

BABEŞ-BOLYAI UNIVERSITY  
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

# Contributions to solving real world problems using machine learning models

Abstract of the Ph.D. Thesis

Ph.D. student: Ionescu Vlad-Sebastian  
Scientific supervisor: Prof. Dr. Czibula Gabriela

2017

**Keywords:** *machine learning, archaeology, software engineering, support vector machines.*

# Contents of the abstract

<b>Contents of the thesis</b>	<b>2</b>
<b>List of publications</b>	<b>4</b>
<b>Introduction</b>	<b>6</b>
<b>1 Archaeology</b>	<b>7</b>
1.1 Predicting stature from archaeological skeletal remains using long bone lengths	7
1.2 Body mass estimation in bioarchaeology . . . . .	8
1.3 Age at death estimation from long bone lengths . . . . .	8
<b>2 Software engineering</b>	<b>10</b>
2.1 Software defect prediction . . . . .	10
2.2 Software development effort estimation . . . . .	11
2.2.1 Machine learning models using software metrics . . . . .	11
2.2.2 Machine learning models using text processing . . . . .	12

# Contents of the thesis

<b>Keywords</b>	<b>7</b>
<b>Acknowledgements</b>	<b>8</b>
<b>List of publications</b>	<b>9</b>
<b>Introduction</b>	<b>11</b>
<b>1 Background for Archaeological problems</b>	<b>14</b>
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
<b>2 Contributions to machine learning models for archaeology</b>	<b>21</b>
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
<b>3 Background for Software Engineering problems</b>	<b>63</b>
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
<b>4 Contributions to machine learning models for software engineering</b>	<b>71</b>
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
<b>Conclusions</b>		<b>111</b>

# List of publications

All rankings are listed according to the 2014 classification of journals<sup>1</sup> and conferences<sup>2</sup> in Computer Science and the associated web service<sup>3</sup>.

## Publications in ISI Web of Knowledge

### Publications in 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**).  
**Rank A, 4 points.**
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**).  
**Rank C, 2 points.**
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**).  
**Rank C, 1 points.**
4. [LI14] Georgia Irina Oros, Gheorghe Oros, Alina Alb Lupa 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**).  
**Rank C, 1 points.**
5. [OOLI16] Andrei Lorian and **Vlad Ionescu** Some differential superordinations using Ruscheweyh derivative and generalized Slgean operator. *Journal of Computational Analysis and Applications*, volume 20(1), pp. 437–444, 2014, (**2016 IF = 0.609**).  
**Rank C, 2 points.**

---

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

<sup>2</sup>[http://informatica-universitaria.ro/getpfile/16/CORE2013\\_Exported.xlsx](http://informatica-universitaria.ro/getpfile/16/CORE2013_Exported.xlsx)

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

### Publications in 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**).  
**Rank C, 2 points.**
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.  
**Rank C, 2 points.**
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.  
**Rank C, 2 points.**

### Publications in international journals and conferences

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**).  
**Rank C, 1 point.**
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**).  
**Rank D, 1 point.**
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**).  
**Rank D, 1 point.**
4. [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*, volume LXI(2), pp. 5–20, 2016 (**indexed Mathematical Reviews**).  
**Rank D, 0 points.**

Publications score: **19 points.**

# Introduction

This Ph.D. thesis consists of research on applying computational intelligence techniques for solving real world problems in the fields of **Archaeology** and **Software Engineering**. All of the original research conducted was done under the supervision of Prof. Dr. Gabriela Czibula.

The two fields were chosen due to the existence of several difficult practical problems in both that we hypothesized would be of significant importance to researchers in archaeology and software engineering and that could be approached under a unified machine learning theme. Our hypothesis proved to be correct: no similar approaches exist for the chosen problems, the results obtained are better and useful to researchers in those fields and our approaches have a central machine learning element.

For **archaeology**, the focus was on the application of several machine learning models – Genetic Algorithms (GA), Support Vector Regression (SVR) and Locally Weighted Regression (LWR) – on three important problems from the field of archaeology: *stature prediction*, *weight (body mass) estimation* and *age at death estimation* using bone lengths. The models we have chosen are known to provide very good results on similar regression problems, and the problems we have decided to approach are of significant interest in the archaeological literature, due to the information that a good solution to them can provide to researchers in the field. Moreover, according to our knowledge, machine learning was not used on the approached archaeological problems until now.

