Proiect PCLP3

Mihai Nan*, George Alexandru Tudor
Departamentul de Calculatoare

9 mai 2024

Notă!

Proiectul poate fi realizat individual sau în echipă de 2 studenți. În cazul în care alegeți să realizați proiectul individual, atunci veți realiza una dintre cele două părți. Dacă alegeți să realizați proiectul în echipă, atunci fiecare student își va alege o parte.

Cuprins

*mihai.nan@upb.ro

1 Introducere

Python a devenit indispensabil în domeniul Data Science și Inteligența Artificială, mai ales în explorarea datelor și dezvoltarea de modele, datorită ecosistemului său bogat de biblioteci specializate. Cu biblioteci precum `numpy`, `pandas`, `matplotlib`, `seaborn`, Python oferă instrumente puternice pentru manipularea, vizualizarea și analiza datelor statistice. Aceste biblioteci permit cercetătorilor și analiștilor să exploreze seturile de date, să identifice modele și tendințe, și să obțină înțelegeri profunde din datele brute.

Setul de date Titanic este unul dintre cele mai cunoscute și utilizate seturi de date în domeniul analizei datelor și învățării automate. Acesta conține informații despre pasagerii care au călătorit cu navele Titanic, precum detalii despre sex, vârstă, clasă socială, tarif plătit, dacă au supraviețuit sau nu și alte variabile relevante. Acest set de date este adesea folosit pentru a explora factorii care au influențat șansele de supraviețuire în timpul dezastrului Titanic și pentru a dezvolta modele predictive care să anticipeze șansele de supraviețuire ale pasagerilor pe baza atributelor lor.

2 Partea I

Primul pas pe care trebuie să îl realizați constă în descărcarea setului de date[1]. Observăm că acest set de date conține două părți: una este folosită pentru antrenare și una este folosită pentru testarea soluției. În cadrul acestei cerințe ne vom concentra asupra analizării datelor din fișierul `train.csv`.

După ce ați descărcat acest set de date, va trebui să implementați următoarele cerințe.

Cerința 1: Citiți informațiile din fișierul `train.csv` și examinați structura acestora. Pentru acest lucru, trebuie să determinați programatic (utilizând cod Python) următoarele: numărul de coloane, tipurile datelor din fiecare coloană, numărul de valori lipsă pentru fiecare coloană, numărul de linii, dacă există linii duplicate.

Cerința 2: Determinați care este procentul persoanelor care au supraviețuit și procentul persoanelor care nu au supraviețuit. Determinați care este procentul pasagerilor pentru fiecare tip de clasă (coloana `Pclass`). Determinați care este procentul bărbaților și care este procentul femeilor. Realizați un grafic potrivit pentru prezentarea acestor rezultate.

Cerința 3: Această cerință implică generarea de histograme pentru fiecare coloană cu valori numerice din setul de date Titanic. O histogramă este o reprezentare grafică a distribuției frecvențelor unei variabile continue. Pe axa orizontală sunt incluse intervalele de valori ale variabilei, iar pe axa verticală se reprezintă numărul de exemple din setul de date care sunt incluse în fiecare interval. Histograma oferă o imagine vizuală a modului în care valorile sunt distribuite și permite identificarea tendințelor și a modelului de distribuție al datelor. În cadrul acestei cerințe, pentru fiecare coloană numerică din setul de date Titanic, se va realiza o histogramă pentru a vizualiza distribuția datelor și a evidenția caracteristicile importante ale acestora.

Cerința 4: Identificați coloanele pentru care există valori lipsă. Apoi, pentru fiecare coloană identificată determinați numărul și proporția valorilor lipsă. Determinați care este procentul acestora pentru fiecare dintre cele două clase (coloana `Survived`).

