Academia de Studii Economice din București

Facultatea de Cibernetică, Statistică și Informatică Economică

Program de masterat – Statistică Aplicată și Data Science

Clasificarea Restaurantelor din Ghidul Michelin:

Analize și predicții folosind tehnici de Data Mining

Coordonator:

Profesor univ. dr. COSTEA Adrian

Student:

DRĂGHICESCU Alexandra-Elena

București 2024

CUPRINS

I.INTRODUCERE

Ghidul Michelin este unul dintre cele mai prestigioase și cunoscute sisteme de clasificare a restaurantelor la nivel mondial. Publicat pentru prima dată în 1900 de către compania franceză Michelin, ghidul a început ca un ajutor pentru șoferi, oferind informații despre locuri de cazare, benzinării și reparații auto. Începând cu anii 1920, ghidul s-a extins pentru a include recenzii despre restaurante, iar din 1936 a adoptat sistemul de evaluare în stele, care s-a menținut până astăzi.

Clasificarea Michelin cuprinde următoarele categorii principale:

- **1 Stea**: Restaurant de foarte bună calitate în categoria sa
- **2 Stele**: Restaurant cu o calitate excelentă, merită un ocol.
- **3 Stele**: Restaurant de excepție, merită o călătorie specială.
- **Bib Gourmand**: Acest premiu este acordat restaurantelor care oferă mâncare decalitate la prețuri accesibile, fiind o alternativă mai puțin costisitoare față de restaurantele cu stele Michelin, dar care menține standardele ridicate.
- **Selected Restaurants**: Această categorie include restaurante recomandate de ghid, dar care nu au obținut încă una dintre celelalte distincții.

Recent, distincția **Green Star** a fost introdusă de Ghidul Michelin în anul 2020. Aceasta recunoaște restaurantele care adoptă practici sustenabile și promovează un impact pozitiv asupra mediului, prin aprovizionare locală, reducerea risipei alimentare, utilizarea energiei regenerabile și alte inițiative ecologice.

Sistemul de evaluare Michelin este cunoscut pentru rigurozitatea și discreția inspectorilor săi, care vizitează restaurantele în mod anonim. Inspectorii le evaluează pe baza unor criterii standardizate, cum ar fi calitatea ingredientelor, tehnica de preparare, creativitatea și constanța.

În viața cotidiană, clasificarea restaurantelor după nivelul premiilor obținute în **Ghidul Michelin** este o provocare pentru entitățile din industria culinară, dar și pentru cei care doresc să înțeleagă ce face un restaurant cu o asemenea distincție. În acest context, apare problema **identificării factorilor** care contribuie la obținerea unei astfel de distincții. Deși Ghidul Michelin nu face publice criteriile exacte, o analiză a datelor, în special asupra celor calitative, poate dezvălui tiparele semnificative și caracteristicile relevante care fac un restaurant să se distingă la nivel înalt.

Scopul acestui proiect este de a construi un model predictiv care să clasifice restaurantele din Ghidul Michelin pe baza unor trăsături precum locația, tipul de bucătărie, prețul sau facilitățile oferite. Utilizând metode de **data mining** și tehnici de clasificare, proiectul își propune să determine cum caracteristicile contribuie la obținerea unui anumit premiu Michelin. Acest model poate fi utilizat atât pentru înțelegerea factorilor specifici, cât și pentru a oferi recomandări restaurantelor care aspiră să avanseze în clasificarea Michelin.

Problema identificată: înțelegerea factorilor de influență în clasificarea Michelin

Variabilă „Target”: tipul de distincție primită de fiecare restaurant(*Award type*)

Ca și tehnici specifice de data mining, pentru comparabilitate, am ales :

- **Random Forest** (pădure de arbori decizionali)
- **Logistic Regression** (regresie logistică)
- **Naïve Bayes** (clasificator Gaussian)

Diferite studii de specialitate au arătat utilitatea acestor tehnici de data mining în analiza și predicția tendințelor în industria ospitalității, în diferitele ei ramuri.

Pe de-o parte, plecând de la idea arborilor decizionali, prin așa-numită metoda „pădure de arbori decizionali"(random forest), un rezultat semnificativ a fost obținut în analiza asupra factorilor ce influențează alegerea unui anumit cartier pentru o locație de cazare de tip AirBnB, la nivelul orașului Beijing(Jiang et al., 2022).

Un exemplu reprezentativ pentru regresia logistică îl reprezintă aplicarea tehnicii pentru a obține satisfacția clienților într-un restaurant unde lucrează studenți(Larasati et al., 2012). Prin lucrarea lor, autorii au vrut să vadă care sunt factorii de influență pentru satisfacția consumatorilor restaurantului și cum, eventual, ar putea îmbunătății anumite aspecte.

Pe de-altă parte, clasificatorul Naive Bayes are atât importanță pentru consumatori, cât și pentru antreprenorii din industrie. De exemplu, prin intermediul acestei metode, pe baza recenziilor turiștilor care au efectuat deja sejururi în cadrul anumitor hoteluri, se pot crea sisteme de recomandare, în funcție de diferiții factori de interes personal sau de caracteristicile unității de cazare(Awotunde et al., 2023). Prin astfel de sisteme, atât consumatorii de rând pot alege după propriile preferințe locul ideal, cât și administratorii de hoteluri pot îmbunătății serviciile acolo unde este cazul.

