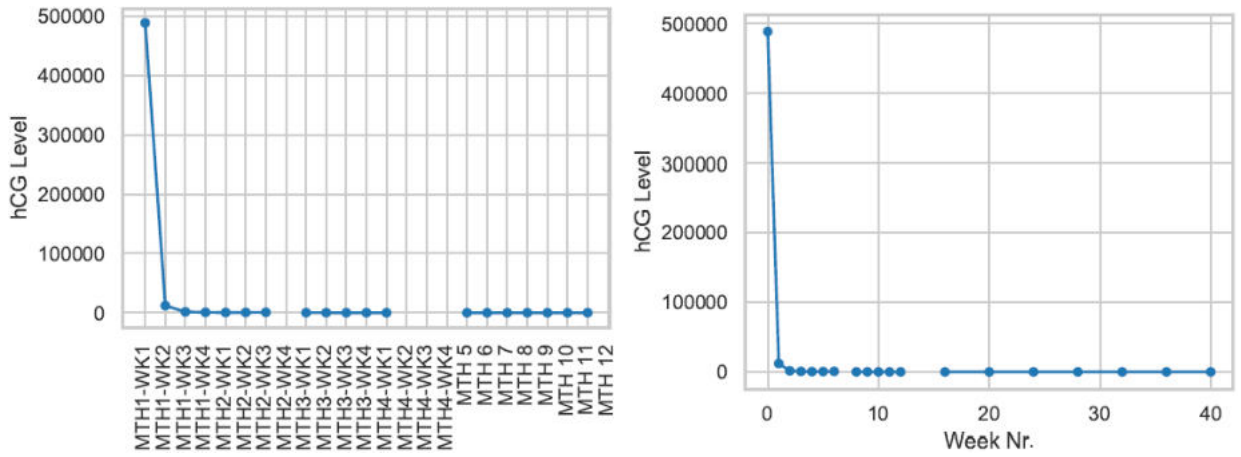
importante din perspectiva unui algoritm de învățare. Apoi, data diagnosticului, deși a ajutat la înțelegerea perioadei în care s-au făcut măsurătorile, nicio informație din literatura de specialitate nu sugerează că ar putea fi corelată cu dezvoltarea sarcinii molare, deci este în regulă să presupunem că este irelevantă, astfel putând fi eliminată din setul de date.

Vârsta și caracteristicile de diagnosticare (DX) necesită mai multă gândire înainte de a lua o decizie cu privire la soarta lor. Literatura de specialitate sugerează că vârsta unui pacient ar putea fi un factor important în dezvoltarea bolii, dar setul de date actual nu reflectă în întregime acest lucru. Având în vedere că un algoritm de învățare vede doar datele, este ușor să ne imaginăm că ar putea dezvolta ipoteze posibil greșite. Pe de altă parte, ar putea exista o posibilă corelație între vârsta pacientului și devoluția sau reapariția bolii. Decizia este mai mult sau mai puțin aceeași pentru trăsătura diagnostic. Astfel, alegerea păstrării ambelor sau a oricăreia dintre caracteristicile de vârstă și diagnostic (DX) va fi amânată până în momentul în care algoritmul de învățare va fi decis.

După cum se poate vedea în faza de analiză exploratorie, cea mai mare parte a setului de date o reprezintă măsurătorile hCG. Cu toate acestea, în starea lor actuală, ele sunt într-o oarecare măsură inutilizabile, deoarece ascund informații importante despre rata la care s-a făcut eșantionarea. Acesta este motivul pentru care caracteristicile trebuie să fie transformate astfel încât să reflecte intervalul de timp corect între măsurători, păstrând în același timp măsurătorile hCG actuale.

Transformarea menționată mai sus constă în reetichetarea și maparea fiecărei trăsături originale la un întreg care reprezintă numărul de săptămâni care au trecut de la începutul perioadei de supraveghere. Astfel, "MTH1-WK1" devine 0, ..., "MTH3-WK3" devine 10, și așa mai departe. Având numele trăsăturilor de măsurare ca numere permite reprezentarea corectă a acestora pe o axă a timpului sau calcularea diferențelor de timp dacă și când apare nevoia.

Figura 8 ilustrează nivelurile hCG ale unui pacient înainte și după transformare. Se poate observa cu ușurință în Figura 8a, distanța dintre două puncte la început, când eșantionarea a fost săptămânală, este aceeași ca între două puncte la sfârșitul perioadei în care eșantionarea a fost lunară. În Figura 8b, acest aspect este îmbunătățit și se poate observa o schimbare chiar și în formele curbelor. Înainte de transformare, curba a fost mai puțin abruptă decât după transformare.



(a) Înainte de transformare (b) După transformare

Figura 8: Nivelurile hCG ale unui pacient.

Conform celor menționate anterior, setul de date conține destul de multe valori lipsă, în special privind trăsăturile care reprezintă măsurătorile hCG. Secțiunea 3.3.4 din teză enumeră și discută unele dintre soluțiile posibile, dar o decizie este amânată până la capitolul 4, unde decizia va fi luată pe baza algoritmului de învățare ales. Printre metodele propuse pentru manipularea valorilor lipsă, există metode care ajută ca problema să rămână univarată, în timp ce alte metode sunt mai potrivite pentru algoritmi precum rețelele neuronale recurente.

În starea sa actuală, este dificil să se aplice orice tip de algoritmi de învățare automată pe setul de date, care ar putea extrage cunoștințe și ar putea face predicții privind nivelurile hCG viitoare, astfel încât setul de date trebuie să treacă printr-un alt set de transformări pentru a fi potrivit pentru algoritmii de învățare a secvențelor. Secțiunea 3.3.5 a tezei împreună cu Algoritmul 1 din teză arată modul în care setul de date este transformat astfel încât algoritmii din capitolul 4 să îl poată utiliza atât pentru învățarea, predicția și prognozarea secvențelor univariate, cât și pentru cele multivariate.

În sprijinul acestui capitol au fost publicate următoarele articole de cercetare:

- *Feature Inspection and Elimination in the Context of Breast Cancer Prediction*, de Árpád Kerestély, în Proceedings of the 36th International Business Information Management Association (IBIMA), 2020. Această lucrare își propune să revizuiască unele dintre cercetările existente din domeniul cancerului de sân, să adune un set de metode și procese utile și, în cele din urmă, să le aplice pe un set de date privind cancerul de sân disponibil public, cu accent pe înțelegerea modului în care diferitele trăsături sau absența lor pot influența rezultatul unei predicții, fiind astfel baza pentru secțiunea 3.2.
- *Vertically Shifted Exponential Best-Fit*, de Árpád Kerestély, Catherine Costigan și Marius-Sabin Tăbîrcă, în Proceedings of the 35th International Business Information Management Association (IBIMA), 2020. Această lucrare introduce o nouă metodă de potrivire a datelor la o curbă exponențial descrescătoare și reprezintă baza pentru

secțiunea 3.3 deoarece efectuează experimente pe setul de date privind sarcina molară.

Capitolul 4: Serii de timp și pronosticuri în predicția cancerului

Acest capitol își propune să introducă noi metode [61, 63] pentru prognozarea evoluției nivelurilor de hCG la pacienții diagnosticați cu sarcină molară, deoarece acest lucru ar putea reduce numărul de teste de sânge săptămânale de care un pacient ar avea nevoie pe parcursul perioadei de supraveghere post-operație. Nivelurile hCG sunt modelate ca o curbă exponențială deplasată vertical, iar acest capitol propune și demonstrează o soluție matematică pentru a găsi cei mai buni parametri pentru acest model, luând în considerare fiecare pacient în parte. În a doua parte a capitolului , RNR-urile vor fi utilizate pentru a modela comportamentul acestei devoluții exponențiale, folosind date de la un grup de pacienți. Metodele vor fi validate folosind date sintetice, precum și date reale.

4.1 Introducere în seriile de timp exponențial descrescătoare

Exemple de date în descreștere exponențială pot fi găsite peste tot în lume. În timp ce majoritatea acestora scad la zero, există și câteva exemple care scad cu o deplasare verticală. Datele de acest gen pot fi dificil de încadrat într-un model matematic. Algoritmi iterativi, cum ar fi Levenberg-Marquardt pentru potrivirea unei curbei neliniare cu metoda celor mai mici pătrate, există, dar au unele constrângeri care trebuie să fie luate în considerare, unii hiperparametri care trebuie să fie ajustați și pot avea convergență lentă.

Descreșterea exponențială este frecventă în științele naturii. Câteva exemple în acest sens sunt descreșterea în timp a unui pendul care se balansează în aer [51] și creșterea în timp a unei colonii de bacterii inițial mici [107]. Datele care se descresc exponențial în forma $y(t) = Ae^{-\alpha t}$ sunt foarte simple pentru a fi potrivite unui model prin simpla luare a logaritmului natural al fiecărui punct de date și prin utilizarea regresiei liniare simple pentru a adapta datele transformate la o linie. Când datele nu descresc la zero, nu este la fel de simplu. Datele în forma

$$y(t) = Ae^{-\alpha t} + B \tag{4.1}$$

apar de asemenea în mod natural. Date de această formă nu sunt la fel de ușor de potrivit la un model, mai ales dacă există doar un număr mic de valori de intrare.

Un set de date de acest tip sunt măsurătorile gonadotropinei corionice umane (hCG) la femeile cu boală trofoblastică gestațională (GTD) [90, 97]. S-a demonstrat în [103, 104, 41] că nivelurile de hCG la aceste femei scad exponențial în funcție de (4.1)(4.1).

Figura 9 (Figura 4.1 din teză) prezintă un eșantion de măsurători hCG. Se poate observa că măsurătorile hCG urmează o curbă exponențială de descreștere, că pot fi zgomotoase și că există intervale de timp în care nu există măsurători hCG, ceea ce crește semnificativ

dificultatea utilizării anumitor algoritmi de învățare. Dacă se dorește o prognoză din primele câteva măsurători, atunci va trebui luată în considerare o ușoară deplasare verticală.

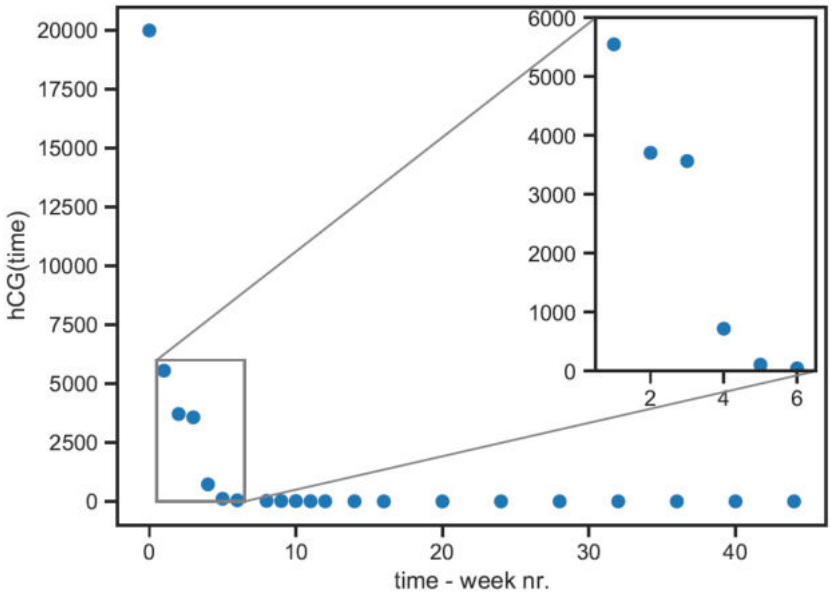


Figura 9: Un eșantion de măsurători hCG.

Problema modelării matematice a măsurătorilor hCG la femeile cu boală trofoblastică gestațională a fost realizată în [31] folosind transformări logaritmice. Mai târziu, în capitol, această metodă va fi denumită metoda "PseLogLin". Un rezumat al acestei metode poate fi găsit în secțiunea 4.1.1 al tezei și implementarea metodei în algoritmul 3 din teză.

O altă soluție pentru a potrivi un eșantion de măsurători hCG la o curbă este " The Levenberg-Marquardt method for nonlinear least squares curve-fitting problems" [44]. O sinteză a acestei metode poate fi găsită în teză la secțiunea 4.1.2, iar mai târziu în capitol, această metodă va fi denumită metoda "Iterative".

4.2 Metode de calcul directe bazate pe cunoștințele de domeniu

O nouă metodă va fi prezentată în această secțiune care este competitivă cu algoritmii menționați mai sus. Păstrând acuratețea lor, nu are hiperparametrii și singura constrângere pe care o are este că datele, care pot fi zgomotoase, trebuie să urmeze *aproximativ* o curbă de descreștere exponențială: $Ae^{-\alpha x} + B$ cu A, α și $B \geq 0$. Metoda a fost testată atât pe date simulate, cât și pe date reale și s-a comportat la fel de bine în ambele cazuri.

Definiția 1. *Având punctele* $(x_i, y_i)_{i=0}^{n-1}$ *și modelul* $f(x)$ *care a fost potrivit pe puncte, valorile reziduale (erorile) sunt:*

$$r_i = y_i - f(x_i) \tag{4.3}$$

pentru fiecare valoare al lui $i \in \{0, \dots, n-1\}$.

