

UNIVERSITATEA BABES BOLYAI, CLUJ NAPOCA, ROMANIA  
FACULTATEA DE MATEMATICA SI INFORMATICA

# Reconstituiri istorice

– MIRPR –

## Membrii echipei

Andrei-Danut Blagoi, Informatica romana, grupa 231

Andreea Bolonyi, Informatica romana, grupa 231

Stefan-Nicolae Parvanescu, Informatica engleza, grupa 936

Oana-Alexandra Sidorencu, Informatica romana, grupa 236

2021-2022

## **Rezumat**

Proiectul propus este menit sa vina in ajutorul persoanelor pasionate de istorie si arheologie care isi doresc informatii plastice despre anumite date introduse. Utilizatorii aplicatiei au posibilitatea sa exploreze niste date numerice pe care le gasesc, reusind sa cunoasca care este sexul sau varsta unui os pe care acestia il studiaza si, de asemenea, utilizatorii au posibilitatea sa perceapa informatia si intr-un mod vizual.

# Cuprins

<b>1</b>	<b>Introducere</b>	<b>1</b>
1.1	Motivarea temei . . . . .	1
1.2	Tehnologii . . . . .	1
1.3	Setul de date . . . . .	1
<b>2</b>	<b>Problema stiintifica</b>	<b>5</b>
2.1	Definitia problemei . . . . .	5
2.2	Algoritm machine learning - arbore de decizie . . . . .	5
2.3	Rezultate arbore de decizie . . . . .	6
<b>3</b>	<b>Lucrari stiintifice</b>	<b>9</b>

# Capitolul 1

## Introducere

### 1.1 Motivarea temei

Problema abordata in acest proiect este de interes pentru persoanele care lucreaza in acest domeniu, studenti ce vor ajunge angajati sau persoane care sunt doar pasionate. Aplicatia dezvoltata este usor de folosit, intuitiva si menita sa ofere doar ajutor, nu probleme utilizatorului.

Pasionatii folosind aplicatia noastra vor putea sa aiba o idee mult mai aprofundata in legatura cu subiectul pe care il studiaza, nu doar sa se bazeze pe niste cifre pe care le vad. De asemenea, o functionalitate pe care dorim sa o oferim utilizatorului este posibilitatea de a vedea 3D aceste informatii pe care le introduce.

### 1.2 Tehnologii

Dezvoltarea aplicatiei se bazeaza pe limbajul Python pentru partea de backend si PyQt pentru partea de frontend. Am decis sa folosim Python datorita usurintei cu care se poate scrie codul si multitudinea de solutii pe care le putem gasi, fie ca este vorba de o problema obisnuita pentru un programator, fie ca este vorba de biblioteci din domeniul inteligentei artificiale pe care le putem folosi pentru rezolvarea subiectului (spre exemplu Scikit-learn, TensorFlow, Theano).

### 1.3 Setul de date

<b>TML</b>	Tibia Maximum Length
<b>TPB</b>	Tibia Plateau Mediolateral (Bicondylar) Breadth
<b>TMLD</b>	Tibia 50% Diaphyseal Mediolateral Diameter
<b>TAPD</b>	Tibia 50% Diaphyseal Anteroposterior Diameter

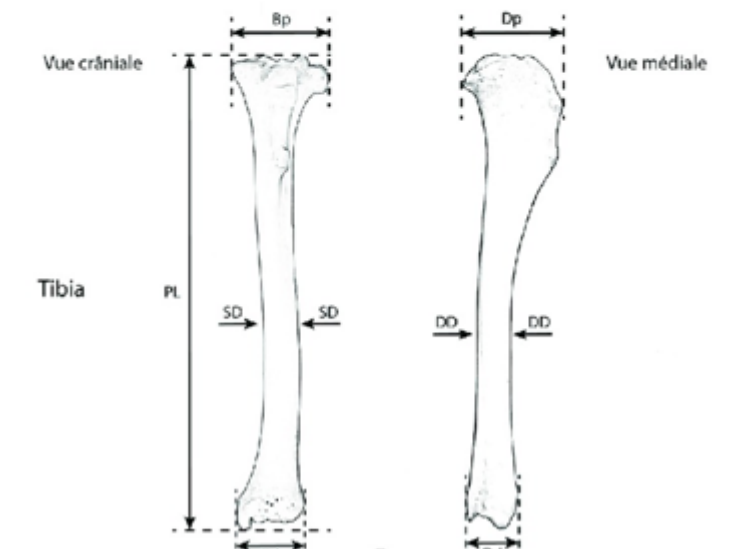


Figura 1.1: Caracteristici tibie.