Mai mult, și cu ajutorul tehnicilor avansate de Machine Learning, Naive Bayes poate veni cu rezultate surprinzătoare în ceea ce privește procesarea și clasificarea recenziilor consumatorilor restaurantelor, pentru ca serviciile oferite potențialilor noi clienți să fie la cele mai înalte standarde și eventual, pentru a primi diferite distincții(Patil et al., 2024).

Având un punct de pornire din studiile deja efectuate, în studiul de caz am dorit a prezenta într-o manieră exploratorie potențialii factori care contribuie la clasificarea restaurantelor după Ghidul Michelin.

II.DESCRIEREA DATELOR

Pentru acest studiu de caz, am ales un set de date deja prelucrat pe Kaggle(**Michelin Guide Restaurants**) descărcat în jurul datei de 20 oct. 2024. Datele sunt actualizate cu frecvență lunară și au ca sursă directă site-ul oficial Michelin Guide.

În setul inițial sunt prezente 14 variabile cu informații despre cele 17294 restaurante atestate în **Michelin Guide** la nivel mondial. Printre aceste variabile am identificat atât date cantitative(coordonatele geografice, nr. tel.), dar și calitative(nume, adresă), după cum urmează în secvența din Python.

După necesitate, aceste date au fost transformate în format specific pentru a putea fi prelucrate.

```
Data columns (total 14 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Name                   17294 non-null  object
 1   Address                17294 non-null  object
 2   Location               17294 non-null  object
 3   Price                  17293 non-null  object
 4   Cuisine                17293 non-null  object
 5   Longitude              17294 non-null  float64
 6   Latitude               17294 non-null  float64
 7   PhoneNumber            16928 non-null  float64
 8   Url                    17294 non-null  object
 9   WebsiteUrl             14884 non-null  object
 10  Award                  17294 non-null  object
 11  GreenStar              17294 non-null  int64
 12  FacilitiesAndServices  16409 non-null  object
 13  Description            17293 non-null  object
```

Fig. 1 Descrierea setului de date – prelucrare Python

II.1 Analiza Exploratorie a Datelor

La nivel vizual, în Python, atributele setului de date se pot vedea prin tehnicile de Analiza Exploratorie a Datelor. Pentru analiza din acest proiect, din cele 14 variabile am ales a fi reprezentative doar 8, după cum urmează în secvența de cod.

```
Data columns (total 8 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Name                  17294 non-null  object
 1   Location              17294 non-null  object
 2   Price                 17293 non-null  object
 3   Cuisine               17293 non-null  object
 4   Url                   17294 non-null  object
 5   Award                 17294 non-null  object
 6   GreenStar             17294 non-null  int64
 7   FacilitiesAndServices 16409 non-null  object
dtypes: int64(1), object(7)
```

Fig. 2 Descrierea noului set – prelucrare Python

```
df.isnull().sum()
```

```
Name                      0
Location                  0
Price                     1
Cuisine                   1
Url                       0
Award                     0
GreenStar                 0
FacilitiesAndServices   885
```

Printre acestea am identificat și înregistrările cu valori nule. Pentru a nu afecta foarte mult modelul și rezultatele, am ales eliminarea din listă a restaurantului despre care nu exista informații în câmpurile *Price* și *Cuisine*, considerând că variabila *FacilitiesAndServices* ca fiind opțională, ajungând la un număr de **17293** restaurante în ghidul Michelin la nivel mondial.

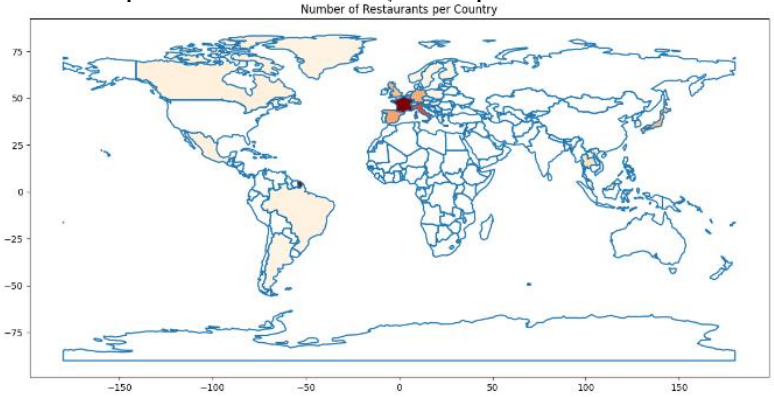
Fig. 3 Verificare valori nule - prelucrare Python

În funcție de diferitele variabile, am identificat și cele mai populare valori, de exemplu, cele mai multe restaurante se află în Tokyo,Japonia(547 dintr-o listă de 5640 de orașe), iar cele mai multe au primit distincția de Selected Restaurants, considerând a fi dificil de obținut măcar una dintre cele 3 stele Michelin.

	count	unique	top	freq
Name	17293	16633	Racines	9
Location	17293	5640	Tokyo, Japan	547
Price	17293	32	€€	4684
Cuisine	17293	1612	Modern Cuisine	2735
Url	17293	17293	https://guide.michelin.com/en/prague/prague/re...	1
Award	17293	5	Selected Restaurants	10509
FacilitiesAndServices	16409	878	Air conditioning	1492

Fig. 4 Descrierea variabilelor de interes - prelucrare Python

Răspândirea la nivel de țară se poate vedea atât sub formă tabelară, cât și hartă.




```
Location
Tokyo, Japan                547
Paris, France               446
New York, USA               390
London, United Kingdom      343
Singapore                   283
                            ...
Eext, Netherlands             1
Medemblik, Netherlands        1
Flobecq, Belgium              1
Hardenberg, Netherlands       1
Wijdewormer, Netherlands      1
```

Fig. 5 Reprezentare distribuție restaurante în funcție de țară - prelucrare Python

Din această reprezentare, se poate observa ca aceste restaurante din rețeaua Michelin se concentrează, în principal, în jurul zonelor dezvoltate cu un nivel de trai foarte ridicat.