Se utilizează metoda celor mai mici pătrate, pentru a minimiza suma pătratelor reziduurilor pentru a găsi cei mai buni parametri care să potrivească modelul cu datele. Atunci când se

utilizează metoda celor mai mici pătrate pentru a potrivi un model, să zicem $f(\mathrm{x}; \mathrm{a}_1, \dots, \mathrm{a}_\mathrm{k})$, unde $\mathrm{a}_1, \mathrm{a}_2, \dots, \mathrm{a}_\mathrm{k}$ sunt parametrii ce trebuie estimați, la un set de date $\{\mathrm{x}_\mathrm{I}, \mathrm{y}_\mathrm{i}\}_{\mathrm{i}=0}^{\mathrm{n}-1}$, scopul este de a minimiza suma pătratelor reziduurilor

$$\phi(a_1, a_2, \dots, a_k) = \sum_{i=0}^{n-1} (f(x_i; a_1, \dots, a_k,) - y_i)^2 . \tag{4.4}$$

De obicei, atunci când se utilizează această metodă ϕ este minimizat prin rezolvarea sistemului de ecuații

$$\begin{aligned} \frac{\partial \phi}{\partial a_1}(a_1, \dots, a_k) &= 0 \\ \frac{\partial \phi}{\partial a_2}(a_1, \dots, a_k) &= 0 \\ &\vdots \\ \frac{\partial \phi}{\partial a_k}(a_1, \dots, a_k) &= 0 \end{aligned} \tag{2.5}$$

pentru $\mathrm{a}_1, \mathrm{a}_2, \dots, \mathrm{a}_\mathrm{k}$.

În cazul modelului (4.1), funcția ϕ este definită ca fiind

$$\phi(A, B, \alpha) = \sum_{i=0}^{n-1} (Ae^{-\alpha x_i} + B - y_i)^2 \tag{4.6}$$

Astfel

$$\begin{aligned} \frac{\partial \phi}{\partial A}(A, B, \alpha) &= \sum_{i=0}^{n-1} 2(Ae^{-\alpha x_i} + B - y_i)e^{-\alpha x_i} \\ &= 2A \sum_{i=0}^{n-1} e^{-2\alpha x_i} + 2B \sum_{i=0}^{n-1} e^{-\alpha x_i} - 2 \sum_{i=0}^{n-1} y_i\, e^{-\alpha x_i} \end{aligned} \tag{4.7}$$

$$\begin{aligned} \frac{\partial \phi}{\partial B}(A, B, \alpha) &= \sum_{i=0}^{n-1} 2(Ae^{-\alpha x_i} + B - y_i) \\ &= 2A \sum_{i=0}^{n-1} e^{-\alpha x_i} + 2B \sum_{i=0}^{n-1} 1 - 2 \sum_{i=0}^{n-1} y_i \end{aligned} \tag{4.8}$$

$$\begin{aligned} \frac{\partial \phi}{\partial \alpha}(A, B, \alpha) &= \sum_{i=0}^{n-1} -2(Ae^{-\alpha x_i} + B - y_i)Ax_i e^{-\alpha x_i} \\ &= -2A^2 \sum_{i=0}^{n-1} x_i\, e^{-2\alpha x_i} - 2AB \sum_{i=0}^{n-1} x_i\, e^{-\alpha x_i} + 2A \sum_{i=0}^{n-1} x_i\, y_i e^{-\alpha x_i} \end{aligned} \tag{4.9}$$

Următoarele funcții au fost definite f_k, g_k, h_k și l_k

$$\begin{aligned} f_k(\alpha) &= \sum_{i=0}^{n-1} e^{-k\alpha x_i} \\ g_k(\alpha) &= \sum_{i=0}^{n-1} y_i\, e^{-k\alpha x_i} \\ h_k(\alpha) &= \sum_{i=0}^{n-1} x_i\, e^{-k\alpha x_i} \\ l_k(\alpha) &= \sum_{i=0}^{n-1} x_i\, y_i e^{-k\alpha x_i} \end{aligned} \tag{4.10}$$

pentru a avea o formă mai simplă a ecuațiilor de mai sus

$$\frac{\partial \phi}{\partial A}(A,B,\alpha) = 2Af_2(\alpha) + 2Bf_1(\alpha) - 2g_1(\alpha) \tag{4.11}$$

$$\frac{\partial \phi}{\partial B}(A,B,\alpha) = 2Af_1(\alpha) + 2Bf_0(\alpha) - 2g_0(\alpha) \tag{4.12}$$

$$\frac{\partial \phi}{\partial \alpha}(A,B,\alpha) = -2A^2h_2(\alpha) - 2ABh_1(\alpha) + 2Al_1(\alpha). \tag{4.13}$$

Pentru a minimiza $\phi(A,B,\alpha)$, următoarele ecuații vor trebui rezolvate

$$\begin{aligned} \frac{\partial \phi}{\partial A}(A,B,\alpha) &= 0 \\ \frac{\partial \phi}{\partial B}(A,B,\alpha) &= 0 \\ \frac{\partial \phi}{\partial \alpha}(A,B,\alpha) &= 0. \end{aligned} \tag{4.14}$$

care după înlocuire arată așa

$$\begin{aligned} 2Af_2(\alpha) + 2Bf_1(\alpha) &= 2g_1(\alpha) \\ 2Af_1(\alpha) + 2Bf_0(\alpha) &= 2g_0(\alpha) \\ 2A^2h_2(\alpha) + 2ABh_1(\alpha) &= 2Al_1(\alpha). \end{aligned} \tag{4.15}$$

Utilizând regula lui Cramer pe primele două ecuații

$$\begin{aligned} \Delta &= \begin{vmatrix} f_2(\alpha) & f_1(\alpha) \\ f_1(\alpha) & f_0(\alpha) \end{vmatrix} = f_0(\alpha)f_2(\alpha) - f_1^2(\alpha), \\ \Delta_1 &= \begin{vmatrix} g_1(\alpha) & f_1(\alpha) \\ g_0(\alpha) & f_0(\alpha) \end{vmatrix} = g_1(\alpha)f_0(\alpha) - g_0(\alpha)f_1(\alpha), \\ \Delta_2 &= \begin{vmatrix} f_2(\alpha) & g_1(\alpha) \\ f_1(\alpha) & g_0(\alpha) \end{vmatrix} = g_0(\alpha)f_2(\alpha) - g_1(\alpha)f_1(\alpha), \end{aligned} \tag{4.16}$$

rezultă în

$$A = \frac{\Delta_1}{\Delta} = \frac{g_1(\alpha)f_0(\alpha) - g_0(\alpha)f_1(\alpha)}{f_0(\alpha)f_2(\alpha) - f_1^2(\alpha)} \tag{4.17}$$

$$B = \frac{\Delta_2}{\Delta} = \frac{g_0(\alpha)f_2(\alpha) - g_1(\alpha)f_1(\alpha)}{f_0(\alpha)f_2(\alpha) - f_1^2(\alpha)} \tag{4.18}$$

Valoarea pentru α poate fi aflată prin găsirea uneia dintre rădăcinile celei de-a treia ecuații

$$Ah_2(\alpha) + Bh_1(\alpha) = l_1(\alpha) \tag{4.19}$$

care după înlocuirea lui A si B, devine

$$\frac{g_1(\alpha)f_0(\alpha) - g_0(\alpha)f_1(\alpha)}{f_0(\alpha)f_2(\alpha) - f_1^2(\alpha)} h_2(\alpha) + \frac{g_0(\alpha)f_2(\alpha) - g_1(\alpha)f_1(\alpha)}{f_0(\alpha)f_2(\alpha) - f_1^2(\alpha)} h_1(\alpha) = l_1(\alpha). \tag{4.20}$$

Propoziția 1. *Având un set de date* $\{x_i, y_i\}_{i=0}^{n-1}$ *și un model*

$$y(t) = Ae^{-\alpha t} + B \tag{4.21}$$

cea mai bună valoare pentru α *poate fi determinată prin găsirea unei rădăcini a ecuației*

$$\frac{\mathrm{g_1(\alpha)f_0(\alpha) - g_0(\alpha)f_1(\alpha)}}{\mathrm{f_0(\alpha)f_2(\alpha) - f_1^2(\alpha)}} \mathrm{h_2(\alpha)} + \frac{\mathrm{g_0(\alpha)f_2(\alpha) - g_1(\alpha)f_1(\alpha)}}{\mathrm{f_0(\alpha)f_2(\alpha) - f_1^2(\alpha)}} \mathrm{h_1(\alpha)} = \mathrm{l_1(\alpha)}. \tag{4.22}$$

și apoi folosind această valoare al lui α *, cele mai bune valori ale lui* A *și* B *pot fi găsite folosind ecuațiile*

$$A = \frac{g_1(\alpha)f_0(\alpha) - g_0(\alpha)f_1(\alpha)}{f_0(\alpha)f_2(\alpha) - f_1^2(\alpha)} \tag{4.23}$$

$$B = \frac{g_0(\alpha)f_2(\alpha) - g_1(\alpha)f_1(\alpha)}{f_0(\alpha)f_2(\alpha) - f_1^2(\alpha)}. \tag{4.24}$$

Algoritmul pentru a găsi cei mai buni parametri pentru ecuația (4.1) este trivial și poate fi văzut în algoritmul (4.1)2 din teză.

Ecuația (4.22) nu are o singură soluție și cu calculul numeric nu este ușoară găsirea niciuneia dintre rădăcinile sale. (4.22)Astfel, sunt introduse următoarele modificări.

Ecuația (4.22) va fi rescrisă ca polinom(4.22), a se vedea secțiunea 4.2.5 din teză pentru demonstrații, făcând următoarea substituție

$$e^{-\alpha} = t. \tag{4.26}$$

Această substituție reduce funcțiile de la (4.10) o formă mai simplă(4.10)

$$\begin{aligned} f_k(t) &= \sum_{i=0}^{n-1} t^{kx_i} \\ g_k(t) &= \sum_{i=0}^{n-1} y_i\, t^{kx_i} \\ h_k(t) &= \sum_{i=0}^{n-1} x_i\, t^{kx_i} \\ l_k(t) &= \sum_{i=0}^{n-1} x_i\, y_i t^{kx_i}. \end{aligned} \tag{4.27}$$

și rezultă în modificarea ecuației (4.22) la următoarea formă(4.22)

$$\frac{g_1(t)f_0(t) - g_0(t)f_1(t)}{f_0(t)f_2(t) - f_1^2(t)}h_2(t) + \frac{g_0(t)f_2(t) - g_1(t)f_1(t)}{f_0(t)f_2(t) - f_1^2(t)}h_1(t) - l_1(t) = 0. \quad (4.28)$$

Una dintre proprietățile utile ale acestei ecuații este că are o rădăcină în intervalul $(0,1)$, din nou, a se vedea secțiunea 4.2.5 din teză pentru demonstrații, ceea ce face ca găsirea lui t să fie garantată și independentă de punctele de date, cu metoda bisecției (a se vedea secțiunea 4.2.3 din teză). Găsirea lui t ajută la găsirea lui α folosind transformarea inversă

$$\alpha = -ln(t) \quad (4.29)$$

Pe scurt, găsirea lui α folosind ecuația (4.22) și metoda bisecției este dificilă, (4.22)deoarece are rădăcini multiple și intervalul în care trebuie căutat cel puțin una dintre ele lipsește, dar folosind ecuația (4.28), t poate fi găsit în intervalul $(0,1)$, independent de punctele de date și independent de ceea ce ar trebui să fie valoarea lui α. În cele din urmă, folosind ecuația (4.29), cea mai bună valoare a lui α poate fi găsită din valoarea lui $(4.28)(4.29)$t. A se vedea algoritmul 4 din teză pentru detalii cu privire la implementare. Codul sursă complet pentru această secțiune este public accesibil pe GitHub la adresa: https://github.com/akerestely/nonlinearBestFit.

Pentru a testa metoda de estimare a valorilor lui A, B și α, date sintetice de forma $\{x_i, y_i\}_{i=0}^{n-1}$ au fost creat folosind algoritmul 5 prezentat în teză. Mai multe teste au fost realizate folosind datele sintetice generate astfel.

Pentru început, un model a fost potrivit peste date pentru a vedea dacă metoda de estimare poate găsi parametrii inițiali (A, B, α și noise având valorile inițiale $1000, 3, 1$ respectiv 0). Rezultatele pot fi văzute în Figura 10 (Figura 4.3 din teză). Pentru a valida acuratețea modelului găsit atât numeric cât și vizual, s-a folosit metoda: *rădăcină din eroarea medie pătratică* (RMSE).

Următorul pas a fost testarea robusteții metodei. A fost adăugat zgomot la datele generate și au fost testate diferite combinații de parametri. Figura 4.4 din teză sau din apendicele A.1 prezintă rezultatele și demonstrează că metoda este capabilă să funcționeze cu date care conțin zgomot.

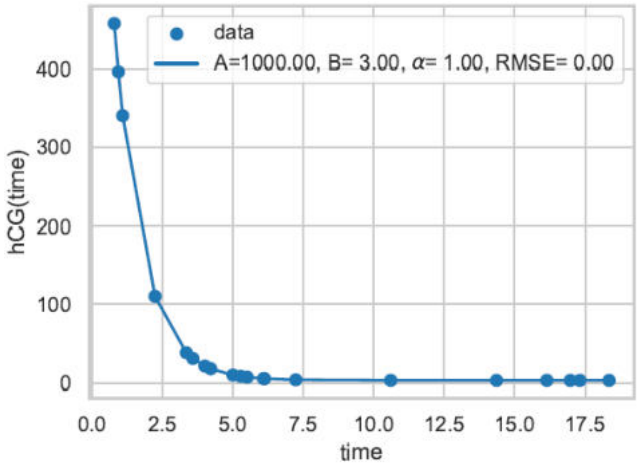


Figura 10: Curba generată de parametrii găsiți se potrivește perfect cu datele.

Pentru a percepe cu adevărat eficiența noii metode, s-a făcut o comparație cu alte două metode existente din literatura de specialitate, *curve_fit* din biblioteca SciPy din Python, care utilizează intern metoda Levenberg-Marquardt, denumită în continuare metoda "Iterative", și abordarea pseudo logaritmică prezentată în [31], denumită în continuare metoda "PseLogLin". Noua metodă este denumită în continuare "BestFit".**Error! Reference source not found.** (Figura 4.5 din teză) prezintă rezultatele rulării celor trei algoritmi pe diferite seturi de date generate. A se observa că rezultatele metodelor "Iterative" și "BestFit" se suprapun complet, indicat de asemenea și de valorile identice de RMSE, în timp ce "PseLogLin" are performanțe mai slabe în toate cele trei cazuri. Teza prezintă, de asemenea, și cazuri în care metoda "Iterative" nu converge, în timp ce "BestFit" și "PseLogLin" o fac. A se vedea Figura 4.6 din teză.