<b>HML</b>	Humerus Maximum Length -b (33,4)M (30,7)F
<b>HEB</b>	Humerus Epicondylar Breadth -c
<b>HHd</b>	Humerus Head Diameter -g
<b>HMLD</b>	Humerus 50% Diaphyseal Mediolateral Diameter -a
<b>HAPD</b>	Humerus 50% Diaphyseal Anteroposterior Diameter -a

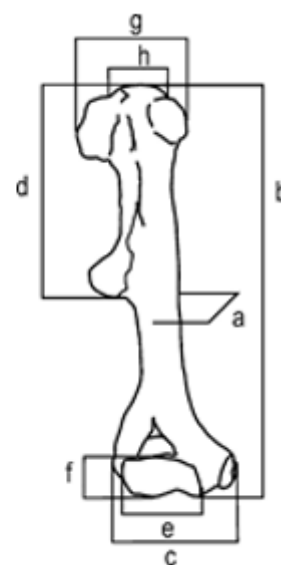


Figura 1.2: Caracteristici humerus.

<b>RML</b>	Radius Maximum Length
<b>RMLD</b>	Radius 50% <u>Diaphyseal</u> Mediolateral Diameter (MAX) -a
<b>RAPD</b>	Radius 50% <u>Diaphyseal</u> Anteroposterior Diameter (MIN) -a

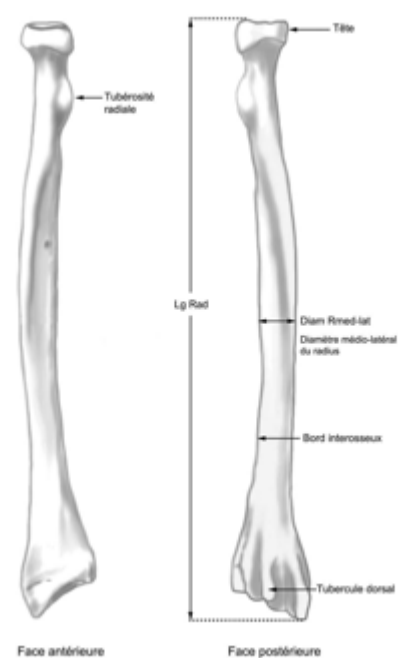


Figura 1.3: Caracteristici radius.

<b>FML</b>	Femur Maximum Length –FML
<b>FBL</b>	Femur <u>Bicondylar</u> Length - FBL
<b>FEB</b>	Femur <u>Epicondylar</u> Mediolateral Breadth – FEB
<b>FAB</b>	
<b>FHD</b>	
<b>FMLD</b>	Femur 50% <u>Diaphyseal</u> Mediolateral Diameter
<b>FAPD</b>	Femur 50% <u>Diaphyseal</u> Anteroposterior Diameter