Mai mult, se poate observa distribuția restaurantelor și în funcție de celelalte caracteristici, cea mai reprezentativă fiind, în mod evident, distribuția în funcție categoria atribuită de ghidul Michelin.

```
Absolute values of restaurants for each award category:
Award Category: Selected Restaurants, Absolute Count: 10509
Award Category: Bib Gourmand, Absolute Count: 3240
Award Category: 1 Star, Absolute Count: 2905
Award Category: 2 Stars, Absolute Count: 494
Award Category: 3 Stars, Absolute Count: 145

Total number of restaurants: 17293
```

Fig. 6 Clasificarea restaurantelor în funcție de distincția Michelin - prelucrare Python

Se poate concluziona deci, că o înaltă distincție era o raritate și relativ greu de obținut, fiind reprezentată de mai puțin de 1% dintre restaurantele incluse în ghidul Michelin.

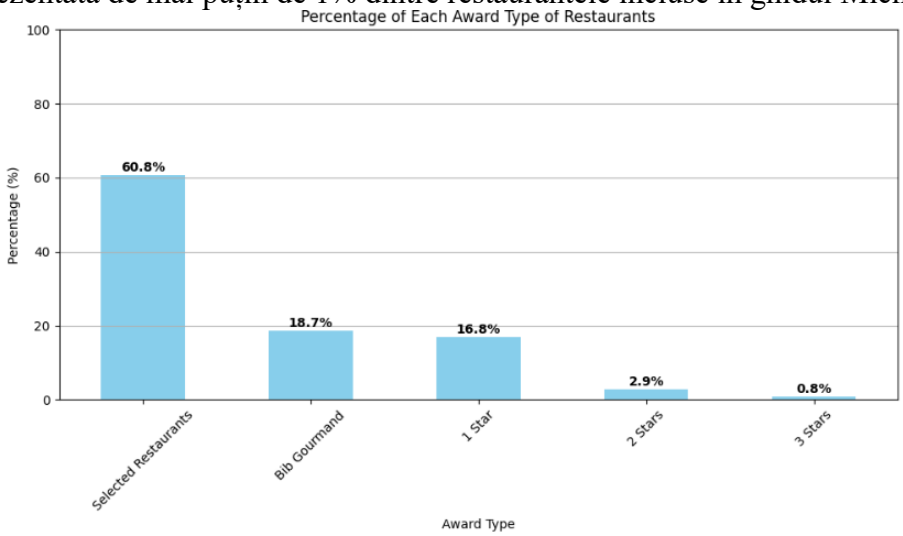


Fig. 7 Distribuția restaurantelor în funcție de distincția Michelin - prelucrare Michelin

II.2 Preprocesarea datelor

Etapa de preprocesare a datelor este esențială pentru a pregăti datele brute obținute din setul din Ghidul Michelin. Această etapă constă în a le transforma într-un format adecvat pentru modelele de clasificare, pentru a putea fi prelucrate.

Pentru preprocesare, am utilizat urmat etapele:

- **Transformarea variabilelor categorice:**

Transformarea variabilei *Award* pentru a se codifica într-un format numeric utilizând label encoding pentru a permite clasificarea. În mod similar am procedat și pentru codificarea variabilei *Price.* Variabila este exprimată sub formă de simboluri ($, $$,$$$,$$$$) pentru a indica nivelul de preț, fiind o variabilă categorică am ales codificare cu simboluri de la 1 la 4.

```
Award Encoding Legend:    Price_enc
1 Star: 0                 2    7104
2 Stars: 1                3    5393
3 Stars: 2                4    3231
Bib Gourmand: 3           1    1565
Selected Restaurants: 4   Name: count, dtype: int64
```

Fig. 8 Codificarea variabilelor Award&Price - prelucrare Python

- **Prelucrarea variabilelor cu atribute multiple:**

În cazul în care o variabilă conține liste de atribute (ex.: Location, FacilitiesAndServices), se aplică **one-hot encoding** pentru fiecare atribut în parte. Astfel, fiecare facilitate devine o variabilă binară, indicând prezența sau absența acesteia. În cazul varibilei despre locație, informația a fost împărțită în două variabile separate, cu numele orașului și țara de origine.

- **Agregarea noilor variabile codificate într-un set de date**
- **Împărțirea noului set de date, în set de antrenare și testare**:

Pentru a permite efectuarea de calcule și predicții folosind tehnicile menționate în ipoteză, 80% din valori reprezintă setul de antrenare, iar 20% reprezintă setul de test, valori care vor fi previzionate.

În urma acestor pași, setul de date este pregătit pentru aplicarea tehnicilor de clasificare, asigurându-se că variabilele sunt corect codificate, facilitând o predicție cât mai precisă.

