

UNIVERSITATEA BABEȘ-BOLYAI
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

Contribuții în rezolvarea problemelor din lumea reală folosind modele de învățare automată

Rezumatul Tezei de Doctorat

Student doctorand: Ionescu Vlad-Sebastian
Conducător de doctorat: Prof. Dr. Czibula Gabriela

2017

Cuvinte cheie: *învățare automată, arheologie, inginerie software, mașini cu vector suport.*

Cuprinsul rezumatului

Cuprinsul tezei de doctorat	2
Lista publicațiilor	4
Introducere	6
1 Arheologie	7
1.1 Determinarea staturii pe baza rămășițelor arheologice folosind lungimile oaselor lungi	7
1.2 Estimarea masei corporale în arheologie	7
1.3 Estimarea vârstei la moarte pe baza oaselor lungi	8
2 Inginerie software	9
2.1 Predicția defectelor	9
2.2 Estimarea efortului în dezvoltarea software	9
2.2.1 Modele de învățare automată care folosesc metrici software	10
2.2.2 Modele de învățare automate folosind procesare de text	11

Cuprinsul tezei de doctorat

Keywords	7
Acknowledgements	8
List of publications	9
Introduction	11
1 Background for Archaeological problems	14
1.1 Predicting stature from archaeological skeletal remains using long bone lengths	14
1.1.1 Motivation	14
1.1.2 Related work on the stature prediction problem	15
1.2 Body mass estimation in bioarchaeology	18
1.2.1 Motivation	18
1.2.2 Literature review	18
1.2 Age at death estimation from long bone lengths	19
1.3.1 Literature review	19
2 Contributions to machine learning models for archaeology	21
2.1 Proposed machine learning models	21
2.1.1 Genetic Algorithms	21
2.1.2 Support Vector Regression	22
2.1.3 Locally Weighted Regression	23
2.1.4 Data Preprocessing	24
2.1.5 Evaluation methodology	26
2.2 Computational Experiments	27
2.2.1 Stature prediction	27
2.2.2 Body mass estimation	49
2.2.3 Age at death estimation	57
3 Background for Software Engineering problems	63
3.1 Software defect prediction	63
3.1.1 Motivation	63
3.1.2 Related work	64
3.2 Software development effort estimation	65
3.2.1 Motivation	65
3.2.2 Related work	66
4 Contributions to machine learning models for software engineering	71
4.1 Proposed machine learning models	71
4.1.1 Fuzzy Self Organizing Maps	71
4.1.2 Fuzzy Decision Trees	75

4.1.3	Learning models for software development effort estimation	78
4.2	Computational experiments	84
4.2.1	Software defect detection using FSOM	84
4.2.2	Software Defect Prediction using FuzzyDT	89
4.2.3	Software Development Effort Estimation	97
Conclusions		111

Lista publicațiilor

Toate categoriile sunt enumerate conform clasificării din 2014 a jurnalelor¹ și conferințelor² din domeniul Informatică și serviciului web asociat³.

Publicații în ISI Web of Knowledge

Publicații în ISI Science Citation Index Expanded

1. [CIMM16] Gabriela Czibula, **Vlad-Sebastian Ionescu**, Diana-Lucia Miholca and Ioan-Gabriel Mircea Machine learning-based approaches for predicting stature from archaeological skeletal remains using long bone lengths. *Journal of Archaeological Science*, volume 69, pp. 85–99, 2016 (**2016 IF = 2.602**).
Categoria A, 4 puncte.
2. [IDC17] **Vlad-Sebastian Ionescu**, Horia Demian and Istvan-Gergely Czibula Natural language processing and machine learning methods for software development effort estimation. *Studies in Informatics and Control*, volume 26(2), pp. 219–228, 2017 (**2016 IF = 0.776**).
Categoria C, 2 puncte.
3. [CCMI16] Istvan-Gergely Czibula, Gabriela Czibula, Zsuzsanna-Edit Marian and **Vlad-Sebastian Ionescu** A Novel Approach Using Fuzzy Self-Organizing Maps for Detecting Software Faults. *Studies in Informatics and Control*, volume 25(2), pp. 207–216, 2016 (**2016 IF = 0.776**).
Categoria C, 1 punct.
4. [LI14] Georgia Irina Oros, Gheorghe Oros, Alina Alb Lupaș and **Vlad Ionescu** Differential subordinations obtained by using a generalization of Marx-Strohhacker theorem. *Journal of Computational Analysis and Applications*, volume 20(1), pp. 135–139, 2016, (**2016 IF = 0.609**).
Categoria C, 1 punct.
5. [OOLI16] Andrei Lorian and **Vlad Ionescu** Some differential superordinations using Ruscheweyh derivative and generalized Sălăgean operator. *Journal of Computational Analysis and Applications*, volume 20(1), pp. 437–444, 2014, (**2016 IF = 0.609**).
Categoria C, 2 puncte.