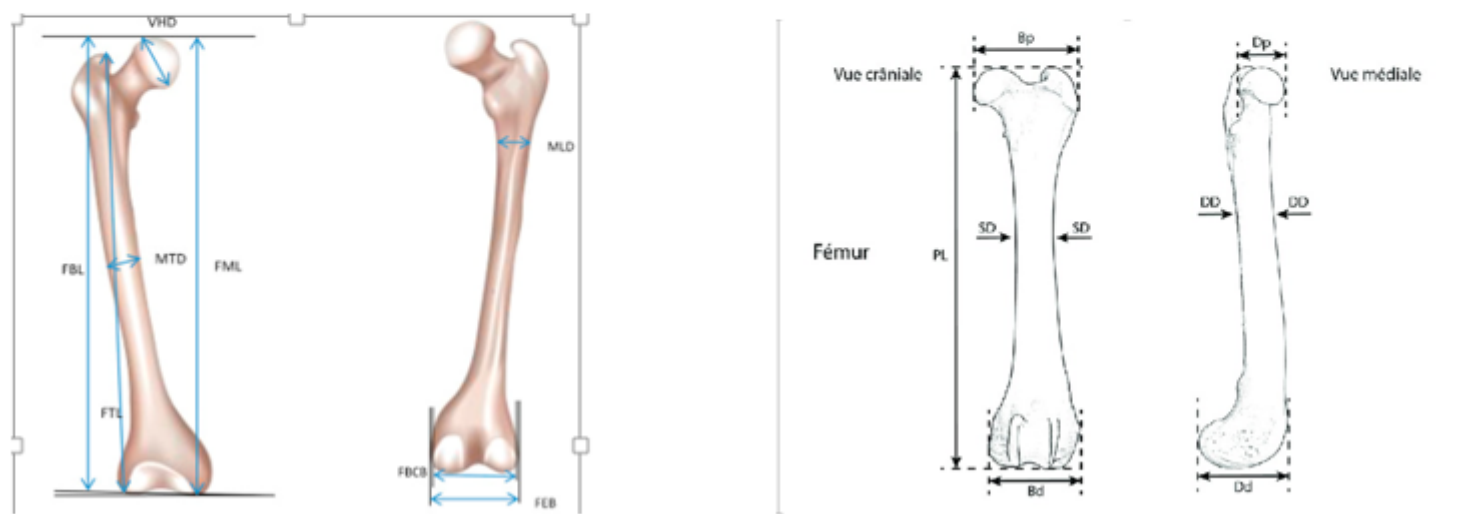


Figura 1.4: Caracteristici femur.



## Capitolul 2

# Problema stiintifica

### 2.1 Definitia problemei

Subiectul abordat tine de domeniul istoriei si al arheologiei pentru a obtine informatii revelante despre obiectele identificate in santierele arheologice. Astfel se doreste o aplicatie care, plecand de la informatii deja studiate de arheologi umani, sa permita vizualizarea 3D a unor "descoperiri deja efectuate" in intregime sau partial, din diferite unghiuri, reliefand anumite detalii. Mai mult, ofera posibilitatea determinarii sexului sau varstei pe baza unor caracteristici numerice ale unui anumit tip de os.

Pentru simplitatea aplicatiei si usurinta folosirii exista o interfata grafica care va permite utilizatorului sa introduca caracteristicile pe baza carora se va stabili rezultatul. Dupa apasarea unui buton de trimitere a datelor, utilizatorul va fi intrebat daca este de acord ca datele introduse sa fie pastrate in baza de date pentru imbunatatirea solutiei. In urma procesarii datelor, utilizatorul va vedea care este sexul sau varsta osului specificat si va avea posibilitatea sa observe si in 3D cum ar arata acesta.

### 2.2 Algoritm machine learning - arbore de decizie

Ca prim algoritm am folosit un arbore de decizie datorita simplitatii de a intelege cum mai exact se iau deciziile pentru a determina clasele din care face parte o instantă; este usor de inteles pentru o persoana din domeniul informaticii, dar si pentru orice alta persoana.

Pentru un arbore de decizie avem un nod radacina, noduri intermediare si noduri finale(frunze). Nodul radacina reprezinta atributul cel mai semnificativ (aceasta alegere se face in functie de niste indici de

care se va discuta ulterior), nodurile intermediare care reprezinta decizii de tip if-else si frunzele care reprezinta clasa din care face parte o instanta. Pentru a determina rezultatul final practic se imparte setul de date in instante care respecta un anumit set de reguli.

Selectarea unui atribut ca fiind nod radacina se poate face folosind mai multi indici. Indexul Gini este varianta care se foloseste de clasa din Sklearn (`DecisionTreeClassifier`) daca nu este specificat alt index. Acesta este o metrica care masoara cat de des un element ales aleator este clasificat gresit; un index Gini mai mic inseamna un atribut care va fi preferat pentru a deveni nod radacina. In implementarea aplicatiei s-a folosit de asemenea un index Gini. Folosind formula de calcul de mai jos avem ca notatii  $D$  instanta cu caracteristicile corespunzatoare si  $P_i$  este probabilitatea ca  $D$  sa apartina clasei de la momentul respectiv in decizie.

## 2.3 Rezultate arbore de decizie

O prima varianta de implementare pentru problema curenta de clasificare a folosit clasa din Sklearn (`DecisionTreeClassifier`) cu indexul Gini pentru a stabili nodul radacina. S-a folosit un set de date de dimensiune 100 care avea distributia descrisa ulterior. De asemenea, s-a creat un arbore in care se poate observa cum un nod intermediar reprezinta o decizie care se poate lua, ea fiind de tipul "daca conditia este adevarata, atunci mergi pe partea stanga a subarborelui, altfel pe partea dreapta; daca avem o frunza atunci ne oprim si returnam clasa corespunzatoare".

