Ion Andrei – Cristian GRUPA 313CC

Documentație

PARTEA I

În acest proiect încerc să prezic valoarea totală a produselor returnate pentru un anumit brand, pe baza următoarelor criterii:

1. Denumirea brandului
2. Țara în care s-au vândut produsele
3. Vânzări brute (înainte de reduceri sau taxe)
4. Vânzări nete (după reduceri sau taxe)
5. Vânzări totale
6. Suma cheltuită pe reclame pentru promovarea produsului
7. Numărul de comenziâ

Datasetul l-am luat de pe Kaggle. Prima data am adăugat zgomot pe coloanal Vânzări brute.

Am făcut mai întâi EDA pe datele nemodificate, adică nestandardizate sau codificate.

a) Valori lipsă

Valori lipsa (train):

	Missing Values	%
Brand Name	0	0.0
Country	0	0.0
Gross Sales	0	0.0
Net Sales	0	0.0
Total Sales	0	0.0
Total Ad Spend	0	0.0
Order Count	0	0.0
Return Amount	0	0.0

Nu avem nicio valoare lipsă în dataset. Acest lucrum mă duce cu gândul la faptul că lipsesc eventuale probleme reale care apar în date de producție cum ar fi pachetele pierdute. Nu aplic imputare sau ștergere, deoarece nu am valori lipsă și toate valorile sunt relevate.

b) Statistici descriptive

Am analizat statisticile numerice din setul de date de train și de test de asemenea.

Set de date train

Statistici descriptive (train - numerice):

	Gross Sales	Net Sales	Total Sales	Total Ad Spend	Order Count	Return Amount
count	1340.00000	1340.000000	1340.000000	1340.000000	1340.000000	1340.000000
mean	44722.90706	38217.170036	40264.810075	6760.718267	5.573881	10004.861940
std	46189.90789	41376.716577	44594.387400	3510.861397	4.146828	6290.914017
min	32.00000	-212732.354825	-222364.800000	0.000000	0.000000	18.000000
25%	19584.75000	15691.539614	15987.000000	4826.238138	2.000000	4725.500000
50%	33245.50000	26899.368274	27877.000000	6459.750000	5.000000	9980.500000
75%	47821.25000	39733.818395	39685.250000	8065.000000	8.000000	15407.000000
max	741965.00000	397189.911575	437840.000000	24675.816400	30.000000	84765.000000

Observ că unele braduri au pierderi masive întrucât vând foarte puțin și după taxe ies foarte mult pe minus. Depine și de țară întrucât unele țări au taxe foarte mari.

Observ de asemenea că în media Total Sales este mai mare decât cea Net Sales, ceea ce s-ar putea explica prin faptul că la Total Sales se mai adună și alte valori cum ar fi banii pe transport. Din totalul de vânzări se returnează cam un sfert ceea ce este plauzibil, depinde foarte mult și de ce produs este vândut.

Set de date test

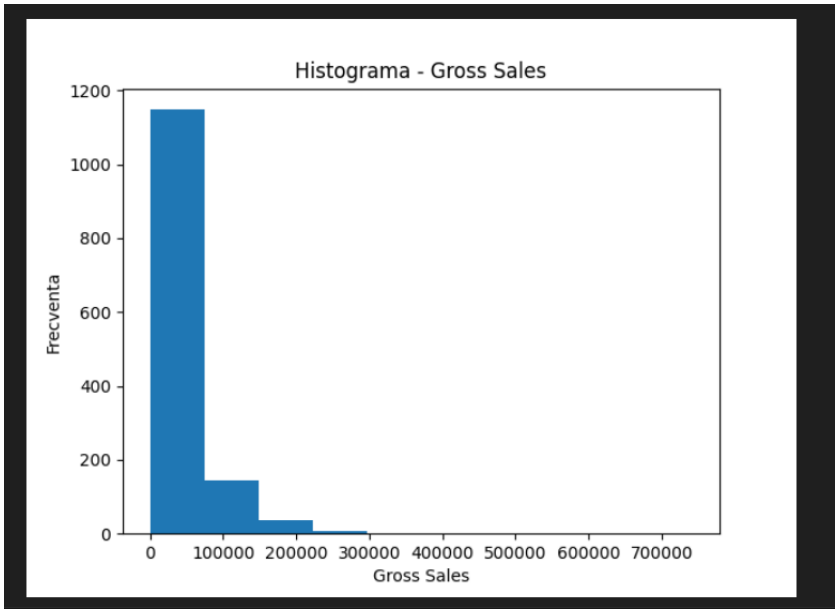
Statistici descriptive (test - numerice):