Așadar, scopul acestui proiect se reduce la a previziona pentru cele 3459 restaurante din setul de test, pe baza evidențelor din setul de antrenare, categoria de clasificare după ghidul Michelin, codificată sub denumirea de Award.

```
test_restaurants
```

	Restaurant Name	Location	True Award
0	Grand Cru	Prague, Czech Republic	Selected Restaurants
1	Wirtshaus Meyers Keller	Nördlingen, Germany	1 Star
2	Hammer & Pincers	Wymeswold, United Kingdom	Selected Restaurants
3	Kitchen Table	London, United Kingdom	2 Stars
4	Roe	London, United Kingdom	Selected Restaurants
...	...	...	...
3454	Bootshaus	Weissenhaus, Germany	Selected Restaurants
3455	Summer Pavilion	Singapore	1 Star
3456	Liza	Paris, France	Selected Restaurants
3457	Bún Bò Bà Rơi (Hai Chau)	Da Nang, Vietnam	Bib Gourmand
3458	Das Fetzwerk	Oberstdorf, Germany	Bib Gourmand

3459 rows × 3 columns

Fig. 9 Lista restaurantelor din setul de testare - prelucrare Python

III.APLICAREA METODELOR

În acest capitol dedicat **aplicării celor trei metode** de clasificare pentru predicția premiilor Michelin, sunt acoperiți pașii necesari pentru antrenarea, testarea și evaluarea modelelor de clasificare selectate pe baza unor metrici de performanță.

III.1 Random Forest

Random Forest este un algoritm de tip *ensemble*, care folosește o colecție (numită și „pădure”) de arbori de decizie pentru a obține clasificări precise și robuste. Fiecare arbore este antrenat pe un subset aleatoriu al datelor, iar predicția finală a modelului este dată de votul majoritar al tuturor arborilor din pădure.

```
# Random Forest
rf_model = RandomForestClassifier(n_estimators=10, random_state=42)
rf_model.fit(X_train, y_train)
y_pred_rf = rf_model.predict(X_test)
```

Fig. 10 Implementare model Random Forest - prelucrare Python

Descriere parametrii ai funcției de activare:

- **n_estimators=10**: specifică numărul de arbori din „pădurea” de decizie. În acest caz, înseamnă că vor fi create 10 arbori de decizie independenți.
- **random_state=42**: controlează generatorul de numere aleatoare folosit pentru a crea variații între arbori, asigură că modelul va avea aceleași rezultate de fiecare dată când rulează algoritmul.

Pentru fiecare restaurant din setul de tastare, se creează pe baza variabilelor independente, o serie de probabilități pentru ca acel restaurant să fie încadrat într-o anumită categorie Michelin. Astfel că, acolo unde probabilitatea este cea mai apropiată de 1, în acea categorie va fi previzionată respectiva unitate.

	RF_Prob_0	RF_Prob_1	RF_Prob_2	RF_Prob_3	RF_Prob_4	RF_Pred_Class
0	0.000	0.000	0.0	0.000000	1.000000	4
1	0.000	0.000	0.0	0.100000	0.900000	4
2	0.150	0.000	0.0	0.000000	0.850000	4
3	0.350	0.475	0.0	0.000000	0.175000	1
4	0.100	0.000	0.0	0.000000	0.900000	4
...	...	...	...	...	...	...
3454	0.000	0.000	0.0	0.100000	0.900000	4
3455	0.425	0.000	0.0	0.000000	0.575000	4
3456	0.100	0.000	0.0	0.200000	0.700000	4
3457	0.000	0.000	0.0	0.364307	0.635693	4
3458	0.000	0.000	0.0	0.000000	1.000000	4

3459 rows × 6 columns

Fig. 11 Tabel probabilități de previziune metoda Random Forest - prelucrare Python

Ulterior, în urma rulării algoritmului Random Forest obținut predicția categoriei Michelin restaurantelor din setul de test, după cum este prezentat în Matricea de confuzie. Dintre cele 3459 de restaurante din setul de testare, 2258 au fost încadrate în clasa corectă.

```
restaurants in testset: 3459

restaurants with Award accuratelly predicted with Random Forest:2258
```

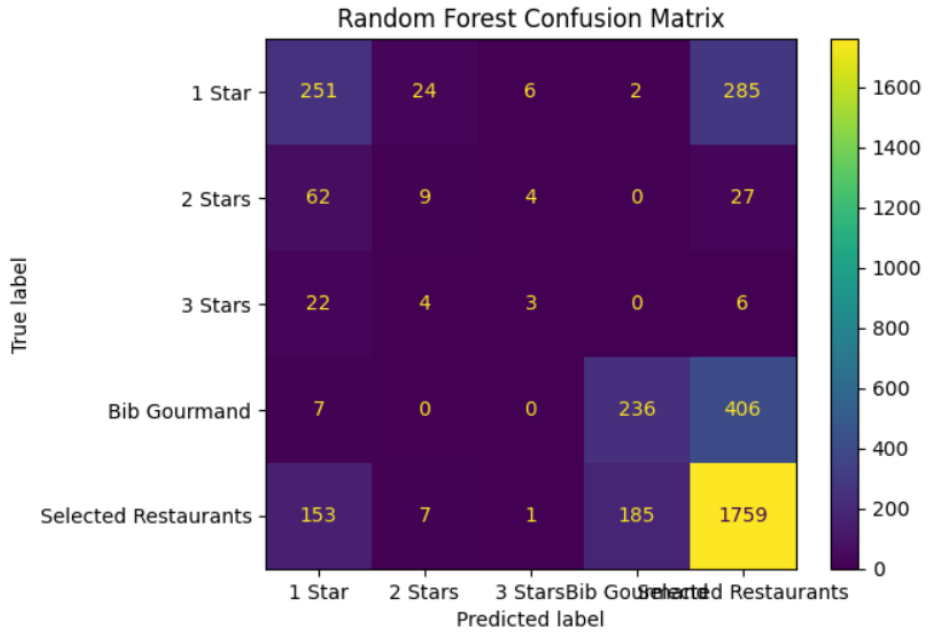


Fig. 12 Matricea de confuzie previziune Random Forest - prelucrare Python

În etapa finală, nu rămâne decât evaluarea performanței prin metricele specifice:

```
Model: Random Forest
Accuracy: 0.6528
Precision: 0.6273
Recall: 0.6528
F1-score: 0.6317
```