Folosind aceasta implementare am observat rezultate bune, avand o acuratete (probabilitatea ca un exemplu sa fie clasificat corect) de aproximativ 65 la suta cu un timp de executie redus pentru antrenarea arborelui. Totusi, un dezavantaj ce a fost remarcat a fost faptul ca arborele este destul de sensibil la modificarea datelor de intrare si o mica eroare de tipar poate da peste cap algoritmul (de exemplu, daca arborele primeste o instanta care are numele caracteristicilor cu alt format fata de cum este dat in structura lui s-ar putea sa aiba probleme la asocierea valorilor cu sensul lor).

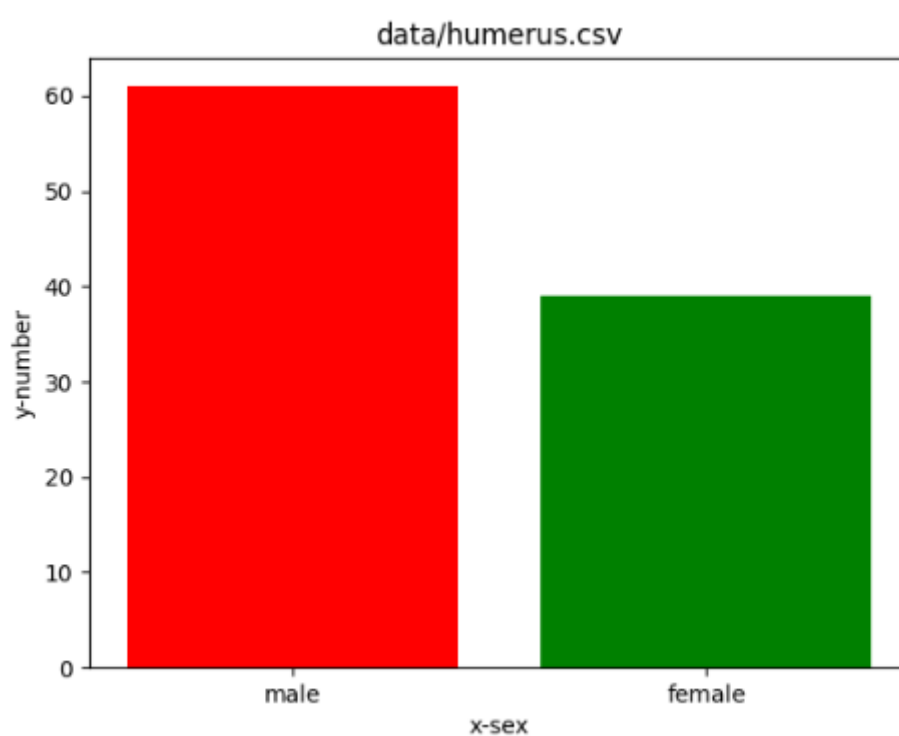


Figura 2.1: Distribuție date.