	Gross Sales	Net Sales	Total Sales	Total Ad Spend	Order Count	Return Amount
count	336.000000	336.000000	336.000000	336.000000	336.000000	336.000000
mean	43154.241190	35953.467036	38003.248899	6736.045255	5.657738	10314.211310
std	36528.687857	44822.390670	47023.421372	3238.502107	3.760613	5832.243924
min	2099.000000	-445963.490883	-445742.260000	0.000000	1.000000	81.000000
25%	20040.500000	16816.835074	17233.750000	4797.351425	2.750000	5114.000000
50%	33323.500000	28309.778701	29052.000000	6465.000000	5.000000	10208.000000
75%	47590.000000	40125.006318	41450.000000	8190.750000	8.000000	15780.000000
max	201630.710000	184636.903684	200208.000000	24876.327200	21.000000	19996.000000

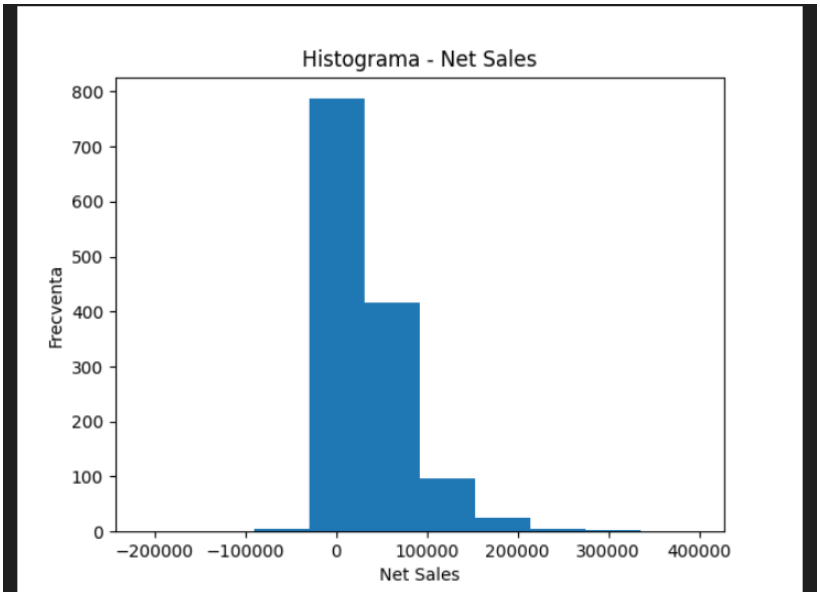
În setul de date test se observă în mare parte aceleași lucruri ca în setul de date train. Gross Sales maxim este mult mai mic decât la setul de date train.

3)Histograme

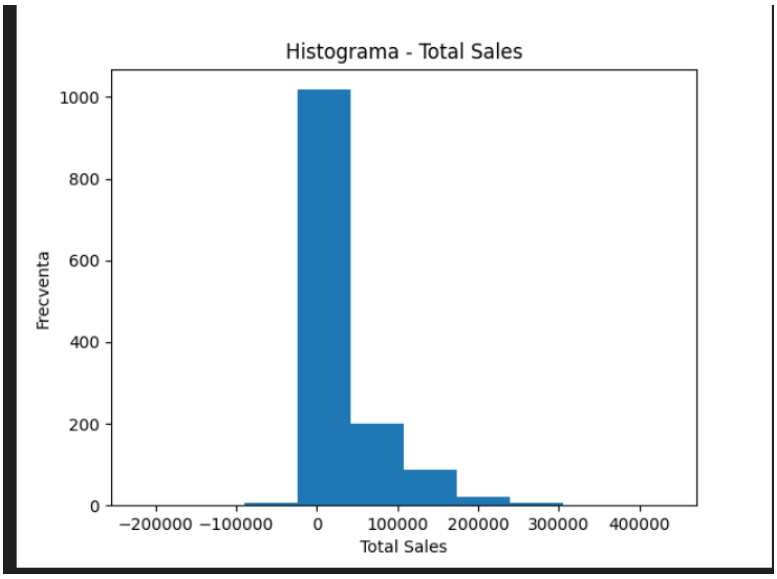
Majoritatea valorilor pentru Gross Sales sunt situate în intervalul 0–100.000, cu o frecvență foarte mare în primele intervale. Avem și câteva vârfuri mari, chiar peste 700.000, dar acestea sunt rare. Distribuția este puternic asimetrică spre dreapta, ceea ce sugerează că cele mai multe branduri înregistrează vânzări brute modeste, în timp ce doar câteva au vânzări excepționale. Această variație extremă poate influența modelul și poate necesita tratamente speciale, cum ar fi transformări logaritmice sau tratarea outlierilor, pentru a evita efecte disproporționate în predicții.