Fig. 13 Performanța previziunii Random Forest - prelucrare Python

III.2 Logistic Regression

Regresia Logistică este un model liniar de clasificare care prezice probabilitatea de apartenență la o clasă, bazându-se pe relațiile liniare dintre variabilele predictive și clasa țintă. Deși a fost dezvoltată inițial pentru clasificare binară, se poate adapta și la probleme multi-clasă, cum este cazul restaurantelor din ghidul Michelin. Într-o clasificare multi-clasă, sunt calculate probabilitățile fiecărei clase, iar clasa cu probabilitatea cea mai mare devine predicția finală.

```python
# Logistic Regression
lr_model = LogisticRegression(multi_class='ovr', solver='liblinear')
lr_model.fit(X_train, y_train)
y_pred_lr = lr_model.predict(X_test)
```

Fig. 14 Implementare model Logistic Regression - prelucrare Python

Descriere parametrii ai funcției de activare:

- **multi_class='ovr' (one-vs-rest)**:specifică abordarea **„unul vs retul”** pentru clasificarea multi-clasă,modelul antrenează câte un clasificator pentru fiecare clasă în parte, tratând fiecare clasă ca o problemă binară, fiecare clasificator produce o probabilitate pentru clasa sa specifică, iar clasa cu cea mai mare probabilitate este aleasă ca predicție finală.
- **solver='liblinear'**:algoritm de optimizare ce funcționează bine în clasificarea binară sau multiclasă.

În mod similar ca și pentru modelul de Random forest, au fost urmărite aceleași etape în dezvoltarea modelului:

1.Tabel probabilități estimare tip clasă:

	LR_Prob_0	LR_Prob_1	LR_Prob_2	LR_Prob_3	LR_Prob_4
0	0.023597	0.001773	0.001001	0.079907	0.893722
1	0.220670	0.012429	0.004863	0.029314	0.732725
2	0.132379	0.002930	0.001281	0.020466	0.842945
3	0.516550	0.056718	0.003098	0.001856	0.421779
4	0.032843	0.001648	0.001262	0.183056	0.781192
...	...	...	...	...	...
3454	0.006212	0.000897	0.000260	0.230495	0.762136
3455	0.201281	0.025868	0.012997	0.250101	0.509754
3456	0.010440	0.000877	0.003572	0.161274	0.823838
3457	0.001526	0.000030	0.000057	0.740738	0.257648
3458	0.013298	0.000686	0.000302	0.230459	0.755255

3459 rows × 5 columns

Fig. 15 Tabel probabilități de previziune metoda Logistic Regression - prelucrare Python

Comparativ cu predicția modelului Random Forest, pare că estimarea prin regresia logistică să aibă o acuratețe mai mare, astfel că 2304 din cele 3459 de restaurante din setul de testare au fost corect reprezentate.

```
restaurants in testset: 3459

restaurants with Award accuratelly predicted with Logistic Regression:2304
```

2.Reprezentare Matrice de confuzie:

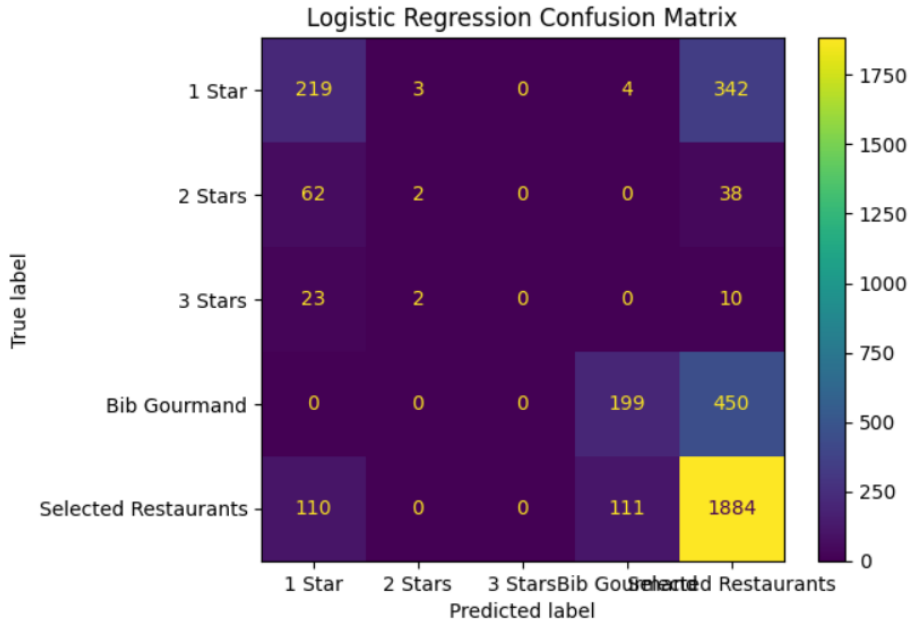


Fig. 16 Matricea de confuzie previziune Logistic Regression - prelucrare Python

3.Evaluarea performanței modelului de regresie:

```
Model: Logistic Regression
Accuracy: 0.6661
Precision: 0.6351
Recall: 0.6661
F1-score: 0.6267
```

