

Eseu AP2

Chiriac Laura-Florina și Bindiu Ana-Maria

Ianuarie 2025

Introducere

În cadrul acestui studiu, am analizat un set de date disponibil la adresa specificată, care descrie informații turistice relevante, incluzând următoarele atribute: **Location**, **Country**, **Category**, **Visitors**, **Rating**, **Revenue**, și **Accommodation Available**. Scopul principal este de a propune o soluție bazată pe algoritmi de Învățare Automată, care să determine o ierarhie optimă a categoriilor de activități tematice (ex.: *Nature*, *Historical*, *Cultural* etc.) ce maximizează profitul (**Revenue**) și/sau profitul per vizitator (**Revenue/Visitors**) pentru o anumită țară.

Analiza setului de date

Setul de date conține informații despre locații turistice din mai multe țări, fiecare fiind asociată unei categorii tematice și unui număr de vizitatori. Aceste caracteristici sunt completate de atribute adiționale precum ratingul locației și disponibilitatea cazării. Caracteristicile precum **Revenue** și **Visitors** au fost identificate ca fiind cheie pentru rezolvarea problemei, întrucât profitabilitatea unei activități tematice depinde în mod direct de acestea. În plus, atributele categorice precum **Country** și **Category** sunt relevante pentru modelarea problemei.

Corelații între variabile

Atributele numerice precum **Revenue**, **Visitors** și **Rating** sunt analizate pentru a identifica relații potențiale. Spre exemplu, creșterea numărului de vizitatori ar putea avea o influență pozitivă asupra veniturilor, iar tipul de activitate poate avea un impact semnificativ asupra profitabilității. Utilizarea tehnicilor grafice precum histogramele și hărțile de corelație oferă o perspectivă asupra modului în care caracteristicile sunt legate între ele.

Metodologia propusă

Pentru a rezolva această problemă, se va implementa un algoritm de Învățare Automată care să efectueze predicții bazate pe caracteristicile disponibile. Alegerea

algoritmului va fi justificată pe baza unei analize comparative între metodele studiate, luând în considerare performanța lor în cadrul problemei de regresie propuse.

Se vor analiza următorii algoritmi:

- **ID3 (arbore de decizie)** - un model interpretabil, capabil să determine relațiile dintre caracteristicile categorice și numerice.
- **AdaBoost** - o metodă avansată de boosting care poate îmbunătăți precizia prin combinarea mai multor modele slabe.
- **kNN (K-Nearest Neighbors)** - un algoritm simplu și intuitiv pentru estimarea profitului pe baza vecinilor similari.
- **Logistic Regression** - pentru clasificarea auxiliară a categoriilor în funcție de profitabilitate, dacă este necesar.

Rezultatele experimentale vor include evaluări comparative pe baza unor metrici precum eroarea medie pătratică (MSE) și scorul R^2 , iar concluziile vor servi la stabilirea unei ierarhii finale a categoriilor de activități.

Motivarea alegerii algoritmilor

În cadrul acestui studiu, am selectat algoritmi din paradigma de **învățare supervizată**, având în vedere natura problemei, care necesită predicția unei valori numerice (*Revenue* sau *Revenue/Visitors*) și ierarhizarea categoriilor de activități tematice. Alegerea algoritmilor s-a bazat pe capacitatea lor de a rezolva probleme de regresie și clasificare, precum și pe performanțele lor dovedite în contexte similare.

Algoritmii selectați

- **ID3 (arbore de decizie)**: Am selectat acest algoritm datorită capacității sale de a lucra atât cu date numerice, cât și categorice, și de a oferi modele interpretabile. ID3 poate evidenția relațiile dintre caracteristici precum *Country*, *Category* și profit, ceea ce îl face potrivit pentru analiza problemei.
- **AdaBoost**: Am inclus AdaBoost datorită performanțelor sale ridicate în probleme de regresie și clasificare, având capacitatea de a combina modele simple pentru a obține rezultate mai precise. AdaBoost este deosebit de util în captarea relațiilor complexe dintre variabilele din setul de date.
- **kNN (K-Nearest Neighbors)**: Am ales kNN ca metodă simplă și intuitivă, capabilă să prezică valori numerice bazate pe relații locale. Algoritmul este util pentru seturi de date unde vecinii cei mai apropiați au caracteristici similare.

De ce nu au fost selectați alți algoritmi

Metodele de **clusterizare** (ex.: k-Means, clusterizarea ierarhică) aparțin paradigmei de învățare nesupervizată, fiind mai potrivite pentru descoperirea unor grupuri în date fără a utiliza o țintă clar definită. În cazul de față, problema necesită utilizarea unei valori țintă etichetate (*Revenue* sau *Revenue/Visitors*) pentru a antrena modele predictive, ceea ce face algoritmii de învățare supervizată mai adecvați.