Each of our learning models has been applied for all of the researched archaeological tasks, resulting in a unified approach to the field of archaeology through machine learning. Our approaches are novel in relation to the archaeological field and have been published in journal articles and conferences [[CIMM16](#), [ITV16](#), [ICT16](#), [IMMC15](#), [Ion15b](#)].

Concerning **Software Engineering**, we have focused on two important problems from the Search-based Software Engineering literature: *Software Defect Prediction* (SDP) and *Software Development Effort Estimation* (SDEE). We have proposed several machine learning models for solving the above mentioned problems and we have performed experiments on various data sets relating to these problems using the Fuzzy Self Organizing Maps (FSOM), Fuzzy Decision Trees (FDT) and Support Vector Regression (SVR) machine learning models. Our models are either novel in and of themselves or have not been applied to these problems before in the way that we have. We have obtained very good results with all three approaches, surpassing most approaches from the literature and introducing new research directions for some important problems in the field of Software Engineering.

Our novel solutions for these problems result in a unified machine learning approach for various Software Engineering tasks, which can help developers and project managers alike. Our approaches are novel in relation to the Software Engineering field and have been published in the papers [[CCMI16](#), [MCM16](#), [Ion17](#), [IDC17](#)].

# Chapter 1

## Archaeology

In this chapter we present the archaeological background knowledge regarding stature, weight and age required for our approaches. This background knowledge has been collected in order to facilitate the original research published in [CIMM16, ITV16, ICT16].

### 1.1 Predicting stature from archaeological skeletal remains using long bone lengths

From the perspective of forensic anthropology and archaeology, predicting the stature of an individual based on osteological information is fundamental. The classical mathematical approaches to stature estimation focus on the use of regression methods [SS97] based on statistical analysis of the data

Estimating stature is important in bioarchaeological and forensic research, firstly because stature is a standard biological attribute together with age and weight. It also enables researchers to assess sexual dimorphism or the body size of the past population under study [RR06]. Moreover, stature is an important indicator of individual's physical growth and development within its social and natural environment [BR14]. Despite the individual's natural genetic potential for physical growth, it is the society that nurtures its members through nutrition, hygiene, physical education etc., to reach their potential [BR14].

Stature estimation is a common topic in anthropological analysis, the generated results having wide applications for making biocultural assessments with regard to archaeological populations. Drawing upon methods and theories from human behavioural ecology, physical and cultural anthropology, sociology, and economy, scholars have used stature as a quality-of-life indicator for inferring the complex relationships between skeletal development and ecology, diet, nutrition, genetics, and physical activity [LW10, BR14].

Together with porotic hyperostosis, cribra orbitalia, and dental enamel hypoplasia, living height is used as a measure of health in bioarchaeological studies [Aue11, Mor09, PDS<sup>+</sup>14], allowing for inferences about subsistence strategies or social inequality [BR14]. The *mean height* of a population is considered in [Mor09] to be a marker of its nutritional and health status. In [PDS<sup>+</sup>14], Pietrusewsky et al. classified stature as an indicator of health, as due to non-specific systemic stress through the growth.

Along with the development of computational intelligence and machine learning, it is natural to research the possibility of building computer programs that do not have predefined algorithms for generating predictions, but instead learn from the available data and adapt their models according to the new data samples that are being processed.



## 1.2 Body mass estimation in bioarchaeology

Estimating the body mass from human skeletal remains represents a problem of major importance in paleontological and archaeological research. There is an agreement in the bioarchaeological literature that postcranial measurements are directly related to the body size and provide the most accurate estimates [B.00].

Estimation of the body mass from human skeletons represents a challenge in forensic death investigations concerning unidentified remains [Moo08]. A major problem in research related to body mass estimation is caused by a lack of publicly available benchmarks. The existing methods for body mass estimations from the skeletons are: *mechanical*, *morphometric* [AR04] and a combination of biomechanical and morphometric methods [Moo08]. The morphometric methods consist of directly reconstructing the body size from the skeletal elements, while the mechanical methods provide functional associations between skeletal elements and body mass [AR04].