Fig. 17 Performanța previziunii Logistic Regression - prelucrare Python

III.3 Naive Bayes

Naive Bayes este un model probabilistic de clasificare care se bazează pe aplicarea teoremei lui Bayes, presupunând că toate variabilele predictive sunt independente una de cealaltă (ipoteza „naivă"). Acesta estimează probabilitatea ca o observație să aparțină unei anumite clase, pe baza probabilităților condiționate ale fiecărei variabile.Clasa cu cea mai mare probabilitate a posteriori este atribuită observației respective.

```python
from sklearn.naive_bayes import GaussianNB
# Gaussian Naive Bayes
nb_model = GaussianNB()  # Initialize Naive Bayes model
nb_model.fit(X_train, y_train)
y_pred_nb = nb_model.predict(X_test)
```

Fig. 18 Implementare model Naive Bayes - prelucrare Python

Descriere parametrii ai funcției de activare:

- GaussianNB() : presupune că fiecare variabilă urmează o distribuție normală în fiecare clasă (premiu Michelin), modelul este rapid și eficient, și atribuie o probabilitate pentru fiecare clasă, ceea ce este util în interpretarea predicțiilor pentru distincțiile Michelin.

Ca și în cazul celorlalte tehnici, am urmat aceeași pași menționați anterior:

1.Tabel probabilități estimare tip clasă:

	NB_Prob_0	NB_Prob_1	NB_Prob_2	NB_Prob_3	NB_Prob_4	NB_Pred_Class
0	0.000000e+00	0.0	0.0	1.0	0.0	3
1	2.913715e-23	0.0	0.0	1.0	0.0	3
2	0.000000e+00	1.0	0.0	0.0	0.0	1
3	0.000000e+00	0.0	1.0	0.0	0.0	2
4	0.000000e+00	1.0	0.0	0.0	0.0	1
...	...	...	...	...	...	...
3454	0.000000e+00	1.0	0.0	0.0	0.0	1
3455	0.000000e+00	0.0	1.0	0.0	0.0	2
3456	0.000000e+00	0.0	0.0	1.0	0.0	3
3457	0.000000e+00	0.0	0.0	1.0	0.0	3
3458	0.000000e+00	0.0	0.0	1.0	0.0	3

3459 rows × 6 columns

Fig. 19 Tabel probabilități de previziune metoda Naive Bayes - prelucrare Python

Modelul antrenat de Naive Bayes pare să nu fie tocmai potrivit setului de date și criteriilor din ipoteză, astfel că doar 715 restaurante din testul de set au fost corect încadrate.

```
restaurants in testset: 3459

restaurants with Award accuratelly predicted with Naive Bayes:715
```

2.Reprezentare Matrice de confuzie:

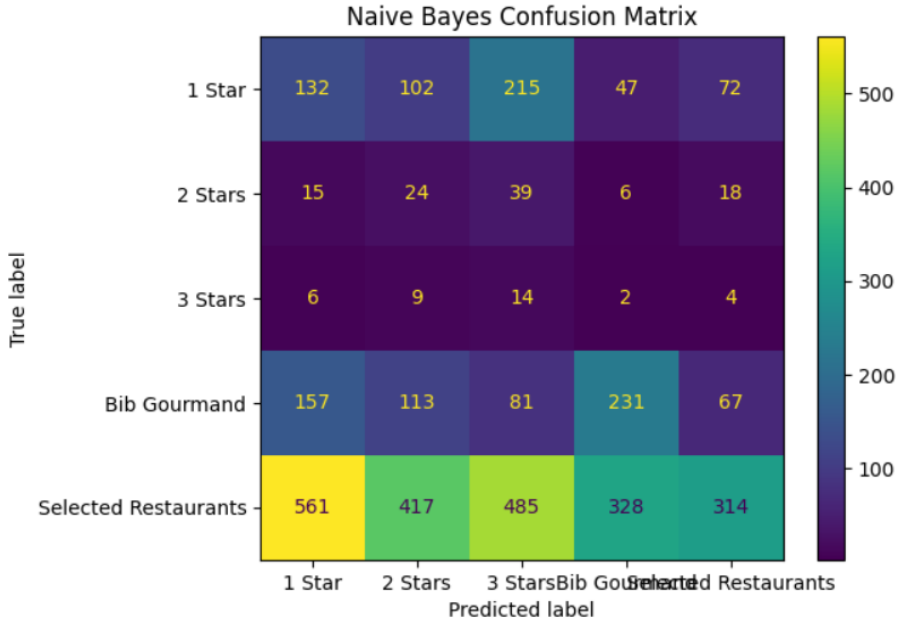


Fig. 20 Matricea de confuzie previziune Naive Bayes - prelucrare Python

3.Evaluarea performanței modelului clasificatorului Naive Bayes:

```
Model: Naive Bayes
Accuracy: 0.2067
Precision: 0.4990
Recall: 0.2067
F1-score: 0.2491
```

Fig. 21 Performanța previziunii Naive Bayes - prelucrare Python

IV.INTERPRETAREA REZULTATELOR

În acest capitol am ales a reprezenta **Compararea finală** a celor trei modele, prin care voi analiza care metodă pare a fi cea mai potrivită pentru clasificarea premiilor Michelin, prin analiza performanței pe baza metodelor de evaluare (acurateță, precizie, scor F1, matricea de confuzie).