Observații

Metoda "BestFit" în comparație cu metoda "Iterative" nu are nevoie de hiperparametri pentru a funcționa corect, atât timp cât datele se află pe o curbă exponențial descrescătoare. Un al doilea avantaj al metodei "BestFit" este că, are o convergență mai rapidă decât metoda "Iterative", aproape în toate cazurile. Variația în timpii de convergență al metodei "BestFit" este dată numai de metoda de bisecție, care caută iterativ rădăcina ecuației (4.28).

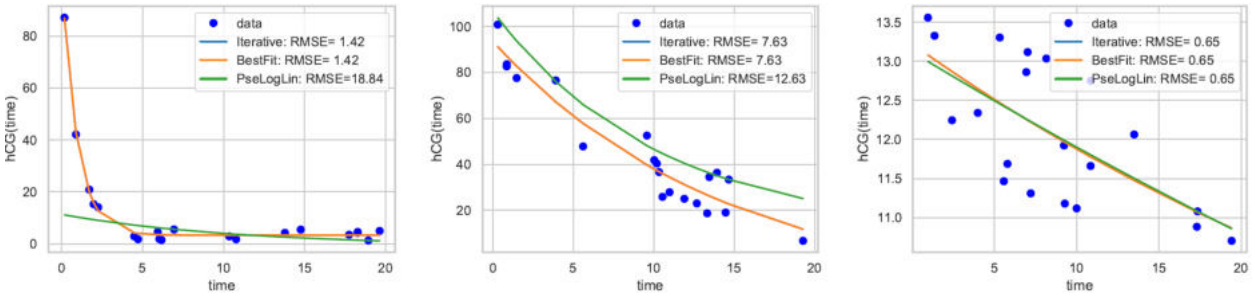


Figura 11: Comparația a trei algoritmi diferiți de potrivire a curbelor pe date sintetice.

Testarea noii metode pe datele reale privind sarcina molară prezentate în capitolul 3, a necesitat mai întâi eliminarea înregistrărilor pentru care numărul de măsurători hCG a fost mai mic de trei (a se vedea secțiunea 4.2.8 din teză pentru mai multe detalii), apoi o tratare a

valorilor hCG lipsă a fost necesară, prin eliminarea perechilor $\{x, y\}$ corespunzătoare din setul de date.

Primul experiment a fost compararea rezultatelor celor trei metode de potrivire a curbelor menționate mai sus. Figura 4.7 din teză arată constatările și consolidează faptul că metodele "BestFit" și "Iterative" sunt la fel de bune, în timp ce "PseLogLin" are performanțe mai slabe.

Apoi, au fost testate câteva aplicații interesante ale noii metode. Precum predicția cu acuratețe a liniei de tendință a măsurătorilor hCG, utilizând doar primele câteva valori, a se vedea Figura 12 (Figura 4.8 din teză).

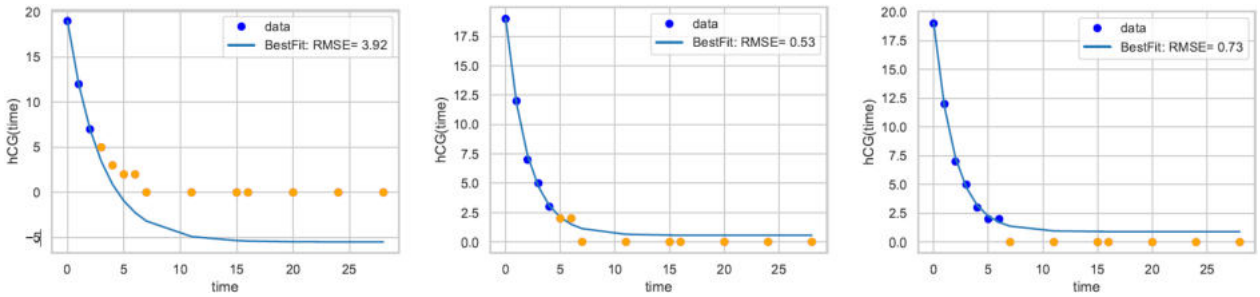
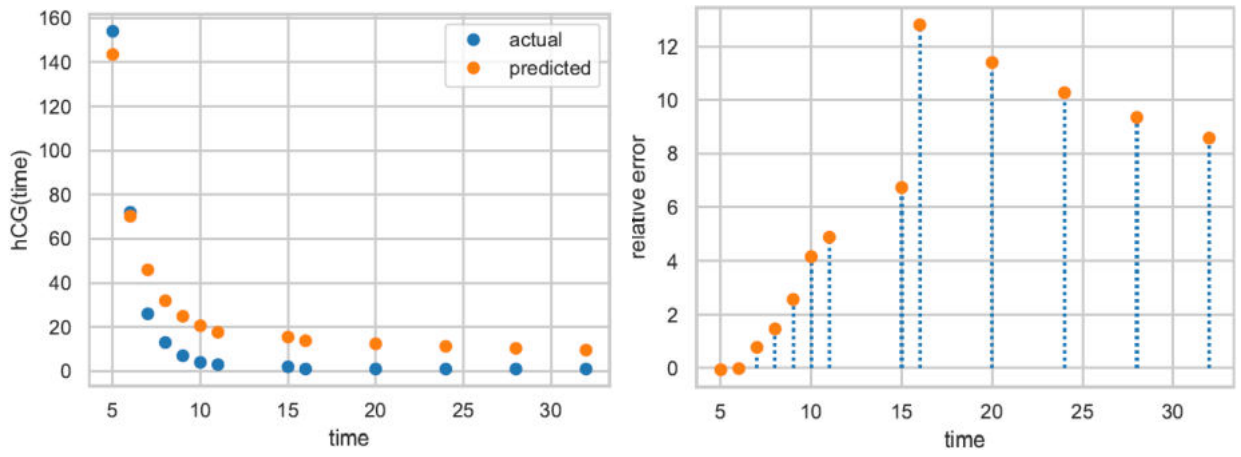


Figura 12: Potrivirea curbei pe același eșantion cu un număr diferit de puncte de date de pornire (puncte albastre).

Un alt caz de utilizare interesant, mai precis al algoritmului prezentat în această secțiune este unul, în care toate măsurătorile disponibile, până la un moment dat, sunt utilizate pentru potrivirea curbei, care apoi dă valoarea următoarei măsurători în secvență. Figura 13 (Figura 4.9 din teză) prezintă un caz în care a fost utilizată această abordare. Primele 5 măsurători (săptămânile 0-4) au fost utilizate pentru a potrivi curba inițială și pentru a prezice valoarea săptămânii 5. Apoi, primele 6 măsurători au fost folosite pentru a potrivi curba și pentru a prezice măsurarea pentru săptămâna 6 și așa mai departe. Figura arată, de asemenea și erorile relative pentru a reflecta mai bine acuratețea fiecărei predicții.



(a) Predicții (b) Erori relative

Figura 13: Următoarea măsurătoare prezisă din toate datele anterioare disponibile.

Modelul generat de algoritm poate fi folosit ca asistent IA pentru a valida noi măsurători, ajutând astfel la judecata unui expert medical. Având o prognoză, un medic poate spune cu ușurință dacă o nouă măsurătoare este în praguri acceptabile și poate decide dacă trebuie luate măsuri suplimentare. Modelul poate fi, de asemenea, actualizat cu noi date, făcându-l să genereze o nouă prognoză. Dacă în orice moment algoritmul nu reușește să conveargă, înseamnă că măsurătorile sunt prea zgomotoase sau că boala nu recidivează corespunzător. În ambele cazuri, ar trebui să fie un avertisment pentru medic. Figura 14 (Figura 4.10 din teză) arată clar aceste cazuri.

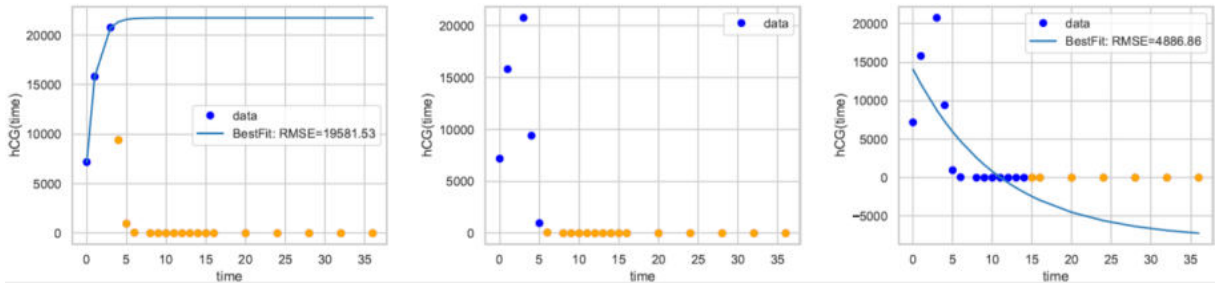


Figura 14: Trei comportamente a noii metode de potrivire a curbelor, în funcție de momentul în care a fost rulată predicția.

Atunci când se utilizează primele 3 puncte de date, algoritmul constată că măsurătorile cresc în loc să scadă. Medicul observă probabil același lucru și ia măsuri. Atunci când se utilizează primele 5 puncte de date, algoritmul nu poate determina o curbă descrescătoare exponențială, deoarece există o mare incertitudine din punctul de vedere al datelor. Cu toate acestea, un medic poate vedea că această incertitudine se datorează doar faptului că pacientul a început să se recupereze numai din săptămâna 3, caz în care medicul poate ajusta fereastra care este utilizată pentru calcularea curbei de tendință, făcând algoritmul să utilizeze punctele de date începând numai cu săptămâna 3, caz în care rezultatele încep să arate ca înFigura 15 (Figura 4.11 din teză). Dacă nu se face această modificare, atunci numai după a 13-a valoare începe algoritmul să învețe că măsurătorile scad exponențial, convergând încet la forma curbei finale.

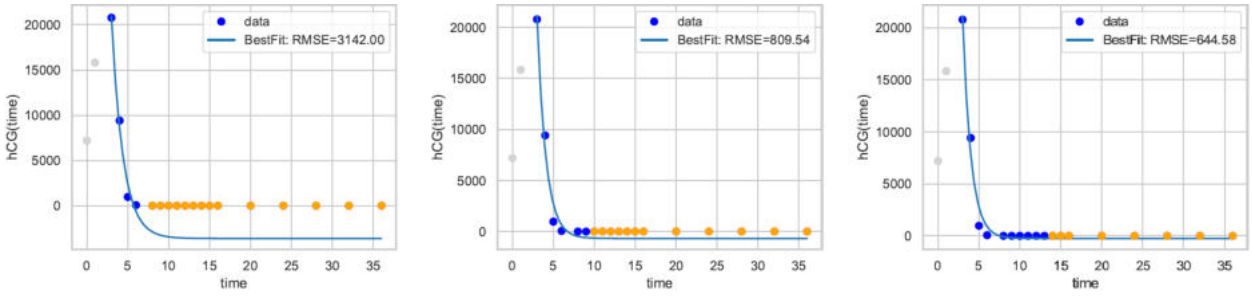


Figura 15: Sărind peste primele două puncte din acest eșantion (puncte gri), se ajută algoritmul să revină pe drumul cel bun și să funcționeze în general mai bine.

Pe scurt, această secțiune a prezentat o nouă metodă pentru a potrivi un model la date care urmează o curbă de descreștere exponențială în forma $Ae^{-\alpha x} + B$ cu A, α și $B \geq 0$. Datele de acest tip apar în măsurătorile hCG ale femeilor diagnosticate și tratate de sarcina molară.

Această nouă metodă utilizează calculul matematic pentru a determina cel mai bun model care se potrivește peste date. Această nouă metodă depășește alte metode existente, fie în ceea ce privește acuratețea, timpul de convergență sau ușurința de utilizare, prin faptul că nu are hiperparametri. De asemenea, în această secțiune au fost prezentate idei cu privire la modul în care metoda ar putea fi utilizată în scenarii din viața reală. Extrapolarea, prognozarea, interpolarea sau validarea punctelor de date sunt câteva dintre aceste cazuri de utilizare.

4.3 Rețele neuronale recurente

Metodele discutate în secțiunea anterioară sunt foarte bune la învățarea datelor care descresc cu o rată exponențială. S-ar putea spune că acestea sunt alegerea perfectă atunci când ne gândim la cazuri normale de sarcină molară, deoarece încorporează și folosesc cunoștințele de domeniu prin definiție. Dar ce se întâmplă atunci când se apar excepții și măsurătorile hCG nu mai scad așa cum era de așteptat, cum ar fi în cazurile în care reapare boala sau când nivelurile hCG ale pacientului au o creștere inițială timp de câteva săptămâni? Răspunsul este simplu, metodele de potrivire a curbelor nu mai sunt precise și în totalitate fiabile. Această secțiune își propune să studieze metode care nu au cunoștințe de domeniu încorporate, dar care învață de fapt rata cu care o secvență crește sau descrește analizând o serie de eșantioane.

Există destul de mulți algoritmi în literatura de specialitate care pot învăța din date secvențiale și pot face predicții pe baza a ceea ce a fost învățat. Cu toate acestea, setul de date disponibil pentru acest studiu este destul de mic, astfel, pe baza literaturii existente, RNR se arată ca fiind cele care ar putea învăța chiar și dintr-un număr redus de eșantioane.