¹<http://informatica-universitaria.ro/getpfile/16/CSafisat2.pdf>

²http://informatica-universitaria.ro/getpfile/16/CORE2013_Exported.xlsx

³<http://informatica-universitaria.ro/php/index.html>

Publicații în ISI Conference Proceedings Citation Index

1. [Ion17] **Vlad-Sebastian Ionescu**. An approach to software development effort estimation using machine learning. In *2017 IEEE International Conference on Intelligent Computer Communication and Processing*, IEEE Computer Society, accepted for publication, 2017 (**indexed IEEE**).
Categoria C, 2 puncte.
2. [ITV16] **Vlad-Sebastian Ionescu**, Mihai Teletin and Estera-Maria Voiculescu Machine learning techniques for age at death estimation from long bone lengths In *IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI 2016)*, IEEE Hungary Section, pp. 457–462, 2016.
Categoria C, 2 puncte.
3. [ICT16] **Vlad-Sebastian Ionescu**, Gabriela Czibula and Mihai Teletin Supervised learning techniques for body mass estimation in bioarchaeology *IEEE 7th International Workshop on Soft Computing Applications*, accepted, 2016, in press.
Categoria C, 2 puncte.

Publicații în jurnale și conferințe internaționale

1. [IMMC15] **Vlad-Sebastian Ionescu**, Ioan-Gabriel Mircea, Diana-Lucia Miholca, and Gabriela Czibula. Novel instance based learning approaches for stature estimation in archaeology. In *2015 IEEE International Conference on Intelligent Computer Communication and Processing*, IEEE Computer Society, pp. 309–316, 2015 (**indexed IEEE**).
Categoria C, 1 punct.
2. [Ion15a] **Vlad-Sebastian Ionescu** New supervised learning approaches for sex identification in archaeology In *International Virtual Research Conference In Technical Disciplines*, EDIS - Publishing Institution of the University of Zilina, pp. 56–64, 2015 (**indexed Google Scholar**).
Categoria D, 1 punct.
3. [Ion15b] **Vlad-Sebastian Ionescu** Support vector regression methods for height estimation in archaeology. *Studia Universitatis Babes-Bolyai Series Informatica*, volume LX(2), pp. 70–82, 2015 (**indexed Mathematical Reviews**).
Categoria D, 1 punct.
4. [MCMI16] Zsuzsanna-Edit Marian, Istvan-Gergely Czibula, Ioan-Gabriel Mircea and **Vlad-Sebastian Ionescu** A study on software defect prediction using fuzzy decision trees. *Studia Universitatis Babes-Bolyai Series Informatica*, volume LXI(2), pp. 5–20, 2016 (**indexed Mathematical Reviews**).
Categoria D, 0 puncte.

Scorul total al publicațiilor: **19 puncte.**

Introducere

Aceasta teză de doctorat constă în cercetări privind aplicarea tehnicilor inteligenței computaționale pentru rezolvarea unor probleme practice în domeniile arheologiei și a ingineriei software. Toate cercetările originale au fost realizate sub supervizarea Prof. Dr. Gabriela Czibula.

Cele două domenii au fost alese datorită existenței mai multe probleme practice dificile în ambele, care am considerat că pot fi de o importanță majoră pentru cercetătorii din cele două domenii și care pot fi abordate unitar prin învățare automată. Ipoteza noastră s-a dovedit a fi validă: nu mai există abordări similare pentru problemele alese, rezultatele obținute sunt mai bune și folositoare cercetătorilor din aceste domenii și abordările noastre au ca temă centrală învățarea automată.