Body mass estimation is a very important problem for modern archeology. It can provide certain knowledge about past populations, such as indicators for [RHN<sup>+</sup>12]: the past population's health, the effects of different environmental factors on past populations (e.g. subsistence strategy, climatic factors), social aspects etc. The ability to obtain accurate body mass estimations from skeletons is also essential in forensic death investigations concerning unidentified skeletal remains [Moo08]. Consequently, it is essential for bioarchaeologists to develop and use body mass estimation methods that are as accurate as possible.

However, designing an accurate method for solving this problem remains a great challenge, because there are many factors which should be taken into account [AR04]. Some decisions that are to be taken in this process [RHN<sup>+</sup>12]: which are the most relevant skeletal measurements to use, which is the appropriate statistical approach to apply, which should be the skeletal sample to use etc.

## 1.3 Age at death estimation from long bone lengths

Age at death estimation is a problem that asks to find a good estimate for the age at death of some human biological remains. These remains can be bones, teeth, a well preserved body etc. In our case, we consider long bone lengths and are interested in finding, using machine learning, a mathematical function of these lengths that gives us the age at death of the instance whose bone lengths are input to the function.

Estimating age at death is essential for determining population demographics and performing individual analysis of human remains. For children and young adults, the problem is easier due to the fact that growth and biological development occur during these stages [CD13].

The age at death estimation problem is a complicated one for adult skeletons. This is due to the fact that age indicators are biologically variable and that certain features respond differently to environmental factors over the course of someone's life, so a lot more individual differences that do not help to pinpoint an exact age (noise) accumulates for older individuals [CD13].

Because the problem is so complex, there are no exact algorithms for solving it. Simple regression formulas that take one or a few bones into consideration exist, but they are usually created for each particular data set, and it is unclear if they would work well or not on other data. By using machine learning, we can come up with models that are easy to retrain on new data sets if needed. By using validation methods, we can estimate how well our algorithms will perform on unseen data of the same kind.

We deal in the thesis with age at death estimation in multiple age categories, but we advise the reader to interpret our results relative to what is known in the literature as being

feasible to achieve for each age group in particular.

## Chapter 2

# Software engineering

In this chapter we present the Software Engineering background knowledge regarding Software Defect Prediction and Software Development Effort Estimation. The literature survey conducted in this chapter has made possible the original research published in [CCMI16, MCM16, IDC17, Ion17].

### 2.1 Software defect prediction

*Software defect prediction* represents the activity of identifying software modules which are likely to develop errors in a forthcoming version of a software system, being of major importance for software testing and for assuring the software quality as well. The methods for detecting faulty software entities are useful for suggesting to developers the software modules that should be rigorously tested. These software entities can be software components, modules, packages, classes, methods, functions or other software artifacts.

The software maintenance process represents a major part of a software life cycle, requiring a large software engineering effort. The software engineering literature reveals that understanding the software represents about half of the amount of effort allocated to the maintenance activity. Fixing defects represents one of the main software maintenance activities, also being referred to as *corrective maintenance* [KMFO01].

For increasing the efficiency of the software defect-fixing process, defect prediction models are useful for anticipating locations in a software system where future defects may appear. Identifying software defects is difficult, mainly for complex software projects. The main difficulty related to building supervised defect predictors is the fact that the defects in software projects are much less than the non-defective entities and thus, the training data is highly imbalanced [ATS15].

Another important problem related to software faults identification is *software defect detection*, which represents the process of identifying software modules that contain errors and it contributes to increasing the effectiveness of the quality assurance process. Fault detection methods would be helpful for suggesting the developers which software modules should be focused on during testing, particularly when, from lack of time, the modules can not be systematically tested.

Code review is an activity frequently used in agile development processes for maintaining the quality of the software system. During code review, an experienced programmer reviews the source code in order to identify vulnerabilities, security problems and other problems overlooked by the initial implementer. Since code review is a time consuming and costly activity, detecting software defects can be used to guide the process of code reviewing by identifying sections in the code that would likely be flagged for revision during a code review session, due to various issues with that code.

## 2.2 Software development effort estimation

*Software development effort estimation* (SDEE) represents the action of estimating the time it will take for each part of a software system to be completed during the development phase of the product. Such estimations are done either at a management level or at the level of the programmer(s) responsible for the actions that will need to be performed. Accurate estimates are important in order to properly plan the development process and allocate human resources accordingly.