Experimente au fost efectuate, în primul rând, pe o serie de date generate sintetic, care devin progresiv neregulate, evaluând astfel performanța RNR-urilor, câte o neregularitate pe rând. Au fost prezentate metode pentru a depăși unele dintre limitări. Apoi, cu aceste cunoștințe, s-au făcut încercări de învățare și predicție pe date reale privind sarcina molară.

Setul de date sintetic încearcă să reproducă date care seamănă foarte mult cu măsurătorile hCG din setul real privind sarcina molară. De la un caz ideal, în care datele scad exponențial, însemnând că pacienții își fac toate testele și se recuperează normal fără complicații sau recurențe, până la cazurile în care măsurătorile lipsesc sau conțin zgomot, la fel ca în datele din lumea reală. Setul de date este compus dintr-un număr de pacienți virtuali. Pentru fiecare pacient, măsurătorile hCG sunt generate folosind o versiune modificată a algoritmului 5 din teză, unde este garantat că valorile de timp sunt numere naturale pozitive, unice și în creștere. Fiecare pacient are măsurătorile sale hCG generate de o combinație diferită de paramterii A, B și α, unde $10 < A < 10^5$, $0 < B < 50$ și $0.3 < \alpha < 4$, a se vedea tabelul 4.1 din teză pentru mai multe detalii.

De asemenea, este necesar un pas suplimentar de preprocesare a datelor înainte de a încerca rularea RNR-urilor pe setul de date menționat mai sus. Mai exact, datele trebuie transformate în intrări și ieșiri. Pentru a realiza acest lucru, o fereastră imaginară va fi plasată peste eșantioanele setului de date, ceea ce va produce o pereche de intrare și ieșire pentru RNR. Apoi, fereastra va fi deplasată, oferind astfel o nouă pereche de intrare-ieșire. Intrările reprezintă un subset al trăsăturilor setului de date, iar ieșirile reprezintă trăsăturile care se doresc a fi prezise, în experimentele curente, măsurarea hCG. Dimensiunea acestei ferestre depinde de numărul de pași pe care RNR îi va procesa simultan. Dacă, de exemplu, pe baza măsurătorilor din primele patru săptămâni, se dorește prezicerea măsurătorii din a cincea săptămână, atunci fereastra va avea o dimensiune de cinci și va împărți datele astfel încât măsurătorile din primele patru săptămâni să reprezinte intrarea, iar a cincea măsurătoare să reprezinte ieșirea pentru RNR.

Arhitectura RNR-ului utilizat pentru experimente este următoarea:

- un strat de intrare cu cel puțin un neuron pentru măsurătorile hCG și, dacă este necesar, neuroni suplimentari pentru alte caracteristici, cum ar fi numărul de săptămâni
- un strat ascuns cu 50 de celule LSTM folosind funcția de activare ReLU, număr care a fost determinat empiric ca fiind atât suficient, cât și îndeajuns de mic, astfel încât parametrii rețelei să poată converge la valori optime chiar și în cazul câtorva eșantioane de intrare
- strat de ieșire complet conectat, cu un neuron pentru valoarea prezisă, care utilizează funcția de activare liniară

În plus, rețeaua a fost configurată să utilizeze funcția de optimizare Adam și să calculeze eroarea medie pătratică (MSE).

Mai multe experimente au fost efectuate pe seturile de date sintetice. Primul set de experimente a folosit trei măsurători consecutive hCG și a încercat prezicerea celei de a patra măsurătoare, pentru imita îndeaproape funcționalitatea metodei prezentate în secțiunea anterioară. În plus, aceste prime experimente au folosit doar măsurătorile hCG atât ca intrări, cât și ca ieșiri, axându-se practic pe cazul problemei univariate. Rezultatele rulării unei rețele timp de 1000 de epoci, pe toate eșantioanele, cu excepția uneia, care a fost folosit pentru testare, pot fi văzute în Figura 16 (Figura 4.13 din teză).

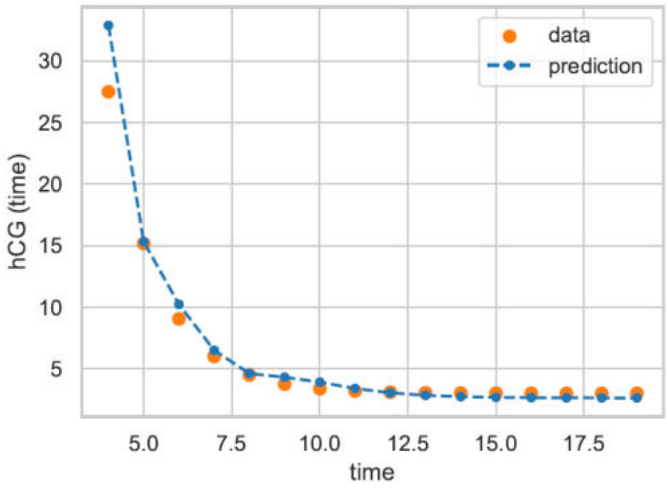


Figura 16: Predicția RNR-ului pe setul de date sintetic utilizând numai trăsătura hCG.

Performanța RNR-ului a fost evaluată și cu ajutorul validării încrucișate (CV), dar pentru că setul de date este mic, a fost utilizat un caz special al CV-ului, validarea încrucișată leave-one-out (LOO CV). Rezultatele pot fi văzute în Figura 4.14 din teză.

În continuare, pentru a îmbunătăți performanța modelului, trăsătura *week_nr* a fost adăugată la intrări, transformând problema într-una multivariată. Acuratețea modelului s-a îmbunătățit ușor în comparație cu scenariul precedent. Rezultatele pot fi văzute în Figura 4.15 din teză.

Apoi, modelul RNR-ul a fost testat pe date care conțin fie valori hCG cu zgomot, fie în care lipsesc valori hCG. În ambele cazuri, performanța s-a degradat, mai accentuat în cazul în care s-a folosit ca și intrare doar trăsătura hCG, dar modelul a reușit să facă totuși predicții acceptabile. Pentru mai multe detalii, se pot vedea Figurile 4.16 și 4.17 din teză. În încercarea de a ajuta RNR-ul să facă predicții mai bune, a fost introdusă o nouă trăsătură în locul trăsăturii *week_nr*, numită *delta_time*. Această nouă trăsătură a fost calculată din *week_nr* și reprezintă numărul de săptămâni care au trecut de la ultima măsurătoare. Figura 4.18 din teză arată că, folosind această inginerie de trăsături, predicția modelului a fost îmbunătățită în cazul unui set de date care conține valori lipsă.

RNR-urile instruite în modurile menționate mai sus, pot fi, de asemenea, utilizate pentru a prognoza mai mult de o singură măsurătoare. Pentru a evidenția mai bine acest comportament, setul de date generat a fost readus la versiunea care nu conținea zgomot și valori lipsă. Ca și anterior, rețeaua a fost antrenată pe toate eșantioanele, cu excepția uneia, având doar măsurătorile hCG ca intrare. În faza de testare, primele 3 măsurători hCG ale pacientului rămas au fost folosite inițial. Singura predicție rezultată în acest fel, a fost concatenată la cele 3 valori inițiale, și o nouă predicție a fost făcută cu ultimele 3 elemente ale secvenței rezultate. Această procedură poate fi repetată recursiv pentru un număr nelimitat de ori, rezultând în predicții chiar și pentru un viitor îndepărtat. În cazul în care se folosește pentru multe iterații această procedură, se pot acumula erori, făcând ca prognoza să se abată

de la un curs normal. **Error! Reference source not found.** (Figura 4.19 din teză) prezintă predicțiile făcute cu această procedură, precum și măsurătorile inițiale pentru comparație.

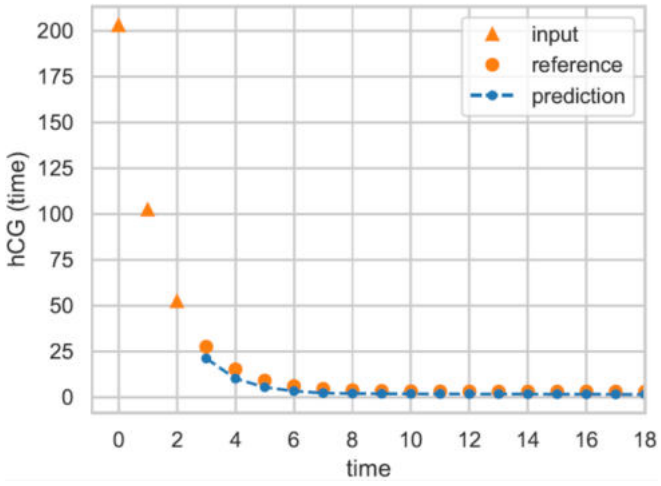


Figura 17: Prognoza RNN, prin reutilizarea predicțiilor.

Modelele secvență-la-secvență sau cele cu secvențe de intrare lungi sau arbitrar de lungi au fost omise din setul de experimente, deoarece nu sunt potrivite pentru contextul actual. Mai multe detalii pot fi văzute în secțiunea 4.3.1 a tezei.

În schimb, se propune o altă soluție interesantă, în care RNR-urile sunt antrenate pe secvențe de lungime fixă, dar în care sunt capabile să ofere predicții bune și din secvențe mai scurte sau mai lungi. Soluția implică returnarea tuturor stărilor ascunse din rețea și instruirea lor folosind variabilele țintă (de ieșire). Această soluție necesită două ajustări ale modului de lucru curent: modificarea ferestrei care generează perechile de intrare-ieșire și modificarea configurării RNR-ului. Având în aceste două ajustări, RNR-ul poate învăța nu numai din eroarea unei ieșiri, ci și din erorile acumulate ale tuturor ieșirilor, oferind astfel evident predicții mai bune chiar și din secvențe mai scurte. Trebuie remarcat faptul că, chiar dacă rețeaua returnează mai multe ieșiri, cea care oferă predicția reală este ultima, în timp ce restul sunt folosite numai pentru antrenare. Figura 18 (Figura 4.20a din teză) prezintă un scenariu, în care un model RNR a fost antrenat să prezică secvențe de lungime 7. La testare, același model a fost folosit pentru a prezice secvențe de lungime 1, 2, ..., 18. Se poate observa că, deși predicțiile nu sunt exacte atunci când lungimile secvenței sunt la extreme, ele se îmbunătățesc cu siguranță de îndată ce lungimile secvenței încep să ajungă în vecinătatea lungimii 7. Figura 4.20b din teză arată MSE-urile rezultate în urma antrenării și testării unui model RNR, de data aceasta raportând MSE-urile pentru fiecare pliu dintr-un LOO CV.

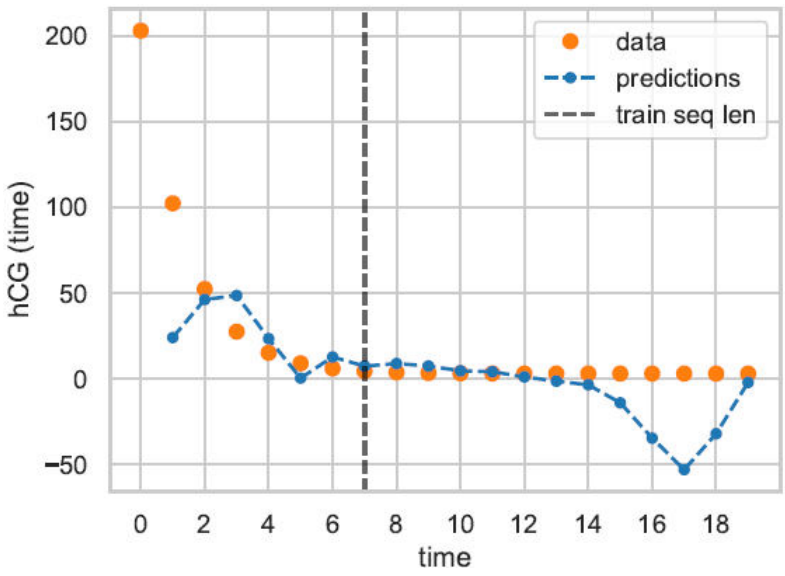


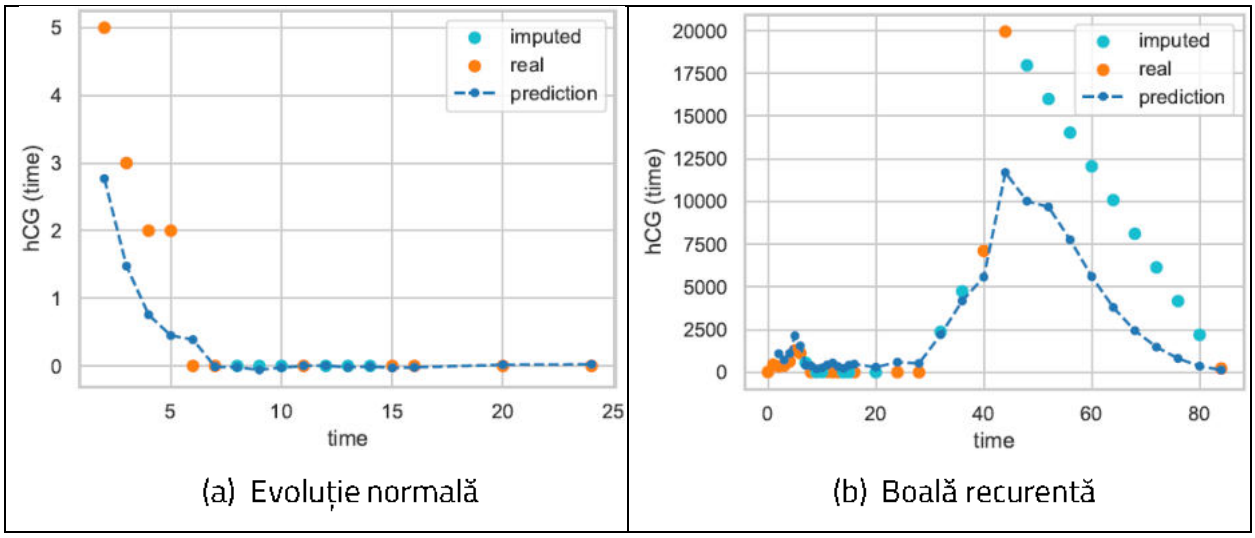
Figura 18: RNR antrenat pe secvențe de lungime fixă și testat pe secvențe de lungimi arbitrare.