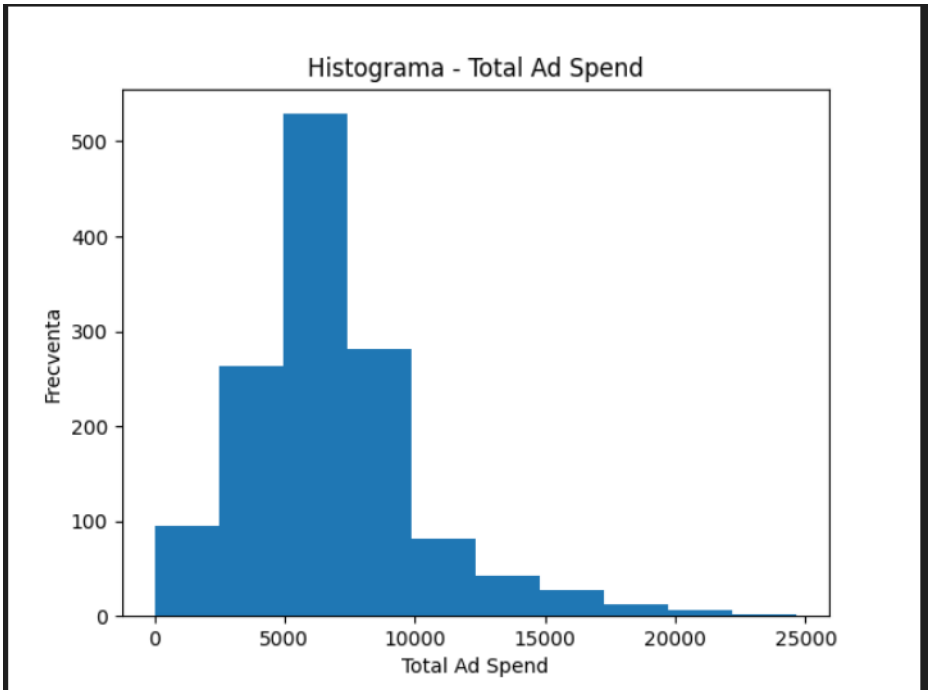
La fel și aici, observăm că de foarte puține ori s-au înregistrat pierderi și în rest s-a obținut un profit asemenător în majoritatea cazurilor.



Se vede clar că majoritatea valorilor pentru Total Sales sunt concentrate aproape de zero, iar frecvența scade rapid pe măsură ce ne îndepărtăm. Avem și câteva valori foarte mari (până spre 400.000) și chiar unele negative, dar ele sunt rare. Distribuția nu este echilibrată și are o coadă lungă spre dreapta, ceea ce înseamnă că avem câteva observații care ies mult din tipar. Acest lucru poate influența modelul de predicție, pentru că aceste valori extreme pot trage rezultatele într-o direcție greșită dacă nu sunt tratate corect.