Pentru arheologie, focusul a fost pe aplicarea mai multor modele de învățare automată – algoritmi genetici (AG), regresie cu vector suport și regresie ponderată local – pe trei probleme importante în arheologie: determinarea staturii, estimarea greutateii, și estimarea vârstei la moarte.

Fiecare model de învățare a fost aplicat pe toate cele trei probleme enumerate, rezultând într-o abordare unitară a problemelor arheologice prin învățare automată. Abordările noastre sunt inovative relativ la domeniul arheologiei și au fost publicate în jurnale și conferințe [CIMM16, ITV16, ICT16, IMMC15, Ion15b].

Legat de inginerie software, am abordat două probleme importante din literatura ingineriei software bazate pe căutare: predicția defectelor și estimarea efortului în dezvoltarea software. Am propus mai multe modele de învățare automată pentru rezolvarea acestor probleme și am efectuat experimente pe diverse seturi de date legate de acestea folosind self organizing maps și arbori de decizie augmentate cu tehnici fuzzy și regresie cu vector suport. Modelele noastre sunt fie complet originale, fie nu au mai fost aplicate în acest scop. Rezultatele obținute sunt mai bune decât majoritatea abordărilor existente.

Rezolvările originale propuse pentru aceste probleme sunt reunite sub tema centrală de învățare automată pentru ingineria software, fiind utile pentru dezvoltatori și manageri deopotrivă. Abordările noastre sunt publicate în [CCMI16, MCM16, Ion17, IDC17].

Capitolul 1

Arheologie

În acest capitol sunt prezentate cunoștințele teoretice de arheologie privind statura, masa corporală și vârsta necesare pentru abordările din teza. Aceste informații au fost culese pentru a facilita cercetările originale publicate în [CIMM16, ITV16, ICT16].

1.1 Determinarea staturii pe baza rămășițelor arheologice folosind lungimile oaselor lungi

Din perspectiva arheologiei, determinarea staturii unui individ pe baza informațiilor osteologice este o problema fundamentală. Abordările matematice clasice folosesc metode de regresie bazate pe analize statistice a datelor existente.

Statura este un atribut biologic standard, împreună cu vârsta și masa corporală. Acestea permit cercetătorilor să determine diferențe sexuale și dimensiuni corporale ale unei populații din trecut [RR06]. Mai mult, statura este un indicator al dezvoltării fizice a unui individ în societatea sa și în mediul său natural [BR14].

Odată cu dezvoltarea inteligenței computaționale și a învățării automate, este naturală cercetarea posibilității de a dezvolta programe informatice care nu necesită algoritmi predefiniți pentru generarea predicțiilor, ci sunt capabile să învețe din date și să se adapteze unor noi date în momentul în care acestea devin disponibile.

1.2 Estimarea masei corporale în arheologie

Estimarea masei corporale din rămășițele scheletice umane reprezintă o problemă de importanță majoră în cercetarea arheologică. Există consens în literatura bioarheologică că măsurătorile postcraniene sunt direct legate de mărimea corpului și oferă cele mai exacte estimări [B.00].

O problemă majoră în cercetarea referitoare la masa corporală este cauzată de lipsa unor criterii de referință disponibile publicului.

Estimarea masei corporale este o problemă foarte importantă pentru arheologia modernă. Aceasta poate oferi anumite cunoștințe despre populațiile din trecut, cum ar fi indicatorii pentru sănătatea populației anterioare, efectele diferiților factori de mediu asupra populațiilor din trecut (de exemplu, strategia de subzistență, factorii climatici), aspectele sociale etc. Abilitatea de a obține estimări corecte ale masei corporale de la schelete este, de asemenea, esențială în anchetele criminalistice referitoare la rămășițele scheletice neidentificate [Moo08]. În consecință, este esențial ca bioarheologii să dezvolte și să utilizeze metode de estimare a masei corporale care să fie cât mai exacte cu putință.

Cu toate acestea, proiectarea unei metode corecte pentru rezolvarea acestei probleme rămâne o mare provocare, deoarece există mulți factori care ar trebui luați în considerare

[AR04]. Câteva dintre deciziile care trebuie luate în acest proces [RHN+12]: care sunt cele mai relevante măsurători scheletice de utilizat, care este abordarea statistică adecvată care trebuie aplicată, care ar trebui să fie eșantionul de scheleți utilizat.