În continuare, se va analiza performanța RNR-ului pe date reale, având în vedere că a avut rezultate bune pe datele sintetice, care fie au conținut zgomot, fie au avut valori lipsă, fie că au fost folosite secvențe de lungimi diferite pentru antrenare și testare.

După cum a fost descris din capitolul 3, setul real de date conține date de la pacienții diagnosticați cu sarcină molară. Cea mai mare parte a setului de date o reprezintă măsurătorile hCG, dar sunt disponibile și câteva înregistrări clinice despre pacienți. În cele mai multe cazuri, evoluția bolii a fost una normală, urmând astfel o curbă exponențială de descreștere, cu toate acestea, au existat și câteva cazuri în care vindecarea a avut loc după o întârziere de câteva săptămâni sau cazuri în care boala a reapărut. Setul de date inițial este transformat în mod similar ca și în cazul datelor sintetice, astfel încât toate măsurătorile hCG să fie într-o singură coloană, ceea ce permite, utilizarea procedurii de împărțire a datelor în perechi de intrare-ieșire folosind o fereastră, ca în secțiunea precedentă.

În plus, pentru a trata problema valorilor lipsă, din motivele prezentate detaliat în secțiunea 4.3.2 din teză, a fost folosită interpolarea liniară pentru a deduce valorile lipsă și a fost introdusă o nouă trăsătură booleană în intrări, numită *imputed*, care a fost setată la valoare de adevăr pentru fiecare măsurătoare hCG interpolată și pe valoare negativă în caz contrar.

Rezultatele de predicție după antrenarea unui RNR cu transformările menționate mai sus pot fi observate în Figura 19 (Figura 4.21 din teză). Rețeaua a fost antrenată să facă o predicție bazată pe 3 valori hCG anterioare. Figura prezintă un caz de evoluție normală a bolii și una în care se poate observa o recurență a bolii. În ambele cazuri, rețeaua a reușit să ofere predicții destul de corecte, demonstrând că RNR-urile sunt capabile să prezică evoluția bolii unui pacient individual pe baza a ceea ce s-a învățat din datele altor pacienți.



(a) Evoluție normală (b) Boală recurentă

Figura 19: Predicția RNR-ului pe date reale.

Pe scurt, rețelele neuronale recurente au arătat un potențial bun de învățare și predicție a secvențelor care descresc exponențial. Deși pot funcționa mai bine atunci când circumstanțele sunt mai ideale, s-a dovedit că fac predicții decente chiar și în cazul datelor ce conțin zgomot sau în cazul în care secvențele sunt incomplete. Diferitele scenarii în care pot fi aplicate RNR-urile, și anume predicția din secvențe de lungime fixă, predicția din secvențe de lungimi diferite sau chiar prognozarea prin introducerea predicțiilor în datele de intrare, le fac folositoare într-o varietate de situații. În general, performanța soluțiilor bazate pe RNR-urile prezentate ar putea fi, îmbunătățită având la îndemână unu set de date mai mare, cu mai multe exemple de niveluri ridicate de hCG sau boli recurente. Deși mai multe eșantioane din lumea reală ar fi cea mai bună opțiune pentru a crește setul de date, extinderea acesteia cu date sintetice ar putea fi, de asemenea, o opțiune bună, în special pentru cazurile subreprezentate. Având mai multe date ar deschide cu siguranță, de asemenea, calea spre explorarea altor algoritmi care lucrează cu serii de timp și date secvențiale.

În sprijinul acestui capitol au fost publicate următoarele articole de cercetare:

- Vertically Shifted Exponential Best-Fit, de Árpád Kerestély, Catherine Costigan și Marius-Sabin Tăbîrcă, în Proceedings of the 35th International Business Information Management Association (IBIMA), 2020. Articolul își propune să introducă o nouă metodă de a prognoza descreșterea nivelurilor de hCG, deoarece aceasta ar putea reduce numărul de teste de sânge săptămânale de care un pacient ar avea nevoie pe parcursul unui an de monitorizare, fiind astfel baza experimentelor efectuate în secțiunea privind noua metodă de determinare a curbei de devoluție în studiul sarcinii molare.
- Theoretical Study of Exponential Best-Fit: Modeling hCG for Gestational Trophoblastic Disease, de Árpád Kerestély, Catherine Costigan, Finbarr Holland și Marius-Sabin Tăbîrcă, în Proceedings of the 14th International Conference on Knowledge Science,

Engineering and Management (KSEM), 2021. Nivelurile hCG sunt modelate ca o curbă exponențială deplasată vertical, iar această lucrare propune și demonstrează o soluție matematică pentru găsirea celor mai buni parametri pentru acest model, fiind astfel baza demonstrațiilor din spatele noii metode propuse în secțiunea privind sarcina molară.

Capitolul 5: Calculul de înaltă performanță în contextul volumelor mari de date și al învățării automate

Calculul de înaltă performanță eficient pentru învățarea automată a devenit o necesitate în ultimii ani. Datele cresc exponențial în domenii precum sănătate, guvern, economie și odată cu dezvoltarea IoT, a smartphone-urilor și a gadgeturilor [59]. Acest volum mare de date are nevoie de un spațiu de stocare pe care nici un sistem de calcul tradițional nu îl poate oferi și trebuie să fie rulat pe algoritmi de învățare automată, astfel încât informațiile utile să poată fi extrase din acesta. Cu cât este mai mare setul de date, cu atât rezultatele vor fi mai precise de obicei, dar totodată și timpul de calcul va crește [60]. Astfel, s-a format nevoia de calcul eficient de înaltă performanță în sprijinul unor algoritmi de învățare automată. Acest capitol își propune să dezvăluie modul în care unul beneficiază de pe urma altuia, ce cercetări s-au realizat până acum și încotro se îndreaptă cercetarea.

5.1 Introducere

Calculul de înaltă performanță, volumele mari de date și învățarea automată au început ca subiecte diferite. Ele s-au dezvoltat analogi unul cu celălalt, conduși de necesități diferite. Cu toate acestea, s-au contopit la un moment dat și acum este greu să ne gândim la unul, fără ca celălalt să fie în fundal. Înainte de a intra în detalii, este important să fie analizate câteva concepte de bază cu privire la fiecare dintre ele.

"Calculul de înaltă performanță (HPC) se referă, în general, la practica de agregare a puterii de calcul într-un mod care oferă performanțe mult mai mari decât s-ar putea obține dintr-un computer desktop sau o stație de lucru tipică pentru a rezolva probleme mari în știință, inginerie sau afaceri." [100] Agregarea puterii de calcul se poate face pentru a avea ca rezultat un singur supercalculator sau un grup de calculatoare.

Cele mai frecvent utilizate sunt clusterele de calculatoare, deoarece un cluster poate fi extins cu ușurință cu noduri suplimentare (calculatoare) pentru a obține rapid mai multă putere de calcul sau spațiu de stocare. Nodurile sunt de obicei conectate și comunică prin conexiune Ethernet pentru a acționa ca un singur calculator.

Există o legătură strânsă între algoritmii de învățare automată și calculul de înaltă performanță [55], chiar dacă la prima vedere nu este atât de evidentă. Majoritatea algoritmilor de învățare automată trebuie să se "antreneze" înainte de a putea prezice (sau generaliza) rezultatele

intrărilor nevăzute. Antrenamentul necesită timp. Un fapt este dovedit: cu cât există mai multe date, cu atât sesiunea de antrenare va fi mai lentă. Un alt fapt este cert: cu cât sunt mai multe iterații de antrenament și și cu cât primește mai multe date un algoritm, cu atât are mai multe șanse să dea rezultate mai bune.

Volumele mari de date [66] se referă la o colecție de seturi mari de date care nu pot fi prelucrate utilizând instrumente tradiționale de administrare a bazelor de date. Acest lucru a generat numeroase provocări științifice legate de stocare, precum și în ceea ce privește prelucrarea și recuperarea datelor.

Literatura de specialitate conexă sugerează că ciclul de viață al procesului de analiză a volumelor mari de date constă în trei etape consecutive: achiziționarea de date, preprocesarea și stocarea datelor și analiza datelor. Acest capitol afirmă că există, de asemenea, unele variații ale acestei taxonomii. Cu toate acestea, o separare a ciclului general de viață al procesului de analiză a volumelor mari de date se face în cele trei etape de mai sus, având în vedere că taxonomia menționată poate capta cu precizie caracteristicile cheie ale analizei volumelor mari de date.

Rezultatele etapei de evaluare a performanței cercetării raportată în acest capitol, precum și contribuțiile relevante raportate în literatura de specialitate existentă demonstrează că Spark, o bibliotecă de calcul de înaltă performanță pentru rularea algoritmilor de învățare automată, prezintă un mare potențial de scalare. Astfel, decizia de proiectare de a lua în considerare Spark pentru nucleul de prelucrare a sistemului de analiză a datelor este complet justificată.

Cercetarea raportată în acest capitol se concentrează pe Spark și comportamentul său în contextul volumelor mari de date, al calculului de înaltă performanță și al învățării automate. În plus, este evaluat dacă Spark este un instrument bun pentru procesarea datelor și rularea algoritmilor de clasificare pe seturile mari de date. Mai mult decât atât, soluțiile sunt analizate cu privire la următorul set de întrebări:

- Cât este considerat ca fiind un volum mare de date?
- Este Spark suficient de bun pentru procesarea volumelor mari de date?
- Este Spark un instrument scalabil de rulare a algoritmilor de învățare automată?
- Pot fi rezolvate restricțiile de memorie utilizând Spark?
- Necesită o mentalitate diferită procesul de prelucrare a volumelor mari de date?
- De ce și când ar trebui să se utilizeze Spark pentru învățarea automată?
- Ar putea beneficia de puterea Spark seturile de date mai mici, sau acesta este potrivit doar pentru seturile mari de date?

5.2 Prezentare generală a literaturii

În era digitalizată în care trăim, cantități mari de date sunt generate de sistemul de sănătate [62], de sistemele guvernamentale și cele economice. Unele dintre sursele notabile de date

sunt "ținerea evidenței, conformitatea și datele referitoare la pacienți,..., datele din registrul național de sănătate, ..." [14]. Aceste date ajută prin furnizarea de "servicii centrate pe pacient, detectarea timpurie a răspândirii bolilor, monitorizarea calității spitalelor și îmbunătățirea metodelor de tratament" [14].

Autorii articolului [40] subliniază că Map-Reduce construit peste HDFS are unele dezavantaje, care sunt cruciale atunci când se lucrează cu algoritmi de învățare automată. Unul dintre dezavantajele majore este că arhitectura Map-Reduce este proiectată pentru a reîncărca date de pe disc la fiecare procedură de procesare Map-Reduce. Astfel, se bazează foarte mult pe viteza de citire/scriere a discului, care este de obicei foarte lentă în comparație cu viteza operațiilor în memorie. Algoritmii de învățare automată nu pot fi eficienți folosind Map-Reduce în forma sa nativă. "Spark" [11] depășește această limitare prin reducerea operațiilor de citire/scriere ale discului și prin oferirea unei soluții care rulează în memorie, păstrând în același timp comportamentul tolerant la erori al Map-Reduce-ului. Câștigul de viteză se pretinde a fi de 100 ori mai mare ca viteza Map-Reduce-ului.

Cercetarea recentă în domeniul învățării automate s-a concentrat foarte mult pe două tipuri de algoritmi: Rețele neuronale convoluționale (CNN) (cu toate variantele lor) și rețele neuronale profunde (DNN). Din cauza procesării lente și a amprentei mari de memorie, acești algoritmi nu pot beneficia de Spark, deoarece o mare parte din timp ar fi pierdut pe schimbul de informații între nodurile de calcul. Pentru a obține o viteză de comunicare mai rapidă între memorie și unitatea de calcul, acești algoritmi au fost mutați în unitatea de procesare grafică (GPU). Rezultatul a fost o creștere a vitezei de la 10 la aproape 60 la sută, în comparație cu o rulare pe CPU [24].

Calculul de înaltă performanță și învățarea automată coexistă, fapt susținut de numărul mare de lucrări care sunt disponibile pe această temă. Mai important este faptul că învățarea automată este ajutată de calculul de înaltă performanță, atingând astfel noi bariere.

5.3 Analiza comparativă dintre Spark și scikit-learn

Această secțiune prezintă un sistem integrat de analiză a datelor, care ia în considerare un model de evaluare care are ca scop optimizarea proceselor de analiză a volumelor mari de date. Deși datele reale folosite nu sunt din domeniul sănătății, ci din cel al industriei auto, rezultatele sunt valabile pentru orice problemă de clasificare care utilizează date tabelare, cum ar fi cea prezentată în capitolul 3 pentru cancerul de sân. Motivația din spatele acestei schimbări de date este că datele medicale sunt greu de obținut în cantități mari, mai ales de când a intrat în vigoare GDPR, și în timp ce obținerea de date de la instituții medicale ar fi posibilă, aprobările necesită mult efort și timp, ceea ce nu a fost disponibil pentru această cercetare.

Schimbarea datelor legate de cancerul de sân din capitolul 3 cu un set de date privind rulmenții (descrise în detaliu mai târziu în această secțiune), a fost posibilă deoarece acestea au multe

trăsături comune. În primul rând, ambele sunt date tabelare, care au ca și coloane trăsăturile datelor, iar ca rânduri, eșantioanele, adică măsurătorile entităților. Trăsăturile seturilor de date sunt reprezentate cu valori numerice în ambele cazuri. În cele din urmă, ambele tratează probleme de clasificare. Există și dezavantaje în ceea ce privește această abordare, și anume că numărul de caracteristici nu se potrivește și că setul de date cu rulmenți nu conține serii de timp, deci nu poate fi un înlocuitor pentru setul de date care tratează sarcina molară.