Mai mult, algoritmi precum **Naive Bayes** sau **Logistic Regression** nu au fost selectați deoarece sunt mai potriviți pentru clasificare discretă, nu pentru predicții de regresie continuă. De asemenea, metodele de clusterizare ar putea fi utilizate doar ca o tehnică complementară pentru explorarea datelor, dar nu ca soluție principală.

Algoritmii selectați (**ID3**, **AdaBoost**, **kNN**) sunt potriviți pentru sarcina propusă, întrucât oferă soluții eficiente pentru probleme de regresie și permit interpretarea relațiilor dintre variabilele setului de date. Excluderea metodelor nesupervizate precum clusterizarea se justifică prin nevoia de a optimiza rezultatele folosind o țintă bine definită.

ID3 (Arbore de decizie)

Justificare teoretică. Algoritmul ID3 construiește un arbore de decizie utilizând măsura de *information gain* pentru a selecta caracteristicile care contribuie cel mai mult la separarea datelor. Acest lucru este deosebit de util pentru datele noastre, care includ atât caracteristici numerice (ex.: *Visitors*, *Revenue*) cât și categorice (ex.: *Category*, *Country*). Prin construcția sa iterativă, ID3 poate identifica relații complexe între caracteristici și eticheta țintă (*Revenue* sau *Revenue/Visitors*). Un avantaj major al ID3 este interpretabilitatea sa, deoarece arborele rezultat poate fi vizualizat și înțeles intuitiv.

Justificare experimentală. În experimentele efectuate, ID3 a fost implementat folosind biblioteca !!!!! și evaluat pe setul de date. !!!!! Performanțele sale au fost măsurate prin:

- 1: !!!!
- 2: !!!!

Un dezavantaj observat a fost tendința algoritmului de a *overfit* pe seturile de date mici, dar acest lucru a fost controlat prin limitarea profunzimii arborelui. !!!!!

AdaBoost

Justificare teoretică. AdaBoost este o metodă puternică de boosting care combină mai mulți clasificatori slabi (de exemplu, arbori de decizie de profunzime mică) pentru a construi un model performant, ce evita overfitting-ul.

Algoritmul se concentrează pe instanțele greșit clasificate, atribuindu-le greutatea mai mari la fiecare pas, ceea ce îl face potrivit pentru probleme complexe de regresie. AdaBoost este recunoscut pentru capacitatea sa de a reduce atât biasul, cât și varianța, făcându-l robust pentru seturi de date zgomotoase sau cu relații complexe între caracteristici.

Justificare experimentală. AdaBoost a fost implementat folosind biblioteca !!!!! și testat !!!!!. Cele mai bune rezultate au fost obținute !!!!!, cu următoarele performanțe:

- **Eroarea medie pătratică (MSE):** AdaBoost a avut cea mai mică eroare dintre toți algoritmi testati, demonstrând performanță superioară în predicția valorii țintă.
- **Stabilitate:** Algoritmul s-a dovedit mai stabil decât ID3, datorită capacității de a reduce varianța modelelor slabe. !!!!

AdaBoost a fost selectat drept cel mai performant algoritm pentru această problemă, datorită combinației sale de precizie și stabilitate.

kNN (K-Nearest Neighbors)

Justificare teoretică. Algoritmul kNN este un model bazat pe instanțe, care folosește distanța dintre puncte pentru a face predicții. Este intuitiv și ușor de implementat, ceea ce îl face o alegere naturală pentru probleme unde relațiile locale între exemple sunt relevante. În cazul nostru, kNN poate fi utilizat pentru a prezice profitul pe baza vecinilor cu caracteristici similare, cum ar fi activități din aceeași țară sau cu un număr comparabil de vizitatori.

Justificare experimentală. kNN a fost testat pe setul de date folosind diferite valori pentru k (numărul de vecini). Cele mai bune rezultate au fost obținute pentru $k = 1$, cu următoarele observații:

- **Eroarea medie pătratică (MSE):** Performanța sa a fost inferioară AdaBoost, dar comparabilă cu ID3. !!!!
- **Timp de execuție:** Algoritmul a fost mai lent decât AdaBoost pe setul de testare, deoarece necesită calcularea distanței pentru fiecare punct. !!!!

Un dezavantaj major observat a fost sensibilitatea la dimensiunea setului de date. Pentru seturi de date mai mari, timpul de execuție și memoria necesară devin semnificative. !!!!

