Aplicație Practică 2

Cirjontu Ionela

January 17, 2025

1 Descrierea problemei

Scopul aplicației este de a prezice ratingul fiecărei activități tematice într-o anumită țară pentru a ajuta în luarea deciziei deschiderii unui hotel.

2 Justificarea abordării

Am folosit algoritmii deja implementați în biblioteca *sklearn* pentru a observa care dintre aceștia oferă un răspuns cât mai apropiat de realitate. Pentru fiecare experiment am verificat performanța algoritmului pe setul de date folosind RMSE: rădăcina pătrată a valorii MSE. MSE: eroarea medie pătratică, suma pătratului diferenței dintre valorile ajustate de model și valorile observate, împărțite la numărul de puncte de istoric, minus numărul de parametri din model.

2.1 Soluția 1

K-nn. Am testat algoritmul pentru diferiți k și am realizat grafice pentru fiecare, în funcție de ratingul prezis și cel real. Răspunsurile corecte ar trebui să se afle pe dreapta roșie desenată.
Primul test l-am făcut pentru k=2, însă rezultatele sunt depărtate de răspunsul corect, fără să se observe măcar o concentrare accentuată în zona liniei roșii. RMSE = 1.3721.

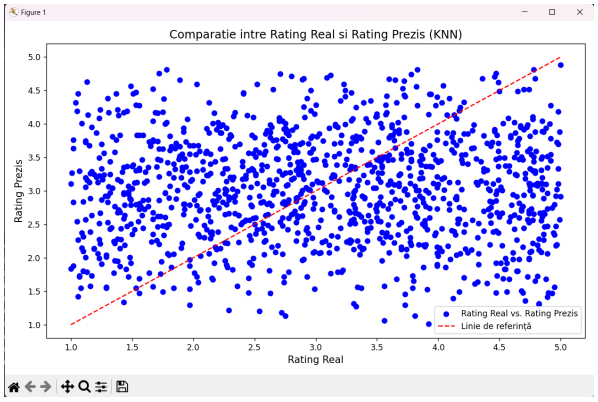

Figure 1: k=2

Pentru k = 3, rezultatele se apropie de linia roșie. RMSE=1.3057.

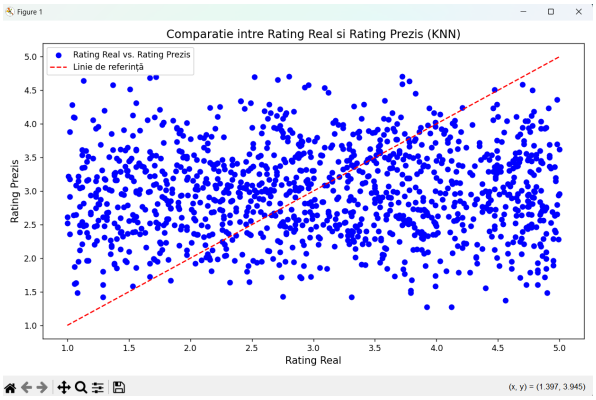

Figure 2: k=3

Pentru k = 4, apropierea de centrul graficului continuă. RMSE=1.2609.

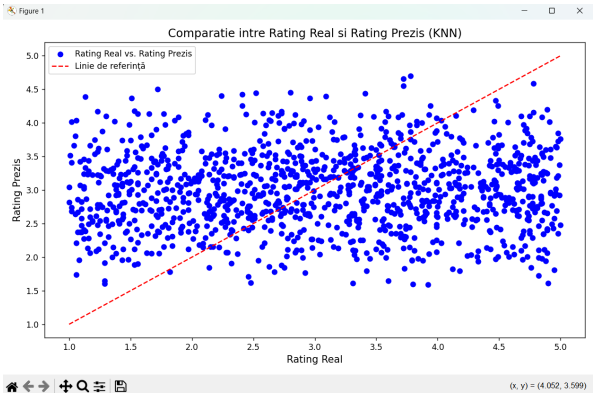

Figure 3: k=4

Pentru k = 5, nu se observă o schimbare vizuală. RMSE = 1.2426.

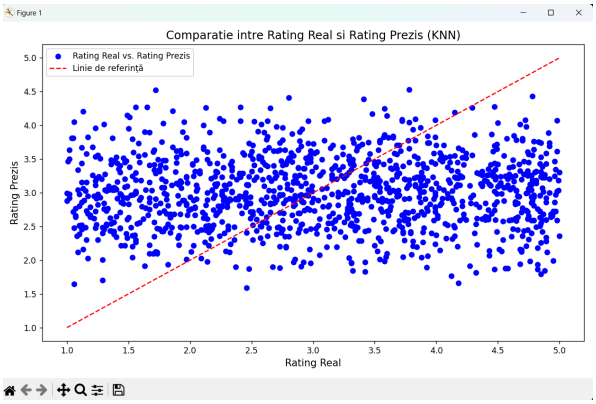

Figure 4: k=5

Pentru k = 6, rezultatele se apropie de centrul graficului, însă pentru rating prezis între 4,3 și 5 nu există instanțe, însă în setul de validare există. RMSE=1.2243.