Evaluarea experimentală raportată în această lucrare ia în considerare un set de date privind detectarea defecțiunilor la rulmenți. Vibrațiile și semnalele acustice au fost măsurate pe un motor electric montat pe rulmenți. Rulmenții au avut patru condiții diferite de defecțiune: sănătoasă, cursă interioară și exterioară, și defect al mingii. Condiția rulmentului marchează clasa fiecărei intrări, permițând utilizarea învățării supravegheate. În ceea ce privește procesul de clasificare, problema poate fi considerată o problemă de clasificare pe mai multe clase, având ca și clase cele patru condiții de defecțiune ale unui rulment. Setul de date a fost generat cu ajutorul unui simulator de defecțiune al mașinilor de la SpectraQuest.

Setul de date pentru detectare a defecțiunilor la rulmenți a fost inițial împărțit în 336 de fișiere MATLAB, cu o dimensiune totală de 19,69 GB. Având în vedere scopurile stocării eficiente și prelucrării mai ușoare a datelor, cele 336 de fișiere MATLAB au fost convertite în același număr de fișiere Apache Parquet, însumând 9,75 GB. Deși nu există un acord general cu privire la pragul pentru ca un set de date să fie clasificat ca fiind mare, prin sugestiile de la [38], acest set de date se încadrează în categoria medie, deoarece dimensiunea sa aparține intervalului de 10 GB-1 TB și poate fi păstrat pe unitatea de stocare a unei mașini, mai degrabă decât în memoria sa.

Setul de date conține 14 coloane float64 și o coloană int32 (trăsături), și aproximativ 262 de milioane de rânduri (intrări / eșantioane). Trăsăturile sunt următoarele: BL_[X, Y, Z] (Rulment stânga - axa X, Y și Z), BR_[X, Y, Z] (Rulment dreapta - axa X, Y și Z), MR_[X, Y, Z] (Motor - axa X, Y și Z), [BL, BR]_AE (emisie acustică la rulmentul stâng și rulmentul drept), [BL, BR]_Mic (microfon rulment stâng și rulment drept), defect_type. Ultima coloană reprezintă clasa și poate avea patru valori diferite: Healthy, Ball, Inner și Outer. Acestea sunt stocate ca numere întregi, care iau valorile 0, 1, 2, respectiv 3. Trăsătura de viteză este, de asemenea, interesantă, deoarece reprezintă rotația motoarelor pe minut (rpm) și are valorile grupate în apropierea valorilor 300, 420, 540, ..., 2580, 2700.

În încercarea de a depăși limitările că datele nu încap în memorie sau că calculul rezultatelor durează prea mult, au fost generate patru seturi de date mai mici. În primul rând, s-au selectat 10^2 de rânduri, prin o eșantionare aleatorie, din fiecare dintre cele 336 de fișiere, fiecare dintre ele conținând inițial 780800 de linii, rezultând astfel într-un set de date necomprimat cu dimensiunea de 3,71 MB, care reprezintă aproximativ 0,01% din întregul set de date. Numele acestui set de date va fi în continuare, *data100*. Mai departe, 10^3, 10^4 și 10^5 linii au fost extrase

în același mod, producând trei seturi de date cu dimensiunile de 37,17 MB (≈ 0,12%), 371,7 MB (≈ 1,28%), și 3,62 GB (≈ 12,8%). Aceste seturi de date vor fi denumite în continuare, *data1k*, *data10k* și *data100k*. Aceste subseturi ale setului de date inițial permit evaluarea rapidă a unor configurări inițiale, dar permit, de asemenea, compararea progresivă a rezultatelor celor două biblioteci pe seturi de date mai mari.

Scikit-learn este o bibliotecă care oferă un API ușor de folosit pentru a rula modele standard de învățare automată, cum ar fi regresia logistică, mașini de suport vectoriale și așa mai departe. Este de obicei utilizat pentru a rula algoritmi de învățare automată pe probleme de clasificare, regresie și grupare pe seturi de date care încap în memoria sistemului.

Având în vedere algoritmii care sunt evaluați de autorii [16] pe setul de date cu rulmenți, rețelele neuronale artificiale, cunoscute și sub numele de perceptroni multistrat, au fost alese pentru a fi testate pe scenariile din această secțiune.

Înainte de a sublinia rezultatele testelor din studiul actual, există câteva aspecte de luat în considerare cu privire la modul în care a fost utilizată versiunea scikit-learn a rețelelor neuronale artificiale (MLPClassifier). Rețeaua considerată este compusă dintr-un strat de intrare de 14 neuroni, trei straturi ascunse cu 50, 100, respectiv 50 de neuroni și un strat de ieșire cu patru neuroni. Scikit-learn a necesitat doar configurarea stratului ascuns, deoarece straturile de intrare și ieșire au fost deduse din setul de date de instruire. Numărul maxim de iterații a fost setat empiric la 500. Funcția de activare a fost ReLu, s-a folosit optimizatorul Adam, termenul de regularizare L2 a avut valoarea de 1e-4, rata de învățare a fost constantă pe tot parcursul antrenamentului și a avut o valoare de 1e-3.

Este relevant de reținut că algoritmii scikit-learn ar putea profita de puterea de calcul paralelă a mașinilor multi-nucleu. Unii dintre algoritmii scikit-learn sunt în mod natural liniari. Prin urmare, acestea nu pot fi rulate în paralel, dar în cazul MLPClassifier, se poate vorbi despre paralelizare, deoarece aceasta beneficiază de implementarea BLAS optimizată, care asigură apariția apelurilor pe fire de execuție pentru diferite rutine de algebră liniară, cum ar fi multiplicările matricelor. În Figura 5.1 din teză, se poate observa că există 4 rulări pe 4 seturi de date *data100* diferite. Precizia mediană este de 48%, în timp ce acuratețea maximă este de aproximativ 53%. În plus, timpul mediu de rulare este de 1 minut și 33 de secunde, dar este aproape întotdeauna direct proporțional cu valorile acurateței. Ultimul rezultat este oarecum diferit în raport cu celelalte rezultate. Este, probabil, o consecință a împărțirii setului de date în seturi de antrenare și testare. În Figura 20, două dintre cele mai bune măsurători de timp și acurateță pot fi observate la scară logaritmică, pentru seturile de date *data100*, *data1k* și *data10k*. Acuratețea și timpul de rulare cresc pentru seturile de date mai mari, ceea ce este oarecum evident. Având în vedere un punct de vedere empiric, precizia crește logaritmic, în timp ce timpul de rulare crește exponențial. Figura arată măsurătorile reale ca puncte. Din cauza constrângerilor de memorie, măsurătorile pentru *data100k* nu au putut fi efectuate, dar

valorile prognozate, marcate cu un "x", iar linia de tendință, marcată cu o linie întreruptă, sugerează că acuratețea ar fi putut ajunge la o valoare de 71%, în timp ce timpul de rulare ar fi putut varia între 24 și 27 de ore.

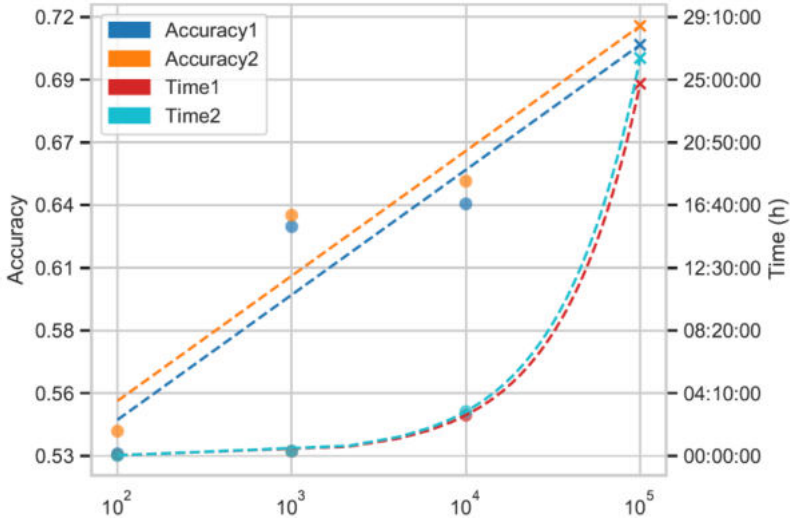


Figura 20: Evaluări și predicții privind acuratețea și timpul de rulare pe diferite seturi de date.

Experimentele scikit-learn oferă o bază solidă de comparație pentru experimentele cu Spark, deși au scos la suprafață și o limitare notabilă sub forma constrângerilor de memorie. În consecință, această abordare nu este scalabilă. Având în vedere un proiect care necesită scalare, se poate afirma că o soluție de tip cloud ar putea fi o alegere mai bună. Prin urmare, trecerea la Spark este o direcție naturală de cercetare.

Apache Spark este o bibliotecă de calcul distribuit cu codul sursă disponibil publicului, care este cunoscut mai ales pentru capacitățile sale de a rula analize pe date de volum mare. În plus, în cursul dezvoltării sale, a fost augmentat cu capacitatea de a rula algoritmi de învățare automată.

Înainte de a sublinia rezultatele testelor din studiul actual, există câteva aspecte de luat în considerare cu privire la modul în care a fost utilizată versiunea Spark a rețelelor neuronale artificiale (MultilayerPerceptronClassifier). Rețeaua considerată are aceeași arhitectură ca cea de la testele scikit-learn. Spark necesită specificarea și a numărului de neuroni din stratul de intrare și ieșire, pe lângă specificarea stratului ascuns. Numărul de neuroni de intrare este egal cu numărul de trăsături din setul de antrenare minus unu, deoarece, evident, coloana defect_type nu este utilizată ca intrare. Numărul de neuroni de ieșire este patru, deoarece există patru tipuri posibile de defecte. Numărul maxim de iterații este din nou 500, la fel ca în cazul testelor scikit-learn. Optimizatorul a fost l-bfgs, rata de învățare, numită stepSize, a fost constantă pe tot parcursul antrenării și a avut o valoare de 0,03, iar în cele din urmă, valoarea toleranței a fost de 1e-6.

În restul secțiunii, mai multe scenarii sunt descrise, analizate, discutate și comparate cu baza scikit-learn din perspectiva performanței.

Scenariul 1

Spark a fost configurat să ruleze un singur executor, cu un singur nucleu și cu 30 GB de memorie asociată. Intrarea a fost un fișier de tip *parquet* pentru fiecare dintre seturile de date *data100*, *data1k*, *data10k* și *data100k*. Când se importă date dintr-un fișier de tip parquet într-un DataFrame de Spark, nu sunt încărcate toate datele în memorie, ci doar subseturile care sunt necesare având în vedere un model la cerere.

Acest scenariu a încercat să reproducă configurația scikit-learn cât mai fidel posibil, având la bază presupunerea că algoritmii scikit-learn rulează pe un singur fir de execuție. Luând în considerare cele două eșantioane selectate, așa cum sunt prezentate în tabelul 5.1 din teză, se poate observa că există o mare discrepanță în ceea ce privește timpul de execuție între acest scenariu și cel care ține de implementarea scikit-learn, ceea ce a condus la unele investigații. S-a dovedit că implementarea rețelelor din scikit-learn utilizează fire de execuție pentru cea mai mare parte a calculului său, astfel încât, pentru a putea într-adevăr alinia cele două biblioteci, în următorul scenariu, s-au alocat mai multe nuclee executorului Spark.

Scenariul 2

Spark a fost configurat să ruleze cu o configurație similară ca în scenariul 1, cu singura diferență că în loc de un nucleu, executorului i s-a permis să utilizeze toate cele 12 nuclee disponibile pe mașină.

Așteptarea de la acest scenariu a fost, ca și în scenariul anterior, de a obține rezultate similare cu experimentele realizate cu scikit-learn. Cu toate acestea, rezultatele din tabelul 5.2 din teză sugerează un comportament neașteptat, mai exact, că timpii de execuție sunt aproape aceleași ca în cazul scenariului 1. În plus, la analiza utilizării procesorului în timpul testelor efectuate confirmă faptul că doar un singur nucleu a funcționat în mod activ. În consecință, eforturi suplimentare de cercetare și investigare au definit cel de-al treilea scenariu, care a implicat găsirea unei soluții care să utilizeze în mod eficient toate nucleele disponibile.

Scenariul 3

Spark a fost configurat pentru a rula cu aceeași configurație ca și în scenariul 2, dar în loc de un singur fișier de tip parquet, au fost utilizate mai multe fișiere parquet ca și intrare. Această modificare a făcut posibilă rularea Spark-ului pe toate nucleele disponibile. Deși la prima vedere poate părea neașteptat acest comportament, s-a dovedit mai târziu din experimente că Spark nu poate executa sarcinile în paralel pe un singur fișier de tip parquet monolitic. Autorul lucrării [36] specifică, de asemenea, că acest comportament se întâmplă numai atunci când datele sunt pe sistemul local și nu pe un HDFS. Pentru a găsi numărul de fișiere parquet

care ar rula optim cu Spark în configurația actuală, s-au realizat experimente privind divizarea datelor, în următoarele moduri:

- pe baza trăsăturii defect_type, ceea ce a dus la 4 fișiere, care, la rândul lor, au permis rularea în paralel a 4 sarcini, în timp ce nucleele disponibile erau 12
- prin gruparea vitezei în valori discrete și împărțirea în funcție de viteză, cu dezavantajul de a pierde informații (cu privire la viteză)
- introducerea unei coloane noi care ar conține restul împărțirii indicelui înregistrărilor la 12, apoi împărțirea pe baza acestei noi coloane, cu dezavantajul de a avea o coloană suplimentară în seturile rezultate
- folosind parametrul maxRecordsPerFile din Spark, a se vedea algoritmul 6 din teză, disponibil din versiunea Spark 2.2

Având în vedere, cazurile în care mai multe sarcini ar accesa același fișier, ultima soluție a fost utilizată pentru divizarea datelor în nu 12, ci 20 de fișiere pentru a evita potențialul acces concurent la fișiere.