Multiple estimation methodologies and protocols exist, including software-aided ones. Two popular ones are simple ad-hoc estimations, in which a programmer provides a time estimate based on the task description and his or her experience with what is required, and group methods, such as planning poker, which are more involved methodologies that require a consensus from a team of programmers [Coh05]. Both of these are considered *expert estimations*, because experts in the field are the ones providing the estimates.

Group methods have the advantage of being more accurate and, in the case of planning poker, allow refinement of estimates over time. The advantage of more complex methods is a better estimate than ad-hoc methods in general, but they take a lot more time that could be used for actual development. As far as we know, there are no studies that analyze this trade-off and whether or not it is a worthwhile endeavour to employ more complex estimation approaches.

According to one study, between 60% and 80% of projects have actual effort values between 30% and 89% of the estimated values [MJ03]. While more recent data is not available, judging from our experience, similar percentages would still be the case.

Software-aided methodologies consist of a computer system that provides an estimate, or helps programmers arrive at one, by algorithmically analysing parts of the requirement or the software under development. The most common methods used nowadays are *parametric estimation models*, such as COCOMO and the Putnam model.

COCOMO uses three formulas that give *effort applied* in man-months, *development time* in months and *people required*. It uses predefined constants for each of three types of projects. It does not account for developer experience, available hardware resources and some other factors [Boe01, BCH<sup>+</sup>00]. The Putnam model uses a formula for effort (expressed in person-years) that includes the size of the product being developed, the productivity of the company and the total scheduled time for the project [PM03].

Most of the existing computer-aided methods rely on software metrics of questionable relevance and on other human estimates, such as complexity and productivity. This makes them highly prone to large errors and far removed from the intuitive approach that most programmers use when providing estimates.

### 2.2.1 Machine learning models using software metrics

By machine learning models using software metrics, we understand frameworks such as COCOMO that are used together with more advanced, machine learning-oriented elements, such as fuzzy logic, neural networks, Bayesian statistics and the like. We also understand approaches that use pure machine learning algorithms applied exclusively on various project metrics and indicators.

A Neuro-Fuzzy approach is used by Du et al. in conjunction with SEER-SEM in [DHC15] in order to obtain lower MMRE values on four case studies consisting of COCOMO-specific data. The obtained MMRE values using the classical SEER-SEM approach are between 42.05% and 84.39%. Using the Neuro-Fuzzy enhancement, they are between 29.01% and 69.05%, which is a significant improvement.

Han et al. apply in [HJLZ15] a larger set of machine learning algorithms: linear regression, neural networks, M5P tree learning, Sequential Minimal Optimization, Gaussian Process,

Least Median Squares and REPTree. The study is conducted on 59 projects having between 6 and 28 developers and between 3 and 320 KLOC. The obtained MMRE values are between 87.5% for the Linear Regression approach and 95.1% for the Gaussian Process model.

Bayesian networks, Regression trees, Backward elimination and Stepwise selection are applied on various metrics from two software project data sets by van Koten and Grayin [vKG06]. The best obtained MMRE is 97.2% on one of the projects, using Bayesian networks, and 0.392%, using Stepwise selection, on the other project.

In a literature review of machine learning models applied to the SDEE problem [WLL<sup>+</sup>12], Wen et al. show that MMRE values fluctuate a lot between different projects as well as different learning algorithms. For example, for Case Based Reasoning, the survey found experiments with MMRE values between 13.55% and 143%. Similar ranges were found for Artificial Neural Networks, Decision Trees, Bayesian Networks, Support Vector Regression and Gaussian Processes. The authors recommend that more empirical research be performed regarding the feasibility of ML for SDEE, with a focus on industry applicability, which is found to be low in the surveyed research.

In [UMWB14], Usman et al. obtain MMRE values between 66% and 90% using linear regression. Using Radial Basis Function networks, MMRE values between 6% and 90% are obtained.

According to our literature review, machine learning models applied on software metrics provide better estimates than pure parametric models. The MMRE values are also less spread out between different data sets, which makes machine learning models more reliable and predictable from an accuracy point of view.

However, a remaining drawback of these approaches is the need for project software metrics, which are not always available or would take substantial effort to collect properly. Sometimes, various parameters must still be inputted by the developers, which takes about as much time as it would take developers to provide their own estimates.