1.3 Estimarea vârstei la moarte pe baza oaselor lungi

Estimarea vârstei la moarte este o problemă care cere să se găsească o estimare bună pentru vârsta la moarte a unor rămășițe biologice umane. Aceste rămășițe pot fi oase, dinți, un corp bine conservat etc. În cazul nostru, luăm în considerare lungimile oaselor lungi și suntem interesați să găsim, folosind învățarea automată, o funcție matematică ce ne dă vârsta la moarte a unui subiect.

Estimarea vârstei la moarte este esențială pentru determinarea demografiei populației și efectuarea analizei individuale a rămășițelor umane. Pentru copii și adulți tineri, problema este mai ușoară datorită faptului că dezvoltarea biologică apar în timpul acestor etape.

Problema este una complicată pentru scheletele adulte. Acest lucru se datorează faptului că indicatorii de vârstă sunt variabili din punct de vedere biologic și că anumite trăsături răspund diferit față de factorii de mediu pe parcursul vieții cuiva, deci diferențe mult mai individuale care nu ajută la identificarea unei vârste exacte (zgomot) [CD13].

Deoarece problema este atât de complexă, nu există algoritmi exacti pentru rezolvarea acesteia. Există formule simple de regresie care iau în considerare una sau mai multe oase, dar ele sunt de obicei create pentru fiecare set de date și nu este clar dacă acestea ar funcționa bine sau nu pe alte seturi de date. Prin utilizarea automate, putem veni cu modele care să fie ușor de recalificat pe noi seturi de date dacă este necesar. Prin utilizarea metodelor de validare, putem estima cât de bine vor funcționa algoritmi noștri pe date necunoscute de același tip.

Ne ocupăm în teză cu vârsta la moarte în mai multe categorii de vârstă, dar sfătuim cititorul să interpreteze rezultatele noastre în raport cu ceea ce este cunoscut în literatura de specialitate ca fiind fezabil pentru fiecare grupă de vârstă în special.

Capitolul 2

Inginerie software

În acest capitol, prezentăm cunoștințele de bază ale ingineriei software (software engineering) în ceea ce privește predicția defectelor software și estimarea efortului în dezvoltarea software-ului. Rezultatele prezentate în acest capitol al tezei au fost publicate în [CCMI16, MCM16, IDC17, Ion17].

2.1 Predicția defectelor

Predicția defectelor software reprezintă activitatea de identificare a modulelor software care sunt susceptibile la dezvoltarea erori într-o versiune viitoare a unui sistem software, fiind de o importanță majoră pentru testarea software-ului și pentru asigurarea calității acestuia. Metodele de detectare a unor entități software defecte sunt utile pentru a sugera dezvoltatorilor modulele software care ar trebui să fie testate mai riguros. Aceste entități software pot fi componente, module, pachete, clase, metode, funcții sau alte elemente constitutive.

Pentru creșterea eficienței procesului de reparare a defecțiunilor software, modelele de predicție a defecțiunilor sunt utile pentru anticiparea locațiilor dintr-un sistem software în care în viitor pot apărea defecte. Identificarea defectelor software este dificilă, în principal pentru proiectele complexe. Principala dificultate legată de construirea predictorilor supravegheați este faptul că defectele proiectelor software sunt mult mai puține decât entitățile non-defecte și, prin urmare, datele de antrenament sunt extrem de dezechilibrate [ATS15].

O altă problemă importantă legată de identificarea defecțiunilor software este detecția defecțiunilor software, care reprezintă procesul de identificare a modulelor software care conțin deja erori și contribuie la creșterea eficacității procesului de asigurare a calității. Metodele de detectare a defecțiunilor sunt utile pentru a sugera dezvoltatorilor pe care module de software ar trebui să se concentreze în timpul testării, în special atunci când, din lipsă de timp, modulele nu pot fi testate sistematic.