Fiecare algoritm este evaluat pe baza unor metrici de performanță, definite astfel:
Acuratețea: Procentul de restaurante corect clasificate în test.
Precizia și scorul F1: Metrici care arată calitatea clasificării pentru fiecare clasă.
Matricea de confuzie: Imagine asupra clasificărilor corecte și greșite pentru fiecare tip.

Agregând rezultatele obținute prin cele 3 metode, am realizat o reprezentare tabelară.

	Model	Accuracy	Precision	Recall	F1-score
0	Random Forest	0.652790	0.627259	0.652790	0.631665
1	Logistic Regression	0.666088	0.635096	0.666088	0.626717
2	Gaussian Naive Bayes	0.206707	0.498998	0.206707	0.249060

Fig. 22 Tabel acurateţe predicţii - prelucrare Python

Considerând ca și criteriu de alegere valoarea maximă, cât mai aproape de 1, se poate observa că Random Forest și Regresia Logistică par ar fi optime pentru clasificarea restaurantelor din studiul de caz.

Pentru Random Forest, Acuratețea (Accuracy) de 0.6528 indică faptul că Random Forest reușește să clasifice corect aproximativ 65% din restaurantele din setul de testare.Precizia (Precision) de 0.6273 sugerează că modelul a identificat relativ bine premiile.Revocarea (Recall) de 0.6528 și scorul F1 de 0.6317 indică un echilibru între identificarea corectă a claselor și gestionarea clasificărilor greșite.

În cazul regresiei logistice, Acuratețea de 0.6661 arată că modelul a avut performanțe ușor mai bune decât Random Forest în a clasifica restaurantele, corectând aproximativ 67% din predicții. Precizia de 0.6351 și revocarea de 0.6661 sugerează că modelul a reușit să gestioneze destul de bine distribuția claselor, având o acuratețe mai bună decât Random Forest. Scorul F1 este puțin mai scăzut, la 0.6267, ceea ce sugerează că deși modelul poate avea o acuratețe mai bună, poate exista o ușoară instabilitate în modul în care clasifică corect unele clase. În general, modelul de regresie logistică s-a dovedit a fi cel mai performant dintre cele trei în acest caz.

În ceea ce privește clasificatorul Naïve Bayes, doar 20% dintre restaurantele din setul de test au fost previzionate în categoria lor corectă de clasă Michelin.

O potențială explicație este aceea că variabilele explicative nu respectă întocmai ipoteze unei distribuții normale Gaussiene. Precizia de 0.4990 și Recall de 0.2067 arată o slabă capacitate a modelului de predicție, ceea ce sugerează că ipoteza de independență a variabilelor și ipoteza de distribuție normală nu se potrivesc bine cu acest set de date.

O performanță mai slabă a clasificatorului Naive Bayes se poate observa și din matricea de confuzie specifică, unde se vede o dispersie clară asupra restaurantelor corect vs fals identificate.

V.CONCLUZII

Dintre cele trei modele, Regresia Logistică a oferit cele mai bune rezultate în clasificarea premiilor Michelin, având acurateţe și scoruri de evaluare echilibrate. Random Forest este aproape la același nivel și poate fi, de asemenea, o alegere bună, mai ales dacă se ajustează numărul de arbori sau alți parametri. Naive Bayes Gaussian nu s-a potrivit bine pentru această problemă și ar putea fi înlocuit cu alte metode de clasificare specifice pentru seturile de date cu variabile corelate și distribuite non-Gaussian.

Ca și etapă exploratorie, am agregat rezultatele celor 3 modele, pentru a vedea cum acestea lucrează împreună.

```
restaurants with Award accurately predicted with all 3 methods:408

Accurately Predicted Values with all 3 methods:
```

	Restaurant Name	Location	True Award	RF Prediction	LR Prediction	NB Prediction
24	Lake Road Kitchen	Ambleside, United Kingdom	1 Star	1 Star	1 Star	1 Star
71	ROM	Roses, Spain	Selected Restaurants	Selected Restaurants	Selected Restaurants	Selected Restaurants
78	Hudson Smokehouse	Bronx, USA	Selected Restaurants	Selected Restaurants	Selected Restaurants	Selected Restaurants
87	Brasserie du Château	Bottmingen, Switzerland	Selected Restaurants	Selected Restaurants	Selected Restaurants	Selected Restaurants
127	Cristó	Zierikzee, Netherlands	Selected Restaurants	Selected Restaurants	Selected Restaurants	Selected Restaurants
...	...	...	...	...	...	...
3408	Forde	Horsforth, United Kingdom	Selected Restaurants	Selected Restaurants	Selected Restaurants	Selected Restaurants
3427	Nils	Koksijde, Belgium	Selected Restaurants	Selected Restaurants	Selected Restaurants	Selected Restaurants
3430	Vecchia Malcesine	Malcesine, Italy	1 Star	1 Star	1 Star	1 Star
3441	Anetseder	Hauzenberg, Germany	Bib Gourmand	Bib Gourmand	Bib Gourmand	Bib Gourmand
3444	Culinair	Lauwe, Belgium	Selected Restaurants	Selected Restaurants	Selected Restaurants	Selected Restaurants

Fig. 23 Lista restaurante previzionate corect de toate 3 modelele - prelucrare Python

Dintre cele 3459 de restaurante din setul de test, doar 408 au fost correct încadrate în clasa Michelin de toate cele 3 metode, puțin peste 10%. Astfel, am concluzionat că fiecare dintre cele 3 modele ține cont individual de variabilele independente ale modelului.