### 2.2.2 Machine learning models using text processing

To the best of our knowledge, the thesis by Alhad in [Sap12] is the only other research that approaches the SDEE problem by inputting task descriptions directly to ML learning pipelines. It uses a bag of words approach on keywords extracted from Agile story cards, which it then feeds to multiple learning models, such as Naive Bayes, J48, Random Forests and Logistic Model Trees. Experiments are conducted both with the Planning Poker estimates included in the actual learning part of the pipeline and without. The author reports 106.81% MMRE for Planning Poker estimates, and 92.32% MMRE using J48 (which outperforms the other models) with the Planning Poker estimates excluded from the learning stage. Including the Planning Poker estimates leads to slightly better results, although not enough so as to not defeat the purpose of an automatic approach.

The approach classifies instances into classes representing Fibonacci numbers, in the same way that Planning Poker estimates are provided.

Because we also use a similar text-based approach, we consider [Sap12] to be the most relevant related work to compare ourselves with, although our data sets are different and neither are publicly available.

# Bibliography

- [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.
- [Aue11] Benjamin M. Auerbach. Methods for estimating missing human skeletal element osteometric dimensions employed in the revised fully technique for estimating stature. *American Journal of Physical Anthropology*, 145:67–80, 2011.
- [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<sup>+</sup>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.
- [HJLZ15] WanJiang Han, LiXin Jiang, TianBo Lu, and XiaoYan Zhang. Comparison of machine learning algorithms for software project time prediction. *International Journal of Multimedia and Ubiquitous Engineering*, 10(9):1–8, 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.
- [KMFO01] Mira Kajko-Mattsson, Stefan Forssander, and Ulf Olsson. Corrective maintenance maturity model (cm3): Maintainer’s education and training. In *Proceedings of the 23rd International Conference on Software Engineering, ICSE ’01*, pages 610–619, Washington, DC, USA, 2001. IEEE Computer Society.
- [LI14] Andrei Loriana and Vlad Ionescu. Some differential superordinations using ruscheweyh derivative and generalized salagean operator. *Journal of Computational Analysis and Applications*, 17(3):437–444, 2014.
- [LW10] C.S. Larsen and L.P. Walker. *Bioarchaeology: Health, Lifestyle, and Society in Recent Human Evolution, A Companion to Biological Anthropology*. Wiley-Blackwell, 2010.
- [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.
- [MJ03] Kjetil Molkken and Magne Jrgensen. A review of surveys on software effort estimation. In *Proceedings of the 2003 International Symposium on Empirical Software Engineering, ISESE ’03*, pages 223–, Washington, DC, USA, 2003. IEEE Computer Society.

- [Moo08] Megan K. Moore. *Body Mass Estimation from the Human Skeleton*. PhD thesis, The University of Tennessee, Knoxville, 2008.
- [Mor09] Alexander Moradi. Towards an objective account of nutrition and health in colonial kenya: A study of stature in african army recruits and civilians, 1880-1980. *The Journal of Economic History*, 69:719–754, 9 2009.
- [OOLI16] Georgia Irina Oros, Gheorghe Oros, Alina Alb Lupa, 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.
- [PDS<sup>+</sup>14] M. Pietruszewsky, M. Douglas, M.K. Swift, R.A. Harper, and M.A. Fleming. Health in ancient mariana islanders: a bioarchaeological perspective. *Journal of Island and Coastal Archaeology*, 9:319–340, 2014.
- [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<sup>+</sup>12] C.B. Ruff, B.M. Holt, M. Niskanen, V. Sladk, M. Berner, E. Garofalo, H.M. Garvin, M. Hora, H. Maijanen, S. Niinimki, K. Salo, E. Schuplerov, 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.
- [SS97] A. Sen and M. Srivastava. *Regression Analysis: Theory, Methods, and Applications*. Springer Texts in Statistics. Springer New York, 1997.
- [UMWB14] Muhammad Usman, Emilia Mendes, Francila Weidt, and Ricardo Britto. Effort estimation in agile software development: A systematic literature review. In *Proceedings of the 10th International Conference on Predictive Models in Software Engineering*, PROMISE ’14, pages 82–91, New York, NY, USA, 2014. ACM.
- [vKG06] C. van Koten and A. R. Gray. An application of bayesian network for predicting object-oriented software maintainability. *Inf. Softw. Technol.*, 48(1):59–67, January 2006.
- [WLL<sup>+</sup>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.