Revizuirea codului este o activitate frecvent utilizată în procesele de dezvoltare Agile pentru menținerea calității sistemului software. În timpul revizuirii codului, un programator cu experiență revizuieste codul sursă pentru a identifica vulnerabilitățile, problemele de securitate și alte probleme trecute cu vederea de către implementatorul inițial. Deoarece revizuirea codului este o activitate consumatoare de timp și costisitoare, detectarea defectelor software poate fi utilizată pentru a ghida procesul de revizuire a codului prin identificarea secțiunilor din cod care ar putea fi marcate pentru revizuire în timpul unei sesiuni de examinare a codului, din cauza diferitelor probleme legate de acest cod.

2.2 Estimarea efortului în dezvoltarea software

Estimarea efortului de dezvoltare a software-ului reprezintă acțiunea de estimare a timpului necesar pentru ca fiecare parte a unui sistem software să fie finalizată în timpul fazei de

dezvoltare a produsului. Aceste estimări se efectuează fie la nivel de conducere, fie la nivelul programatorilor responsabili pentru acțiunile care vor trebui efectuate. Estimările precise sunt importante pentru planificarea corectă a procesului de dezvoltare și alocarea resurselor umane în consecință.

Există mai multe metodologii și protocoale de estimare, inclusiv cele software-suportate. Două dintre cele mai populare sunt estimări simple ad-hoc, în care un programator oferă o estimare a timpului bazată pe descrierea sarcinii și experiența sa cu ceea ce este necesar, precum și metodele de grup, cum ar fi Planning Poker, metode mai complexe care necesită consens din partea unei echipe de programatori [Coh05]. Ambele sunt considerate estimări expert, deoarece experții în domeniu sunt cei care furnizează estimările.

Metodele de grup au avantajul de a fi mai precise și, în cazul Planning Poker, permit rafinarea estimărilor în timp. Avantajul metodelor mai complexe este o estimare mai bună decât metodele ad-hoc, în general, însă acestea necesită mult mai mult timp care ar putea fi utilizat pentru dezvoltarea reală. Din câte știm, nu există studii care să analizeze acest compromis.

Metodologiile asistate de software constau într-un sistem informatic care oferă o estimare sau ajută programatorii să ajungă la una, prin analizarea algoritmică a părților cerințelor sau a software-ului aflat în curs de dezvoltare. Cele mai frecvente metode utilizate în prezent sunt modelele parametrice, cum ar fi modelul COCOMO și modelul Putnam.

COCOMO folosește trei formule care furnizează efortul necesar în luni-persoană, timpul de dezvoltare în luni și numărul de persoane necesar. Acesta utilizează constante predefinite pentru fiecare dintre cele trei tipuri de proiecte considerate. Nu ține cont de experiența dezvoltatorilor, de resursele hardware disponibile și de alți factori [Boe01, BCH⁺00]. Modelul Putnam utilizează o formulă pentru efort (exprimată în ani-persoană) care include dimensiunea produsului în curs de dezvoltare, productivitatea companiei și timpul total programat pentru proiect [PM03].

Cele mai multe dintre metodele automate existente se bazează pe metrici software de relevanță discutabilă și pe alte estimări umane, cum ar fi complexitatea și productivitatea. Acest lucru le face extrem de predispuse la erori mari și departe de abordarea intuitivă pe care majoritatea programatorilor o folosesc atunci când furnizează estimări.

2.2.1 Modele de învățare automată care folosesc metrici software

Prin modelele de învățare automată care utilizează metrici software, înțelegem modele clasice cum ar fi COCOMO, care sunt utilizate împreună cu elemente mai avansate, orientate către învățarea automată, cum ar fi logica fuzzy, rețelele neuronale, statisticile bayesiene și altele asemenea. Înțelegem, de asemenea, abordările care utilizează algoritmi de învățare automată pură aplicați exclusiv pe diferite valori și indicatori ai proiectului (metrici).

O abordare neuro-fuzzy combinată cu SEER-SEM este folosită în [DHC15] pentru a obține valori mai scăzute ale MMRE pe patru studii de caz care constau în date specifice COCOMO. Valorile MMRE obținute utilizând abordarea clasică SEER-SEM se situează între 42.05% și 84.39%. Utilizând accesoriul Neuro-Fuzzy, acestea se situează între 29.01% și 69.05%, ceea ce reprezintă o îmbunătățire semnificativă.