Rezultatele dinTabelul 2arată faptul că în acest ultim scenariu Spark rulează într-adevăr pe mai multe nuclee, depășind viteza de rulare a Spark-ului din scenariul 1 de cinci ori și, ajungând să echivaleze viteza de rulare al lui scikit-learn. Diferența vizibilă de acuratețe dintre scikit-learn și Spark se datorează diferiților optimizatori utilizați de cei doi, dar este cea mai apropiată comparație care poate fi realizată folosind implementarea algoritmilor disponibili implicit în Spark.

Tabelul 2: Compararea scenariului 2 și 3 referitoare la Spark, cu experimentele scikit-learn.

	data100		*data1k*		*data10k*		*data100k*	
	Acc.	**Time**	**Acc.**	**Time**	**Acc.**	**Time**	**Acc.**	**Time**
Spark S2	0.36	0:11:00	0.33	1:34:02	0.35	17:03:22	0.35	176:29:48
Spark S3	0.354	0:02:36	0.354	0:18:37	0.308	3:43:26	0.352	36:43:53
Scikit-learn	0.53	0:02:41	0.63	0:18:18	0.64	2:42:00	N/A	N/A

Scalarea cu Spark

Spark poate rula la fel de repede ca scikit-learn pe o singură mașină, după cum s-a dovedit mai sus, utilizând întreaga putere a procesorului. Următorul pas a fost încercarea de a obține un nou record de viteză în ceea ce privește timpul de antrenament cu Spark. Împreună cu acesta, un alt obiectiv a fost spargerea barierei privind dimensiunea datelor care pot fi prelucrate într-o singură rulare. Ambele obiective ar putea fi realizate teoretic prin creșterea puterii de procesare, prin adăugarea de noi noduri la cluster.

Clusterul pentru următoarele teste a fost realizat având șapte stații de lucru cu aceeași configurație hardware. Acestea folosesc hard disk-uri pentru stocare, 16 GB de memorie RAM

și un CPU cu opt nuclee. Stațiile de lucru au fost conectate fizic la un comutator cu o viteză de legătură de 100 Mpbs. Un calculator a fost configurat pentru a găzdui managerul și driverul, iar restul au fost configurate ca mașini de lucru. Fiecare nod de lucru a fost configurat pentru a produce doi executori cu 3 GB de memorie RAM și 4 nuclee CPU. Rezultatele din Tabelul 3 arată că, având în vedere această nouă configurare, Spark se dovedește a fi mai lent pentru seturile de date mici decât în configurația din scenariul 3, dar rezultatele devin progresiv mai bune pe măsură ce dimensiunea datelor crește, având în vedere o tendință de creștere logaritmică. Tabelul arată, de asemenea, că antrenarea pe întregul set de date este posibilă folosind acest sistem integrat de analiză a datelor pe mai multe mașini și durează aproximativ 45 de ore.

Tabelul 3: Compararea unui cluster Spark având șapte noduri, cu scenariul 3 Spark.

	data100		*data1k*		*data10k*		*data100k*		*all*	
	Acc.	**Time**	**Acc.**	**Time**	**Acc.**	**Time**	**Acc.**	**Time**	**Acc.**	**Time**
Spark S3	0.354	0:02:36	0.354	0:18:37	0.308	3:43:26	0.352	36:43:53	N/A	N/A
Spark cluster	0.35	0:09:17	0.35	0:18:14	0.3	1:17:45	0.346	9:49:37	0.352	44:29:34

5.4 Concluzii și probleme nerezolvate

Contribuția prezentată în acest capitol este semnificativă din mai multe privințe. Acesta descrie un sistem integrat de analiză a datelor, care este capabil să proceseze pe deplin seturi mari de date tabelare folosind o configurare de cluster multi-nod. Această contribuție este semnificativă, având în vedere că multe dintre abordările similare existente sunt capabile să prelucreze numai un subset din seturile de date similar de mari.

Versiunea actuală a sistemului este concepută pentru a optimiza faza de antrenare a procesului de analiză a datelor. În plus, este important de remarcat faptul că arhitectura sistemului de analiză a datelor permite reproiectarea rutinelor relevante de prelucrare a datelor. Acest lucru permite implementarea viitoarelor iterații optimizate ale sistemului într-un mod eficient, astfel devenind disponibile îmbunătățiri utile, cum ar fi o analiză mai puternică a acurateței. Sistemul complet folosit în efectuarea cercetării din această secțiune este disponibil la adresa: https://github.com/akerestely/hpc-hadoop-spark.

Deși multe dintre întrebările din introducere au primit cel puțin un răspuns cel puțin parțial pe parcursul capitolului, un rezumat este prezentat, pentru o corelare rapidă.

- *Cât este considerat ca fiind un volum mare de date?* - În funcție de biblioteca de procesare utilizată, datele care depășesc dimensiunea unei memorii RAM tipice (≈ 32 GB) ar trebui să fie deja considerate ca fiind date de volum mare.

- *Este Spark suficient de bun pentru procesarea volumelor mari de date?* - Spark s-a dovedit a fi bun pentru prelucrarea datelor tabelare disponibile pentru această cercetare cu rețele neuronale artificiale. Sistemul a reacționat într-un mod pozitiv la volume de date din ce în ce mai mari, deci putem să presupunem că ar putea gestiona orice cantitate de date. Cu toate acestea, pentru alte tipuri de date sau alți algoritmi de învățare automată, această concluzie ar putea fi ușor diferită.
- *Este Spark un instrument scalabil de rulare a algoritmilor de învățare automată?* - Prin definiție, Spark împarte o problemă în mai multe subprobleme, putând astfel să gestioneze date în creștere, desigur cu dezavantajul unui calcul mai lent, deoarece rezultatele acestor mici probleme trebuie să fie fuzionate la un moment dat, ceea ce, la rândul său, necesită și putere de calcul. Pe de altă parte, sistemul poate fi extins prin noduri suplimentare de lucru, ceea ce oferă mai multă putere de calcul, rezultând astfel într-o procesare mai rapidă. Toate aceste concluzii sunt susținute de experimentele din acest capitol.
- *Pot fi rezolvate restricțiile de memorie utilizând Spark?* - Da, deoarece Spark este conceput să împartă o problemă în mai multe subprobleme, poate folosi atâta memorie cât este disponibilă pe nodul de lucru respectiv, astfel încât restricțiile de memorie disponibile în alte biblioteci nu se aplică în cazul utilizării bibliotecii Spark.
- *Necesită o mentalitate diferită procesul de prelucrare a volumelor mari de date?* - Prelucrarea volumelor mari de date necesită într-adevăr o mentalitate diferită, una în care accentul se pune pe paralelizare. Deși cea mai mare parte a paralelizării este deja făcută și disponibilă atunci când se utilizează algoritmii existenți de învățare automată, datele introduse în sistem trebuie să fie împărțite în prealabil, pentru a permite procesarea paralelă.
- *De ce și când ar trebui să se utilizeze Spark pentru învățarea automată?* - În primul rând, dacă un sistem Spark este deja pus în funcțiune și este disponibil, experimentele au arătat că datele care depășesc 10 MB pot fi prelucrate mai rapid cu Spark decât cu scikit-learn. Pe de altă parte, dacă un sistem Spark nu este disponibil, configurarea unuia nu este simplă, astfel încât recomandarea acestei cercetări este de a începe configurarea și utilizarea Spark numai dacă alte metode nu au reușit să dea roade.
- *Ar putea beneficia de puterea Spark seturile de date mai mici, sau acesta este potrivit doar pentru seturile mari de date?* - În cazul tipului setului de date, al algoritmului de învățare automată și al configurației clusterului utilizat în această cercetare, seturile de date care depășesc 10 MB pot beneficia deja de puterea lui Spark, astfel chiar și seturile de date mai mici pot beneficia de puterea lui Spark.

În sprijinul acestui capitol au fost publicate următoarele articole de cercetare:

- *High Performance Computing for Machine Learning*, de Árpád Kerestély, în Buletinul Universității Transilvania din Brașov, 2020. Această lucrare își propune să dezvăluie

modul în care învățarea automată și calculul de înaltă performanță beneficiază unul de celălalt, constituind astfel baza pentru partea de revizuire a literaturii din acest capitol.

- *A Research Study on Running Machine Learning Algorithms on Big Data with Spark*, de Árpád Kerestély, Alexandra Băicoianu și Răzvan Bocu, în Proceedings of the 14th International Conference on Knowledge Science, Engineering and Management (KSEM), 2021. Această lucrare descrie un sistem integrat de analiză a datelor bazat pe învățare automată, care procesează cantități mari de date, fiind astfel baza pentru partea experimentală și analiza comparativă a capitolului.

Bibliografie

[1] Martín Abadi et al. "Tensorflow: A system for large-scale machine learning". 12th USENIX symposium on operating systems design and implementation (OSDI 16). 2016, pp. 265–283.

[2] Pedro Henriques Abreu et al. "Predicting Breast Cancer Recurrence Using Machine Learning Techniques: A Systematic Review". In: (2016).

[3] Abien Fred M Agarap. "On breast cancer detection: an application of machine learning algorithms on the wisconsin diagnostic dataset". In: Proceedings of the 2nd International Conference on Machine Learning and Soft Computing. 2018, pp. 5–9.

[4] Bassam Al-Shargabi and Fida'a Al-Shami. "An experimental study for breast cancer prediction algorithms". In:Proceedings of the Second International Conference on Data Science, E-Learning and Information Systems. 2019, pp. 1–6.

[5] R Almufti et al. "A critical review of the analytical approaches for circulating tumor biomarker kinetics during treatment". In: Annals of oncology 25.1 (2014), pp. 41–56.

[6] G Ammar, W Dayawansa, and C Martin. "Exponential interpolation: theory and numerical algorithms". In: Applied Mathematics and Computation 41.3 (1991), pp. 189–232.

[7] Apache Flume. URL: https://flume.apache.org/ (visited on 06/17/2021).

[8] Apache Hadoop. URL: http://hadoop.apache.org/ (visited on 06/17/2021).

[9] Apache HBase. URL: https://hbase.apache.org/ (visited on 06/17/2021).

[10] Apache Hive. URL: https://hive.apache.org/ (visited on 06/17/2021).

[11] Apache Spark. URL: https://spark.apache.org/ (visited on 06/17/2021).

[12] Apache Sqoop. URL: https://sqoop.apache.org/ (visited on 06/17/2021).

[13] Apache Storm. URL: https://hortonworks.com/apache/storm/ (visited on 06/17/2021).

[14] J Archenaa and EA Mary Anita. "A survey of big data analytics in healthcare and government". In: Procedia Computer Science 50 (2015), pp. 408–413.

[15] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling". In: arXiv preprint arXiv:1803.01271 (2018).

[16] Alexandra Baicoianu and Andreea Mathe. "Diagnose Bearing Failures With Machine Learning Models". In: 2021 International Conference on INnovations in Intelligent SysTems and Applications (INISTA). IEEE. 2021, pp. 1–6.

[17] Ross S Berkowitz and Donald P Goldstein. "Molar pregnancy". In:New England Journal of Medicine 360.16 (2009), pp. 1639–1645.

[18] Declan Butler. "When Google got flu wrong". In: Nature News 494.7436 (2013), p. 155.

[19] Antonio Cachuan. A gentle introduction to Apache Arrow with Apache Spark and Pandas. 2019. (Visited on 06/01/2021).

[20] Cancer databases. URL: https://public.opendatasoft.com/explore/dataset/cancer-databases/table/.

[21] Enrique Castillo and Ali S Hadi. "Functional networks". In: Wiley StatsRef: Statistics Reference Online (2014).

[22] Fay Chang et al. "Bigtable: A distributed storage system for structured data". In: ACM Transactions on Computer Systems (TOCS) 26.2 (2008), pp. 1–26.

[23] Min Chen et al. "Disease prediction by machine learning over big data from healthcare communities". In: Ieee Access 5 (2017), pp. 8869–8879.

[24] Yunji Chen et al. "Dadiannao: A machine-learning supercomputer". In:2014 47th Annual IEEE/ACM International Symposium on Microarchitecture. IEEE. 2014, pp. 609–622.

[25] Kyunghyun Cho et al. "Learning phrase representations using RNN encoder-decoder for statistical machine translation". In: arXiv preprint arXiv:1406.1078 (2014).

[26] Edward Choi et al. "Doctor ai: Predicting clinical events via recurrent neural networks". In: Machine learning for healthcare conference. PMLR. 2016, pp. 301–318.

[27] François Chollet et al. Keras. https://keras.io. 2015.

[28] Niamh Clarke et al. "GDPR: an impediment to research?" In: Irish Journal of Medical Science (1971-) 188.4 (2019), pp. 1129–1135.

[29] Adam Coates et al. "Deep learning with COTS HPC systems". In:International conference on machine learning. PMLR. 2013, pp. 1337–1345.

[30] European Commission.Regulation EU 2016/679 of the European Parliament and of the Council. 2016. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679 (visited on 07/2021).

[31] Catherine Costigan, Sabin Tabirca, and John Coulter. "Mathematically Modelling hCG in Women with Gestational Trophoblastic Disease Using Logarithmic Transformations". In: 2016 UKSim-AMSS 18th International Conference on Computer Modelling and Simulation (UKSim). IEEE. 2016, pp. 55–59.