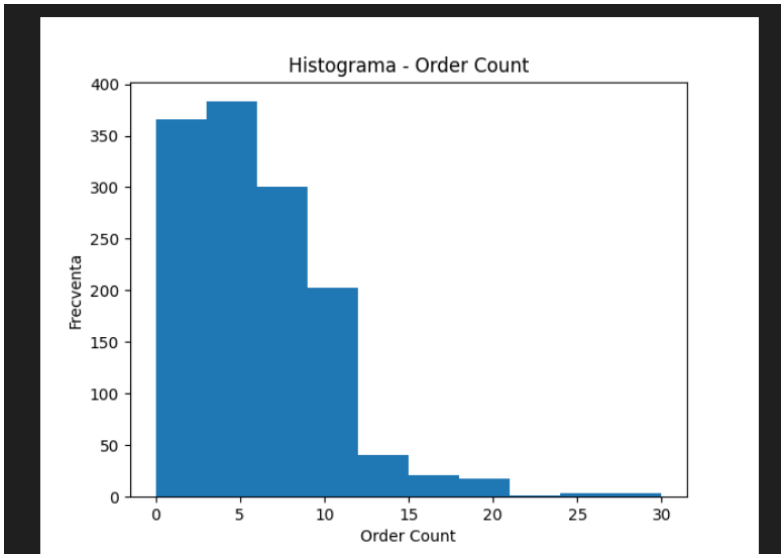


Se observă că cele mai multe valori pentru Total Ad Spend se află între 4000 și 9000, cu un maxim în jur de 6000. Avem și câteva valori mari, dar acestea sunt rare. Asta înseamnă că majoritatea companiilor au bugete de publicitate similare, ceea ce poate limita modelul în a învăța cum se comportă datele în situații foarte diferite. Cu cât avem o diversitate mai mare în date (mai multe niveluri de cheltuieli), cu atât modelul ar putea face predicții mai bune.

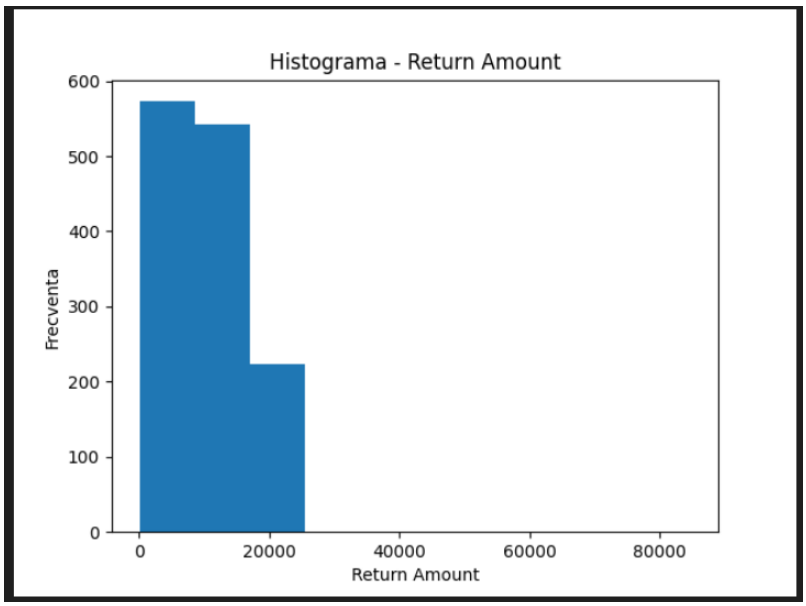


Se observă că majoritatea valorilor pentru Order Count sunt între 1 și 10, cu un maxim în jur de 4–5 comenzi. Există și câteva situații în care numărul de comenzi ajunge la 20 sau chiar 30, dar acestea sunt rare. Distribuția nu este simetrică, cu o concentrație mare în intervalele joase, ceea ce indică faptul că majoritatea brandurilor au un volum redus de comenzi. Acest lucru poate influența modelul, deoarece învățarea va fi dominată de cazuri cu puține comenzi, iar comportamentele asociate