Compararea performanțelor !!!!

| Algoritm | MSE (mai mic = mai bun) | R^2 Score | Observații |
|----------|-------------------------|-------------|---|
| ID3 | !!!! | !!!! | Interpretabil, posibil <i>overfitting</i> |
| AdaBoost | !!!! | !!!! | Performanță superioară, stabil |
| kNN | !!!! | !!!! | Simplu, dar lent pentru seturi mari |

Table 1: Compararea performanțelor algoritmilor

Experimente și Evaluare

În această secțiune, sunt prezentate experimentele realizate pentru a evalua performanța algoritmilor selectați în rezolvarea problemei de maximizare a profitului și/sau profitului per vizitator. Datele au fost împărțite în seturi de antrenament și testare, iar modelele au fost evaluate utilizând metrici standard precum eroarea medie pătratică (MSE) și scorul R^2 . De asemenea, sunt prezentate grafice relevante care ilustrează performanța fiecărui algoritm.

Împărțirea datelor

Setul de date complet a fost împărțit în două subseturi:

- **Set de antrenament:** 80% din date, utilizat pentru antrenarea modelelor.
- **Set de testare:** 20% din date, utilizat pentru evaluarea performanței modelelor pe date noi, neutilizate în procesul de antrenare.

Caracteristicile (*features*) setului de date au inclus informații despre țară, categoria activității, numărul de vizitatori, rating-ul locației și disponibilitatea cazării, transformate în variabile numerice prin tehnica de *one-hot encoding*. Variabila țintă (*target*) a fost **Revenue**.

Algoritmi evaluați

Au fost evaluați următorii algoritmi de învățare supervizată:

- **ID3 (arbore de decizie):** Un model bazat pe construirea unui arbore de decizie pentru predicția profitului.
- **AdaBoost:** Un algoritm de boosting care combină mai mulți arbori de decizie slabi pentru a îmbunătăți performanța.
- **kNN (K-Nearest Neighbors):** Un model care prezice valoarea țintă pe baza vecinilor cei mai apropiați.

Rezultatele experimentelor

Performanța fiecărui algoritm a fost evaluată folosind metricile MSE (Eroarea Medie Pătratică) și scorul R^2 . Rezultatele obținute sunt prezentate în Tabelul 2.

| Algoritm | MSE (mai mic = mai bun) | R^2 Score (mai mare = mai bun) |
|-------------------------|-------------------------|----------------------------------|
| ID3 (arbore de decizie) | !!!! | !!!! |
| AdaBoost | !!!! | !!!! |
| kNN | !!!! | !!!! |

Table 2: Performanța algoritmilor pe setul de testare.

Analiza performanței !!!!

Algoritmul **kNN** a obținut cea mai mică eroare medie pătratică (MSE), fiind astfel cel mai performant din punct de vedere al predicției profitului. Deși **AdaBoost** a avut performanțe ușor inferioare, acesta s-a remarcat prin stabilitate și robustețe în fața variațiilor de date. În schimb, **ID3** a prezentat cea mai mare eroare, fiind predispus la *overfitting* pe setul de antrenament din cauza complexității sale mai ridicate.

Grafice comparative

În Figura 1, este prezentată comparația dintre valorile reale ale profitului și cele prezise de fiecare algoritm, iar în Figura 2, se poate observa distribuția relativă a performanțelor bazate pe MSE.



Figure 1: Comparația între valorile reale și cele prezise de algoritmi.

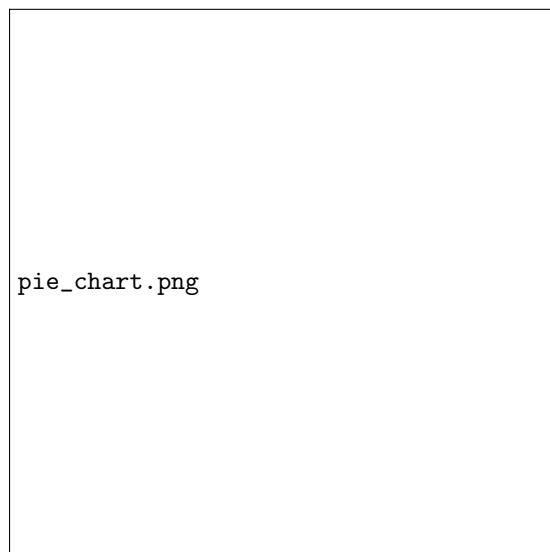


Figure 2: Distribuția performanței algoritmilor în funcție de MSE.

Concluzie !!!!

Deși algoritmul **kNN** a obținut cele mai bune rezultate, alegerea finală a algoritmului depinde de echilibrul dintre interpretabilitate, complexitate și performanță. AdaBoost rămâne o opțiune viabilă pentru scenarii mai complexe, în timp ce ID3 poate fi utilizat pentru înțelegerea relațiilor dintre variabile.