[1] https://www.kaggle.com/c/titanic/data

Cerința 5: Considerăm patru categorii de vârstă: $[0, 20]$, $[21, 40]$, $[41, 60]$, $[61, max]$. Determinați câți pasageri avem pentru fiecare din această categorie. Introduceți o coloană suplimentară și determinați pentru fiecare exemplu din setul de date indexul categoriei din care face parte. Realizați un grafic potrivit pentru a evidenția aceste rezultate.

Cerința 6: Determinați câți bărbați au supraviețuit pentru fiecare dintre cele 4 categorii de vârstă propuse anterior. Realizați un grafic în care să evidențiați cum influențează vârsta procentul de supraviețuire al bărbaților, pe baza informațiilor pe care le avem în setul de date.

Cerința 7: Determinați procentul copiilor aflați la bord (considerăm copii persoane cu vârsta < 18 ani). Realizați un grafic în care să evidențiați rata de supraviețuire pentru copii și pentru adulți.

Cerința 8: Completați valorile lipsă cu cele obținute pentru media pasagerilor care fac parte din aceeași clasă. Spre exemplu, dacă există o înregistrare pentru un pasager care supraviețuiește, dar pentru care nu cunoaștem vârsta, completăm vârsta cu media pasagerilor care au supraviețuit. În cazul în care avem o coloană cu valori categoriale, determinăm cea mai frecventă valoare pentru respectiva clasă.

Cerința 9: Verificați dacă titlurile de noblețe regăsite în coloana Name (Mr., Mrs., Don, etc.) corespund cu sexul persoanei respective. Reprezentați grafic câte persoane corespund fiecărui titlu.

Cerința 10: A influențat starea de a fi singur pe Titanic (nu are deloc rude pe vas) șansele de supraviețuire? Histograma ar putea ajuta la investigarea acestui aspect. Investigați relația dintre tarif, clasă și starea de supraviețuire pentru primele 100 de înregistrări folosind `catplot()` din `seaborn`. (sugestie: folosiți `kind='swarm'` pentru a vedea detalii pe grafic).

3 Partea a II-a

Setul de date Titanic este un set de date clasic utilizat în domeniul învățării automate și al analizei datelor. Cu toate acestea, seturile de date pot conține valori aberante (outliers) care pot influența negativ rezultatele analizei. Scopul este să curățăm setul de date Titanic prin eliminarea acestor valori aberante pentru a putea fi folosit în antrenarea unui model de predicție.

Realizați o analiză a setului de date pentru a identifica posibile outlier-e. O abordare simplă poate fi vizualizarea distribuțiilor variabilelor și identificarea valorilor care se află semnificativ în afara distribuției obișnuite. Pentru acest lucru, puteți porni de la graficele realizate de partenerul de echipă la **Partea I**.

Cerința 1: Folosiți interquartile range pentru a identifica și elimina outlier-ele. O valoare este considerată outlier dacă este mai mică decât Q1 - 1.5IQR sau mai mare decât Q3 + 1.5IQR, unde Q1 și Q3 sunt primul și al treilea percentile, iar IQR este diferența dintre Q3 și Q1. Dacă luăm de exemplu variabila "age" din setul de date Titanic, putem calcula Q1 și Q3, apoi IQR. Valorile care sunt în afara intervalului Q1 - 1.5IQR și Q3 + 1.5IQR pot fi considerate outlier-e și eliminate.

Cerința 2: Calculați Z-score pentru fiecare observație și eliminați valorile care au un Z-score absolut mai mare decât un anumit prag (de exemplu, 3 sau 4). Z-score reprezintă numărul de deviații standard față de media setului de date. Pentru aceeași variabilă "age", putem calcula

Z-score pentru fiecare vârstă și să eliminăm valorile care au un Z-score mai mare de, să zicem, 3. Valorile cu un Z-score mai mare de 3 ar putea fi considerate outlier-e.

Cerința 3: Asigurați-vă că documentați și explicați procesul de curățare a datelor și motivele din spatele eliminării. După eliminarea outlier-elor, puteți efectua o validare suplimentară pentru a vă asigura că distribuția datelor este mai uniformă și că nu ați eliminat în mod accidental valori importante. Pentru acest lucru, puteți ruga partenerul de echipă să ruleze primele cerințe pentru setul de date rezultat după curățarea datelor.