Într-o analiză a literaturii modelelor de învățare automată aplicate la problema de față [WLL⁺12], Wen et al. arată că valorile MMRE fluctuează foarte mult între diferite proiecte, precum și algoritmi de învățare diferiți. De exemplu, pentru raționamentul bazat pe caz, sondajul a constatat experimente cu valori MMRE între 13.55% și 143%. Au fost găsite game similare pentru rețelele neuronale artificiale, arborii de decizie, rețelele Bayesian, regresia cu vector suport și procesele Gaussian. Autorii recomandă să se realizeze mai multe cercetări empirice privind fezabilitatea învățării automate pentru estimarea efortului în ingineria software, cu accent pe aplicabilitatea industrială, care se dovedește a fi scăzută în cercetarea studiată.

2.2.2 Modele de învățare automate folosind procesare de text

Conform celor găsite de noi, teza lui Alhad în [Sap12] este singura altă cercetare care abordează problema estimării efortului prin introducerea descrierilor sarcinilor direct în modelele de învățare automată. Aceasta utilizează o abordare ce folosește cuvinte cheie extrase din *story cards*-urile Agile, pe care apoi le furnizează mai multor modele de învățare, cum ar fi Naive Bayes, J48, *random forests*. Experimentele se desfășoară atât cu estimările Planning Poker incluse în partea reală de învățare a modelului, cât și fără. Autorul raportează estimări de 106.81% MMRE pentru estimările Planning Poker și 92.32% MMRE utilizând J48 (care depășește celelalte modele) cu estimările Planning Poker excluse din etapa de învățare. Includerea estimărilor Planning Poker conduce la rezultate ușor mai bune, dar nu sunt suficiente pentru a nu contrazice scopul unei abordări automate.

Abordarea clasifică instanțele în clase reprezentând numere Fibonacci, în același mod în care sunt furnizate estimările Planning Poker.

Deoarece folosim și o abordare similară bazată pe text, considerăm că [Sap12] este cea mai relevantă cercetare cu care ne putem compara, deși seturile noastre de date sunt diferite și nici nu sunt disponibile public.

Bibliografie

- [AR04] B.M Auerbach and C.B. Ruff. Human body mass estimation: a comparison of “morphometric” and “mechanical” methods. *American Journal of Physical Anthropology*, 125(4):331–342, Dec 2004.
- [ATS15] Ishani Arora, Vivek Tatarwal, and Anju Saha. Open issues in software defect prediction. *Procedia Computer Science*, 46:906 – 912, 2015.
- [B.00] Ruff Christopher B. Body mass prediction from skeletal frame size in elite athletes. *American Journal of Physical Anthropology*, 113(4):507—517, Dec 2000.
- [BCH⁺00] Barry W. Boehm, Clark, Horowitz, Brown, Reifer, Chulani, Ray Madachy, and Bert Steece. *Software Cost Estimation with Cocomo II with Cdrom*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 2000.
- [Boe01] Barry W. Boehm. *Software Engineering Economics*, pages 99–150. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.
- [BR14] CARLES BOIX and FRANCES ROSENBLUTH. Bones of contention: The political economy of height inequality. *American Political Science Review*, 108:1–22, 2 2014.
- [CCMI16] Istvan-Gergely Czibula, Gabriela Czibula, Zsuzsanna-Edit Marian, and Vlad-Sebastian Ionescu. A novel approach using fuzzy self-organizing maps for detecting software faults. *Studies in Informatics and Control*, 25(2):207–216, 2016.
- [CD13] Christian Crowder and Victoria M. Dominguez. Estimation of age at death using cortical bone histomorphometry. Research Report, 2013. Funded by the U.S. Department of Justice, Office of Justice Programs, National Institute of Justice.
- [CIMM16] Gabriela Czibula, Vlad-Sebastian Ionescu, Diana-Lucia Miholca, and Ioan-Gabriel Mircea. Machine learning-based approaches for predicting stature from archaeological skeletal remains using long bone lengths. *Journal of Archaeological Science*, 69:85–99, 2016.
- [Coh05] Mike Cohn. *Agile Estimating and Planning*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2005.
- [DHC15] Wei Lin Du, Danny Ho, and Luiz Fernando Capretz. A neuro-fuzzy model with SEER-SEM for software effort estimation. *CoRR*, abs/1508.00032, 2015.
- [ICT16] Vlad-Sebastian Ionescu, Gabriela Czibula, and Mihai Teletin. Supervised learning techniques for body mass estimation in bioarchaeology. In *IEEE 7th International Workshop on Soft Computing Applications*. accepted, 2016.
- [IDC17] Vlad-Sebastian Ionescu, Horia Demian, and Istvan-Gergely Czibula. Natural language processing and machine learning methods for software development effort estimation. *Studies in Informatics and Control*, 26(2):219–228, 2017.