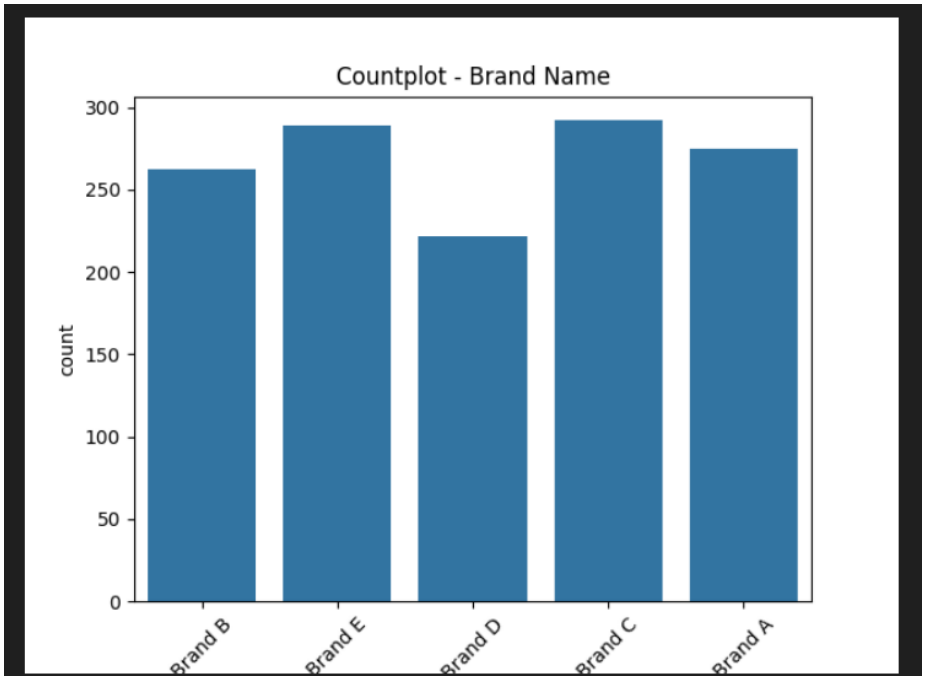
valorilor mari s-ar putea să nu fie învățate corect fără un eșantion mai echilibrat.



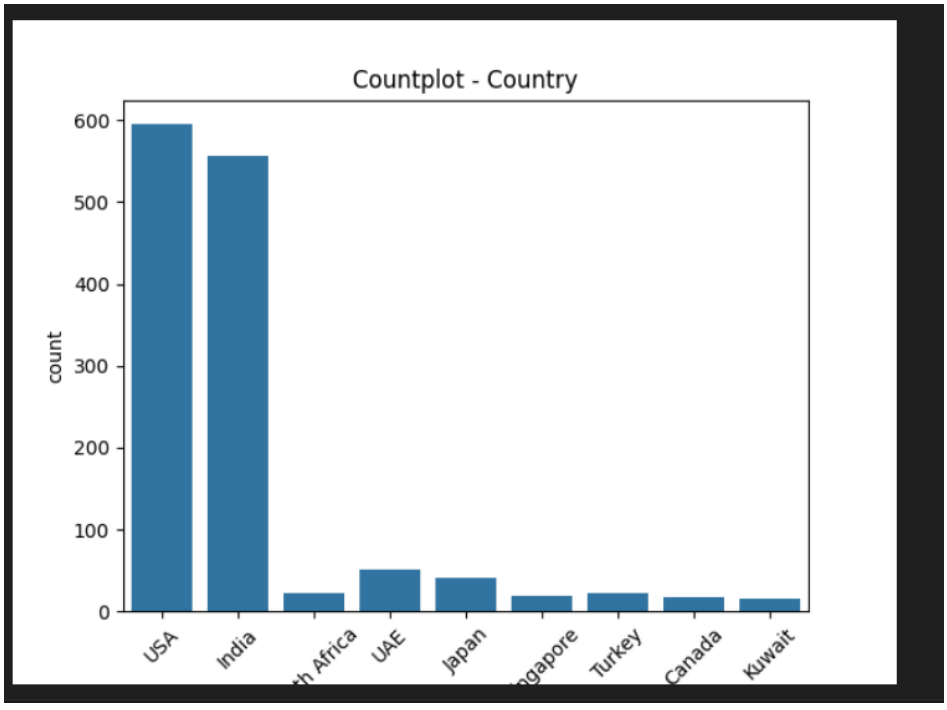
Se observă că majoritatea valorilor pentru Return Amount se află în intervalul 0–20.000, cu cele mai frecvente fiind în jurul valorilor mici. Există câteva valori mult mai mari, care apar rar, ceea ce face ca distribuția să fie asimetrică spre dreapta. Asta înseamnă că, în general, sumele returnate nu sunt foarte mari, dar apar ocazional cazuri izolate cu returnări foarte mari. Aceste extreme pot influența modelul, mai ales dacă nu sunt suficiente exemple similare pentru ca el să învețe corect astfel de situații.