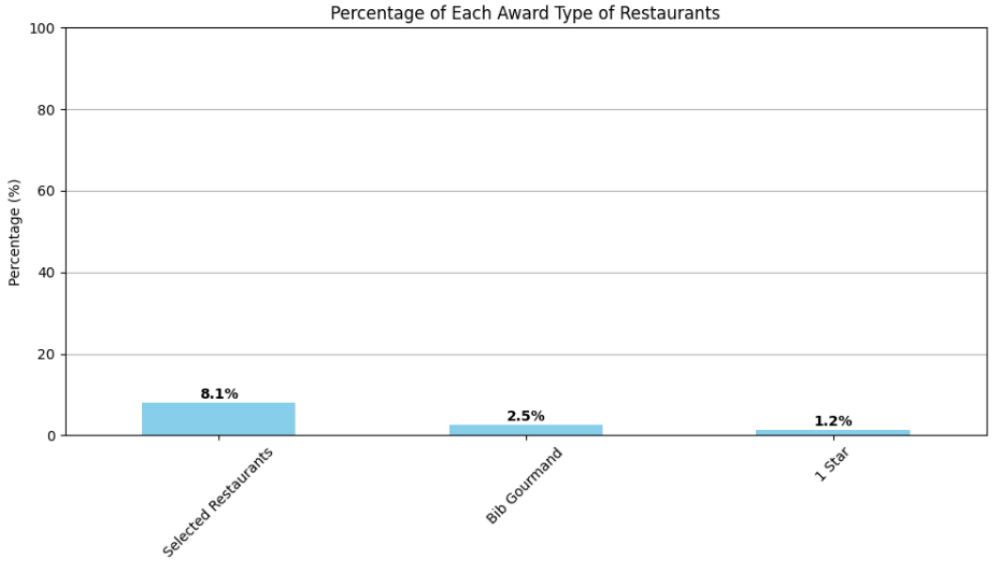


Fig. 24 Distribuția în funcție de distincția Michelin - prelucrare Python

Astfel, cele mai ușor de previzionat, au fost, în mod previzibil, restaurantele fără stele Michelin, ci mai degrabă cele aflate pe lista Selected sau Bib Gourmand.

Modelele au arătat că variabilele precum prețul, locația și bucătăria sunt semnificative în determinarea tipului de premiu Michelin, iar variabilele legate de facilități pot oferi informații suplimentare.

Așadar, așa cum am specificat și în ipoteză, clasificarea restaurantelor din ghidul Michelin poate fi o adevărată provocare chiar și pentru domeniul de analiza datelor.

Pentru o mai bună clasificare, ar putea fi utile mai multe tehnici și/sau testarea altor modele de clasificare. Pe lângă variabilele disponibile, integrarea unor noi variabile, cum ar fi recenziile și scorurile din mediul online, ar putea oferi un plus de informații privind preferințele utilizatorilor și aprecierea restaurantului, contribuind la o clasificare mai exactă.

Prin acest studiu de caz am demonstrat la nivel aplicativ că este posibilă predicția tipului de premiu Michelin pe baza caracteristicilor restaurantelor. În contextul dat, Regresia Logistică și Random Forest par a fi modele robuste pentru această sarcină, însă integrarea mai multor date ți tehnici ar putea îmbunătăți acuratețea predicției în studiile viitoare.

VI. BIBLIOGRAFIE

[1] Awotunde, J. B., Misra, S., Katta, V., & Adebayo, O. C. (2023). An Ensemble-Based Hotel Reviews System Using Naive Bayes Classifier.

[2] Ganesh, C & Kesavulu Reddy, Dr. (2022). Overview of the Predictive Data Mining Techniques. INTERNATIONAL JOURNAL OF COMPUTER SCIENCES AND ENGINEERING. 10. 30. 10.26438/ijcse/v10i1.2836.

[3] Jiang, H., Mei, L., Wei, Y., Zheng, R., & Guo, Y. (2022). The influence of the neighbourhood environment on peer-to-peer accommodations: A random forest regression analysis. *Journal of Hospitality and Tourism Management*, *51*, 105-118.

[4] Larasati, A., DeYong, C., & Slevitch, L. (2012). The application of neural network and logistics regression models on predicting customer satisfaction in a student-operated restaurant. *Procedia-Social and Behavioral Sciences*, *65*, 94-99.

[5] Patil, R. N., Singh, Y. P., Rawandale, S. A., & Singh, S. (2024). Improving Sentiment Classification on Restaurant Reviews Using Deep Learning Models. *Procedia Computer*

[6] Shaik, A. B., & Srinivasan, S. (2019). A brief survey on random forest ensembles in classification model. In *International Conference on Innovative Computing and Communications: Proceedings of ICICC 2018, Volume 2* (pp. 253-260). Springer Singapore.

[7] Speiser, J. L., Miller, M. E., Tooze, J., & Ip, E. (2019). A comparison of random forest variable selection methods for classification prediction modeling. *Expert systems with applications*, *134*, 93-101.

[8] Suport curs & seminar Data Mining – online.ase.ro

[9] Yang, F. J. (2018, December). An implementation of naive bayes classifier. In *2018 International conference on computational science and computational intelligence (CSCI)* (pp. 301-306). IEEE.

[10] Zou, X., Hu, Y., Tian, Z., & Shen, K. (2019, October). Logistic regression model optimization and case analysis. In *2019 IEEE 7th international conference on computer science and network technology (ICCSNT)* (pp. 135-139). IEEE.