- [IMMC15] Vlad-Sebastian Ionescu, Ioan-Gabriel Mircea, Diana-Lucia Miholca, and Gabriela Czibula. Novel instance based learning approaches for stature estimation in archaeology. In *2015 IEEE International Conference on OIntelligent Computer Communication and Processing*, pages 309–316. IEEE Computer Society, 2015.
- [Ion15a] Vlad-Sebastian Ionescu. New supervised learning approaches for sex identification in archaeology. In *International Virtual Research Conference In Technical Disciplines*, pages 56–64. EDIS - Publishing Institution of the University of Zilina, 2015.
- [Ion15b] Vlad-Sebastian Ionescu. Support vector regression methods for height estimation in archaeology. *Studia Universitatis Babes-Bolyai Series Informatica*, LX(2):70–82, 2015.
- [Ion17] Vlad-Sebastian Ionescu. An approach to software development effort estimation using machine learning. In *2017 IEEE International Conference on OIntelligent Computer Communication and Processing*, page To be published. IEEE Computer Society, 2017.
- [ITV16] Vlad-Sebastian Ionescu, Mihai Teletin, and Estera-Maria Voiculescu. Machine learning techniques for age at death estimation from long bone lengths. In *IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI 2016)*, pages 457–462. IEEE Hungary Section, 2016.
- [LI14] Andrei Lorian and Vlad Ionescu. Some differential superordinations using ruscheweyh derivative and generalized salagean operator. *Journal of Computational Analysis and Applications*, 17(3):437–444, 2014.
- [MCM16] Zsuzsanna-Edit Marian, Istvan-Gergely Czibula, Ioan-Gabriel Mircea, and Vlad-Sebastian Ionescu. A study on software defect prediction using fuzzy decision trees. *Studia Universitatis Babes-Bolyai Series Informatica*, LXI(2):5–20, 2016.
- [Moo08] Megan K. Moore. *Body Mass Estimation from the Human Skeleton*. PhD thesis, The University of Tennessee, Knoxville, 2008.
- [OOLI16] Georgia Irina Oros, Gheorghe Oros, Alina Alb Lupaş, and Vlad Ionescu. Differential subordinations obtained by using a generalization of marx-strohhacker theorem. *Journal of Computational Analysis and Applications*, 20(1):135–139, 2016.
- [PM03] Lawrence H. Putnam and Ware Myers. *Five Core Metrics: Intelligence Behind Successful Software Management*. Dorset House Publishing Co., Inc., New York, NY, USA, 2003.
- [RHN⁺12] C.B. Ruff, B.M. Holt, M. Niskanen, V. Sladék, M. Berner, E. Garofalo, H.M. Garvin, M. Hora, H. Maijanen, S. Niinimäki, K. Salo, E. Schuplerová, and D. Tompkins. Stature and body mass estimation from skeletal remains in the european holocene. *American Journal of Physical Anthropology*, 182(4):601–617, Aug 2012.
- [RR06] M. Raxter, M.H.and Auerbach and B. Ruff. Revision of the fully technique for estimating statures. *Am J Phys Anthropol*, 130:374–384, 2006.
- [Sap12] Alhad Vinayak Sapre. Feasibility of automated estimation of software development effort in agile environments. Master’s thesis, The Ohio State University, 2012.

- [WLL⁺12] Jianfeng Wen, Shixian Li, Zhiyong Lin, Yong Hu, and Changqin Huang. Systematic literature review of machine learning based software development effort estimation models. *Inf. Softw. Technol.*, 54(1):41–59, January 2012.