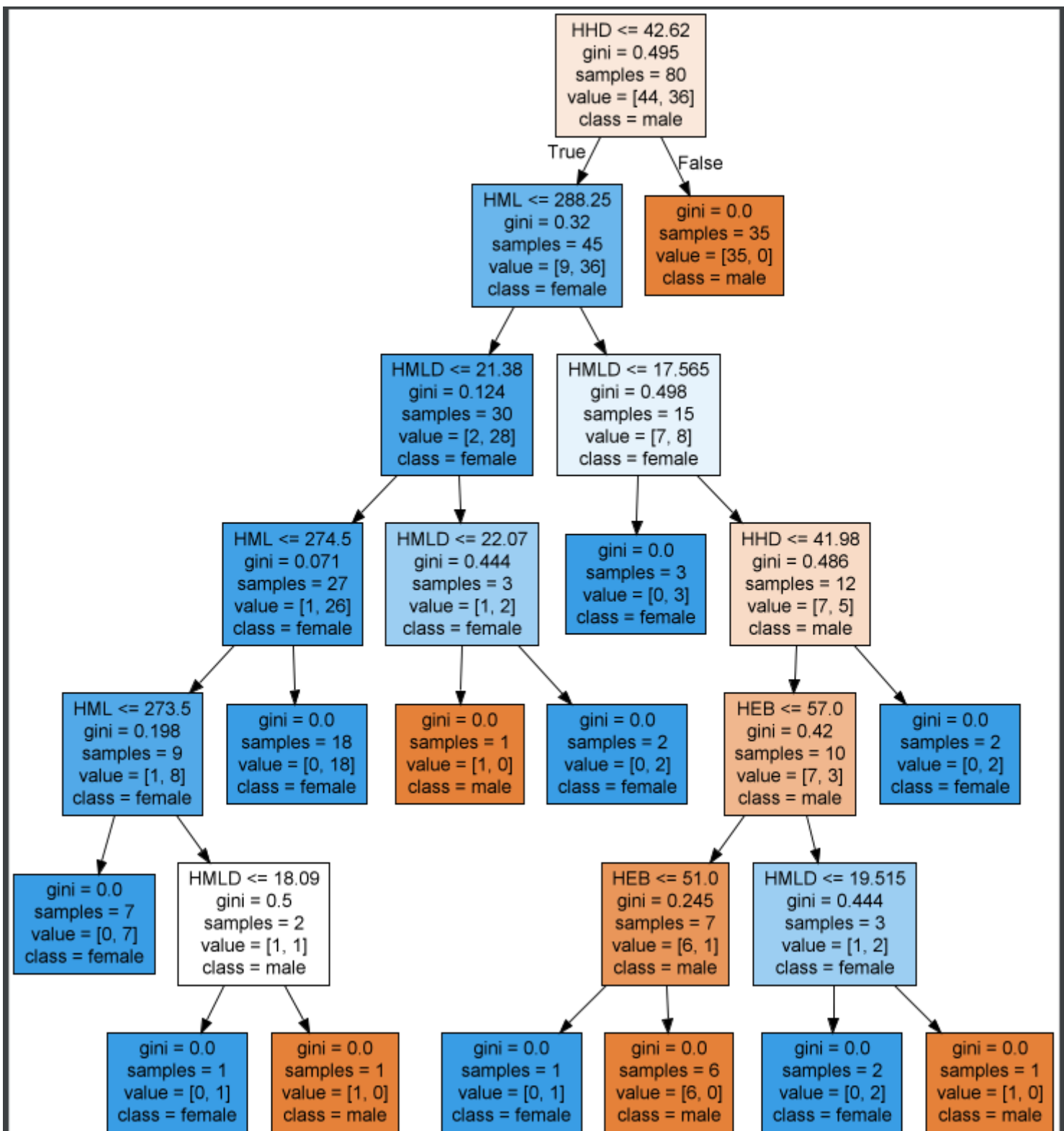


Figura 2.2: Arbore generat.

## Capitolul 3

# Lucrari stiintifice

Punctul de start al proiectului dat a fost constituit de urmatoarele lucrari stiintifice: [2], [4], [5], [3], [1]. Acestea au oferit inspiratia si ajutorul de care am avut nevoie pentru a dezvolta aplicatia.

Pentru implementarea arborelui de decizie si alegerea acestuia in aplicatie s-a folosit lucrarea [3], de unde am retinut mai ales importanta alegerii unui algoritm care sa fie usor de inteles atat pentru programator, cat si pentru un arheolog pentru a verifica corectitudinea deciziilor. De asemenea, pentru alegerea metricii care selecteaza atributul s-au luat in considerare informatiile prezentate in [6].

# Bibliografie

- [1] Gabriela Czibula. *Machine learning-based approaches for predicting stature from archaeological skeletal remains using long bone lengths*. 2016.
- [2] Geertje Klein Goldewijk and Jan Jacobs. *The relation between stature and long bone length in the roman empire*. 2013.
- [3] Gabriela Czibula Ionescu, Vlad-Sebastian and Mihai Teletin. *Supervised Learning Techniques for Body Mass Estimation in Bioarchaeology*. 2016.
- [4] Jan PAM Jacobs Jongman, Willem M. and Geertje M. Klein Goldewijk. *Health and wealth in the Roman Empire*. 2019.
- [5] Diana-Lucia Miholca. *Machine learning based approaches for sex identification in bioarchaeology*. 2016.
- [6] Dr. K. Nirmala R. Aruna devi. *Construction of Decision Tree : Attribute Selection Measures*. 2013.