[32] Karen Daniels et al. "Properties of normalized radial visualizations". In:Information Visualization 11.4 (2012), pp. 273–300.

[33] Saptarshi Das and Koushik Maharatna. "Machine learning techniques for remote healthcare". In: Systems Design for Remote Healthcare. Springer, 2014, pp. 129–172.

[34] Databricks. Parquet files. 2020. URL: https://docs.databricks.com/data/data-sources/readparquet.html (visited on 02/2020).

[35] data.world. URL: https://data.world/datasets/cancer.

[36] Colin Davis. Big Data on a Laptop: Tools and Strategies – Part 3. Ed. by POP Tech. 2018. URL: https://tech.popdata.org/big-data-on-a-laptop-tools-and-strategies-part-3/ (visited on 06/01/2021).

[37] Jeffrey Dean et al. "Large Scale Distributed Deep Networks". In: NIPS. 2012.

[38] Michael Driscoll. Winning with Big Data: Secrets of the Successful Data Scientist. Ed. by Inc. O'Reilly Media. 2010. URL: https://conferences.oreilly.com/datascience/public/schedule/detail/15316.

[39] JC Ehiwario and SO Aghamie. "Comparative study of bisection, Newton-Raphson and secant methods of root-finding problems". In: IOSR Journal of Engineering 4.04 (2014), pp. 01–07.

[40] Emad Elsebakhi et al. "Large-scale machine learning based on functional networks for biomedical big data with high performance computing platforms". In: Journal of Computational Science 11 (2015), pp. 69–81.

[41] Heiko Enderling and Mark AJ Chaplain. "Mathematical Modeling of Tumor Growth and Treatment". In: Current pharmaceutical design 20.30 (2014), pp. 4934–4940.

[42] George Elmer Forsythe. "Computer methods for mathematical computations." In: Prentice-Hall series in automatic computation 259 (1977).

[43] C Freitas et al. "Comparison of vibration and acoustic measurements for detection of bearing defects". In:International Conference on Noise and Vibration Engineering 2016 and International Conference on Uncertainty in Structural Dynamics 2016. Vol. 1. 2016.

[44] Henri Gavin. "The Levenberg-Marquardt method for nonlinear least squares curve-fitting problems". In: Department of Civil and Environmental Engineering, Duke University 28 (2011), pp. 1–5.

[45] Marzyeh Ghassemi, Leo Anthony Celi, and David J Stone. "State of the art review: the data revolution in critical care". In: Critical Care 19.1 (2015), pp. 1–9.

[46] Dan Gillick, Arlo Faria, and John DeNero. "Mapreduce: Distributed computing for machine learning". In: Berkley, Dec 18 (2006).

[47] GLOBOCAN (Global Cancer Observatory). Cancer Statistics – Romania. May 2020. URL: https://gco.iarc.fr/today/data/factsheets/populations/642-romania-fact-sheets.pdf.

[48] Isabelle Guyon et al. "Gene selection for cancer classification using support vector machines". In: Machine learning 46.1 (2002), pp. 389–422.

[49] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: Neural computation 9.8 (1997), pp. 1735–1780.

[50] Patrick Hoffman et al. "DNA visual and analytic data mining". In: Proceedings. Visualization'97 (Cat. No. 97CB36155). IEEE. 1997, pp. 437–441.

[51] Shiuh-Jer Huang and Chien-Lo Huang. "Control of an inverted pendulum using grey prediction model". In: Industry Applications, IEEE Transactions on 36.2 (2000), pp. 452–458.

[52] Introduction to High-Performance Machine learning @SURFsara. 2018. URL: https://events.prace-ri.eu/event/693/attachments/626/ (visited on 06/17/2021).

[53] Tado Juric. "Google Trends as a method to predict new COVID-19 cases". In: medRxiv (2021).

[54] Marcel Adam Just et al. "Machine learning of neural representations of suicide and emotion concepts identifies suicidal youth". In: Nature human behaviour 1.12 (2017), pp. 911–919.

[55] Kadupitiya Kadupitige. Intersection of HPC and Machine Learning. Indiana University Bloomington, 2017. URL: http://dsc.soic.indiana.edu/publications/ENGR-E%20687%20_%20IND%20STUDY%20INTEL%20SYS%20_%20Intersection%20of%20HPC%20and%20machine%20learning.pdf (visited on 6/17/2021).

[56] Holden Karau and Rachel Warren. High performance Spark: best practices for scaling and optimizing Apache Spark. "O'Reilly Media, Inc.", 2017.

[57] Freund Karl. What's Hot At SC17: The Synthesis Of Machine Learning & HPC. 2017. URL: https://www.forbes.com/sites/moorinsights/2017/11/14/whats-hot-at-sc17-the-synthesis-ofmachine-learning-hpc/#2ef32b2759a7 (visited on 06/17/2021).

[58] Arpad Kerestely. "Feature Inspection and Elimination in the Context of Breast Cancer Prediction". In: Proceedings of the 36th International Business Information Management Association (IBIMA). Granada, Spain, 2020, pp. 13487–13493. ISBN: 978-0-9998551-5-7.

[59] Arpad Kerestely. "High Performance Computing for Machine Learning". In: Bulletin of the Transilvania University of Brasov. Mathematics, Informatics, Physics. Series III 13.2 (2020), pp. 705–714.

[60] Arpad Kerestely, Alexandra Baicoianu, and Razvan Bocu. "A Research Study on Running Machine Learning Algorithms on Big Data with Spark". In: Proceedings of the 14th International Conference on Knowledge Science, Engineering and Management (KSEM 2021). Tokyo, Japan: Springer, 2021, pp. 307–318.

[61] Arpad Kerestely, Catherine Costigan, and Sabin Tabirca. "Vertically Shifted Exponential Best-Fit". In: Proceedings of the 35th International Business Information Management Association (IBIMA). Seville, Spain, 2020, pp. 13855–13868. ISBN: 978-0-9998551-4-0.

[62] Arpad Kerestely, Lucian Mircea Sasu, and Marius Sabin Tabirca. "Machine Learning in Healthcare: An Overview". In: Bulletin of the Transilvania University of Brasov. Mathematics, Informatics, Physics. Series III 11.2 (2018), pp. 273–278.

[63] Arpad Kerestely et al. "Theoretical Study of Exponential Best-Fit: Modeling hCG for Gestational Trophoblastic Disease". In: Proceedings of the 14th International Conference on Knowledge Science, Engineering and Management (KSEM 2021). Tokyo, Japan: Springer, 2021, pp. 426–438.

[64] Konstantina Kourou et al. "Machine learning applications in cancer prognosis and prediction". In: Computational and structural biotechnology journal 13 (2015), pp. 8–17.

[65] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: Advances in neural information processing systems 25 (2012), pp. 1097–1105.

[66] Andrew Kusiak. "Smart manufacturing". In: International Journal of Production Research 56.1-2 (2018), pp. 508–517.

[67] Prasanth Lade, Rumi Ghosh, and Soundar Srinivasan. "Manufacturing analytics and industrial internet of things". In: IEEE Intelligent Systems 32.3 (2017), pp. 74–79.

[68] Colin Lea et al. "Temporal convolutional networks: A unified approach to action segmentation". In: European Conference on Computer Vision. Springer. 2016, pp. 47–54.

[69] Megan Lenhart. "Diagnosis and treatment of molar pregnancy". In: Topics in Obstetrics & Gynecology 27.17 (2007), pp. 1–4.

[70] John Levesque and Aaron Vose. Programming for Hybrid Multi/Manycore MPP Systems. CRC Press, 2017.

[71] Zachary C Lipton, David Kale, and Randall Wetzel. "Directly modeling missing data in sequences with rnns: Improved classification of clinical time series". In:Machine learning for healthcare conference. PMLR. 2016, pp. 253–270.

[72] Pek Y Lum et al. "Extracting insights from the shape of complex data using topology". In:Scientific reports 3 (2013), p. 1236.

[73] Mehran Mozaffari-Kermani et al. "Systematic poisoning attacks on and defenses for machine learning in healthcare". In:IEEE journal of biomedical and health informatics 19.6 (2014), pp. 1893–1905.

[74] Abhinav Nagpal and Goldie Gabrani. "Python for data analytics, scientific and technical applications". In: 2019 Amity international conference on artificial intelligence (AICAI). IEEE. 2019, pp. 140–145.

[75] Open Datasets and Machine Learning Projects. URL: https://www.kaggle.com/datasets.

[76] Patison Palee et al. "Image analysis of histological features in molar pregnancies". In: Expert systems with applications 40.17 (2013), pp. 7151–7158.

[77] Patison Palee et al. "Heuristic neural network approach in histological sections detection of hydatidiform mole". In: Journal of Medical Imaging 6.4 (2019), p. 044501.

[78] Ramprasad Pedapatnam. Understanding Resource Allocation configurations for a Spark application. Ed. by Clairvoyant. 2016. URL: http://site.clairvoyantsoft.com/understanding-resourceallocation-configurations-spark-application/ (visited on 06/01/2021).

[79] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: Journal of Machine Learning Research 12 (2011), pp. 2825–2830.

[80] George M Phillips. Interpolation and approximation by polynomials. Vol. 14. Springer Science & Business Media, 2003.

[81] Narendra Pisal, John Tidy, and Barry Hancock. "Gestational trophoblastic disease: is intensive follow up essential in all women?" In:BJOG: An International Journal of Obstetrics & Gynaecology 111.12 (2004), pp. 1449–1451.

[82] Juliana T Pollettini et al. "Using machine learning classifiers to assist healthcare-related decisions: classification of electronic patient records". In: Journal of medical systems 36.6 (2012), pp. 3861–3874.

[83] William H Press et al. Numerical recipes 3rd edition: The art of scientific computing. Cambridge university press, 2007.

[84] Devi Ramanan. NKI Breast Cancer Data. 2017. URL: https://data.world/deviramanan2016/nkibreast-cancer-data.

[85] Gesine Richter et al. "Patient views on research use of clinical data without consent: Legal, but also acceptable?" In: European Journal of Human Genetics 27.6 (2019), pp. 841–847.

[86] Mark R Schoeberl. "A model for the behavior of β-hCG after evacuation of hydatidiform moles". In: Gynecologic oncology 105.3 (2007), pp. 776–779.

[87] scipy.optimize.curve_fit. URL: https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve_fit.html (visited on 08/2019).

[88] Michael J Seckl, Neil J Sebire, and Ross S Berkowitz. "Gestational trophoblastic disease". In: The Lancet 376.9742 (2010), pp. 717–729.

[89] Rebecca L. Siegel et al. "Cancer statistics, 2021". In:CA: A Cancer Journal for Clinicians 71.1 (2021), pp. 7–33. URL: https://acsjournals.onlinelibrary.wiley.com/doi/10.3322/caac.21654.

[90] John T Soper. "Gestational trophoblastic disease". In: Obstetrics & Gynecology 108.1 (2006), pp. 176–187.

[91] Apache Spark. PySpark Usage Guide for Pandas with Apache Arrow. 2020. URL: https://spark.apache.org/docs (visited on 02/2020).

[92] Ferenc Szidarovszky and Sidney J Yakowitz. Principles and procedures of numerical analysis. Vol. 14. Springer, 2013.

[93] UCI Machine Learning Repository: Data Sets. URL: https://archive.ics.uci.edu/ml/datasets.php.

[94] National Health Service UK. Molar pregnancy. 2020. URL: https://www.nhs.uk/conditions/molar-pregnancy/.

[95] Case Western Reserve University. The Case Western Reserve University Bearing Data Center Website. 2020. URL: https://csegroups.case.edu/bearingdatacenter (visited on 06/2020).

[96] National Cancer Institute USA. Gestational Trophoblastic Disease Treatment. 2020. URL: https://www.cancer.gov/types/gestational-trophoblastic/patient/gtd-treatment-pdq.

[97] NE Van Trommel et al. "Early identification of persistent trophoblastic disease with serum hCG concentration ratios". In:International Journal of Gynecological Cancer 18.2 (2008), pp. 318–323.

[98] Laura J Van't Veer et al. "Gene expression profiling predicts clinical outcome of breast cancer". In: nature 415.6871 (2002), pp. 530–536.

[99] Ashish Vaswani et al. "Attention is all you need". In: arXiv preprint arXiv:1706.03762 (2017).

[100] What is high performance computing? URL: https://insidehpc.com/hpc-basic-training/whatis-hpc/ (visited on 06/17/2021).

[101] William Wolberg, W. Street, and Olvi Mangasarian. Breast Cancer Wisconsin (Diagnostic) Data Set. UCI Machine Learning Repository. 1995. URL: https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic).

[102] Xiao Xiao et al. "On the use of log-transformation vs. nonlinear regression for analyzing biological power laws". In: Ecology 92.10 (2011), pp. 1887–1894.

[103] B You et al. "Predictive values of hCG clearance for risk of methotrexate resistance in low-risk gestational trophoblastic neoplasias". In: Annals of oncology 21.8 (2010), pp. 1643–1650.

[104] B You et al. "Early prediction of treatment resistance in low-risk gestational trophoblastic neoplasia using population kinetic modelling of hCG measurements". In: British journal of cancer 108.9 (2013), pp. 1810–1816.

[105] Tracey Young et al. "Predicting gestational trophoblastic neoplasia (GTN): is urine hCG the answer?" In: Gynecologic oncology 122.3 (2011), pp. 595–599.

[106] S. Zhang et al. "Machine learning and deep learning algorithms for bearing fault diagnostics-a comprehensive review". In: arXiv preprints arXiv:1901.08247 (2019).

[107] MH Zwietering et al. "Modeling of the bacterial growth curve". In: Applied and environmental microbiology 56.6 (1990), pp. 1875–1881.

[108] Matjaz Zwitter and Milan Soklic.Breast Cancer Data Set. UCI Machine Learning Repository. 1988. URL: https://archive.ics.uci.edu/ml/datasets/breast+cancer.