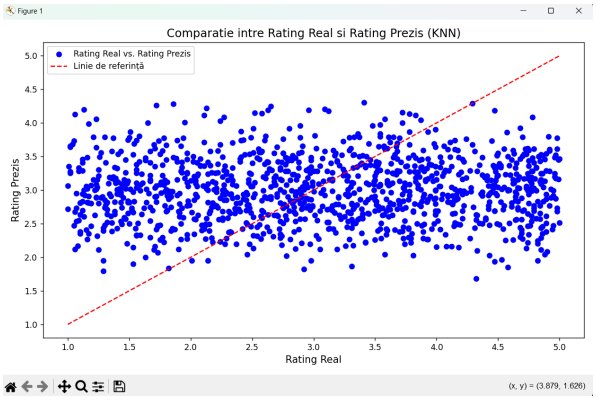

Figure 5: k=6

2.2 Soluția 2

Regresia logistică folosind metoda gradientului ascendent din *sklearn*. Deși regresia este specializată pentru problemele de clasificare cu 2 clase, putem implementa și multi-clasă. Am aplicat regresia doar cu etichetele 1,2,3,4 și 5 (int(eticheta)). Am realizat de asemenea un grafic, însă având valorile reale de rating între 0-5, pentru un rating de 1 și 1,9, algoritmul va aproxima la 1, dar diferența este mai mare de 10% din ratingul total posibil. Etichetele sunt suprapuse în grafic din cauza acestei aproximări. Avem și etichete conforme cu realitatea codificată (int(etichetaReala) = int(etichetaPrezisa)), dar avem și multe instanțe depărtate de rezultatul așteptat. RMSE = 1.5761.

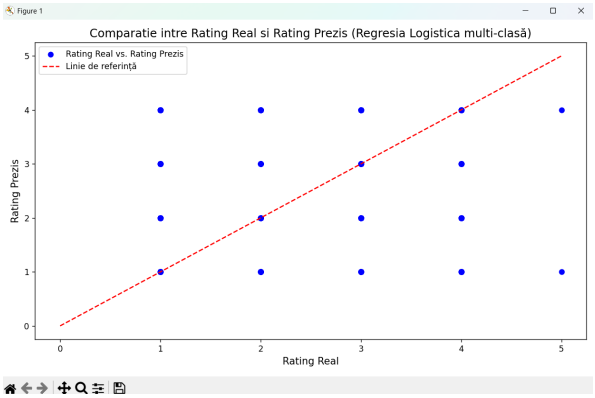

Figure 6: Regresia logistică folosind metoda gradientului ascendent

2.3 Soluția 3

AdaBoost din *sklearn*, cu clasificatorul slab un arbore de decizie de diferite adâncimi.
RMSE = 1.1424.

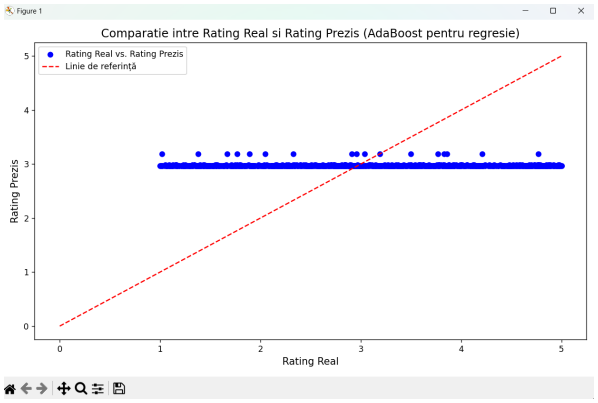

Figure 7: AdaBoost cu arbore de decizie de adâncime maxim 1

RMSE = 1.1422.

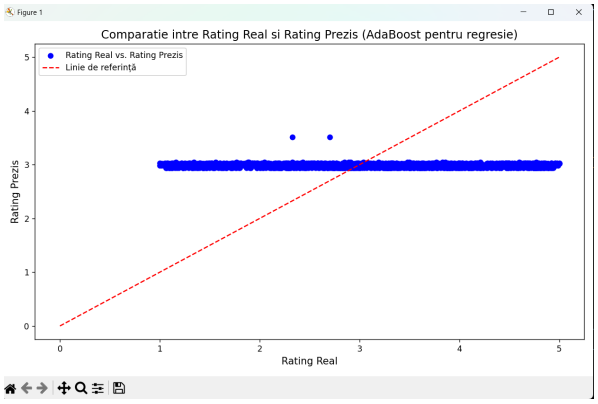

Figure 8: AdaBoost cu arbore de decizie de adâncime maxim 2

RMSE = 1.1409.

Figure 9: AdaBoost cu arbore de decizie de adâncime maxim 3

3 Concluzii

Algoritmul cu RMSE cel mai mic este AdaBoost cu clasificatorul slab un arbore de decizie de adâncime maxim 3. În forma în care am folosit regresia logistică nu ne este de ajutor deoarece pentru 3 ratinguri de 3.1, 3.2, 3.3 nu va putea realiza o departajare. AdaBoost tinde să fie mai eficient pentru problemele de ranking comparativ cu k-NN, deoarece are capacitatea de a îmbunătăți modelele mai slabe, învățând

din greșelile anterioare, gestionând date complexe și prevenind supraînvățarea. În schimb, k-NN poate fi mai vulnerabil la zgomot și variații neașteptate în date, ceea ce îl face mai puțin eficient atunci când este vorba de rankingul categoriilor în contexte complicate.