Cerința 4: Dezvoltarea unui model de clasificare pentru prezicerea șanselor de supraviețuire. Pentru acest lucru, veți avea de realizat următoarele:

- Protocolul de testare: Împărțiți setul de date în două componente: 80% pentru antrenare și 20% pentru validare.
- Preprocesarea datelor: încărcarea datelor (pandas), înlăturarea valorilor lipsă (ex: medie), convertirea coloanele categorice (ex: Sex, Embarked) în valori numerice, normalizarea caracteristicilor numerice.
- Antrenare model: alegerea unui algoritm (Decision-Tree, Random Forest, etc.), antrenare pe setul de date de antrenament
- Evaluare model: folosirea setului de testare pentru predicție pe baza modelului rezultat în urma antrenării, evaluarea performanței prin indicatori (accuracy, loss). Realizarea de grafice relevante.

4 Punctaj

Proiectul va fi încărcat pe Moodle, de fiecare membru al echipei (fiecare student își încarcă partea lui), sub forma unei arhive **.zip** cu următorul conținut:

- un director cu numele **ParteaI** (**pentru cei care au ales Partea I** – 2) ce conține următoarele subdirectoare:
  - **Surse** - toate fișierele cu cod folosite în realizarea temei (`.py` / `.ipynb`)
  - **README.pdf** - fișierul care conține toate histogramele rezultate, toate graficele create și documentația.
  - **Date** - toate fișierele rezultate din modificări ale setului de date (ex. cerința 8)

  În fișierul fișierul README veți descrie modul de rezolvare pentru fiecare cerință din Partea I, răspunsurile la întrebările din cerință și alte observații; prima linie a fișierului va conține numele complet, seria și grupa studentului care a rezolvat partea I.
- un director cu numele **ParteaII** (**pentru cei care au ales Partea a II-a** – 3) ce conține următoarele subdirectoare:
  - **Surse** - toate fișierele cu cod folosite în realizarea temei (`.py` / `.ipynb`)
  - **Date** - toate fișierele rezultate din modificări ale setului de date (ex. cerința 1)
  - **README.pdf** - fișierul care conține toate graficele create și documentația.

În fișierul README veți descrie modul de rezolvare pentru fiecare cerință din Partea a II-a, răspunsurile la întrebările din cerință și alte observații; prima linie a fișierului va conține numele complet, seria și grupa studentului care a rezolvat partea a II-a.

Punctajul pentru fiecare parte este împărțit după cum urmează:

Partea I	Punctaj
Cerința	fiecare cerință valorează 5 puncte (2p calitatea codului, 2p rezultatul obținut, 1p prezentarea din documentație)
BONUS folosire Git	20 puncte
TOTAL	**70 puncte**

Partea a II-a	Punctaj
Cerința 1	10 puncte (40% cod + 40% rezultat + 20% documentație)
Cerința 2	10 puncte (40% cod + 40% rezultat + 20% documentație)
Cerința 3	10 puncte (40% cod + 40% rezultat + 20% documentație)
Cerința 4	20 puncte (40% cod + 40% rezultat + 20% documentație)
BONUS folosire Git	20 puncte
TOTAL	**70 puncte**

Atenție!

- Pentru a primi punctaj, trebuie să **prezentați** proiectul în ultima săptămână a semestrului.
- **Toate soluţiile trimise vor fi verificate, folosind o unealtă pentru detectarea plagiatului. În cazul depistării codului copiat (de pe Internet, colegi, din surse generate cu tool-uri tip ChatGPT), întregul punctaj pentru proiect este anulat.**
- Pentru orice întrebare puteți folosi forumul.
- Punctajul bonus pentru folosirea utilitarului `git` este acordat raportat la numărul de cerințe realizate și la complexitatea funcționalităților utilizate.