Brandurile sunt împărțite în mod simetric.



Majoritatea vânzărilor au avut loc în USA și India. Faptul că vânzările nu sunt distribuite egal între țări poate reprezenta un motiv pentru o predicție cu o acuratețe mica.



Gross Sales: valorile normale sunt între ~19.585 și ~47.821. Peste 10% din date ies din acest interval, în special prin valori foarte mari.

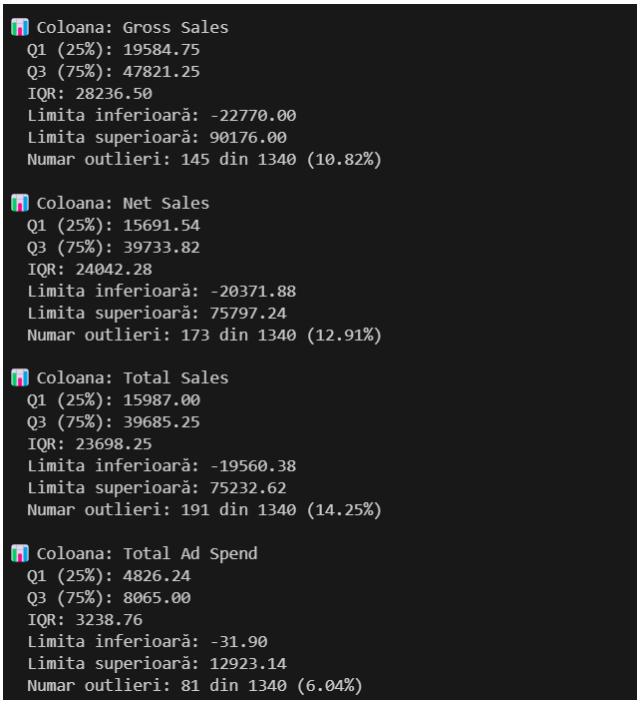
Net Sales: majoritatea vânzărilor nete se încadrează între ~15.692 și ~39.734, dar aproape 13% din valori sunt considerate atipice.

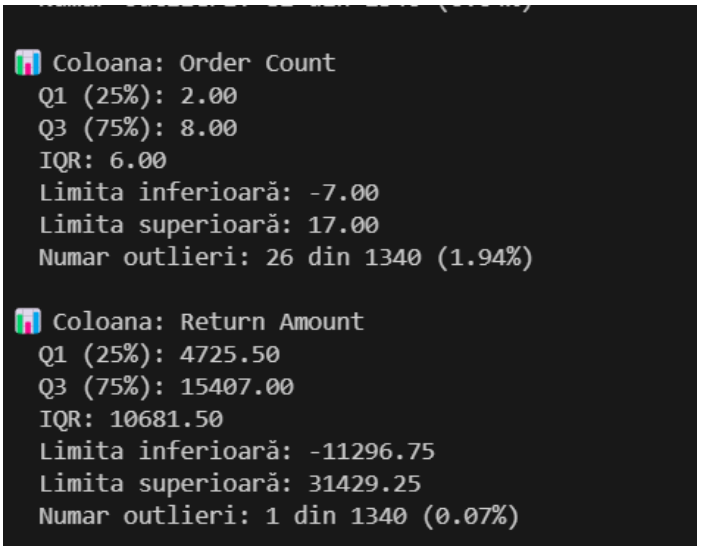
Total Sales: distribuția e similară, cu un interval „normal” între ~15.987 și ~39.685, însă avem chiar mai mulți outlieri (14%).

Total Ad Spend: cele mai multe valori sunt între ~4.826 și ~8.065, iar doar ~6% ies din aceste limite, deci e ceva mai stabilă.

Order Count: majoritatea valorilor sunt între 2 și 8 comenzi. Doar 1.9% ies în afara limitelor, deci distribuția e bine controlată.

Return Amount: valorile returnate sunt în general între ~4.725 și ~15.407. Există un singur outlier în tot setul, ceea ce înseamnă că această variabilă este extrem de stabilă.

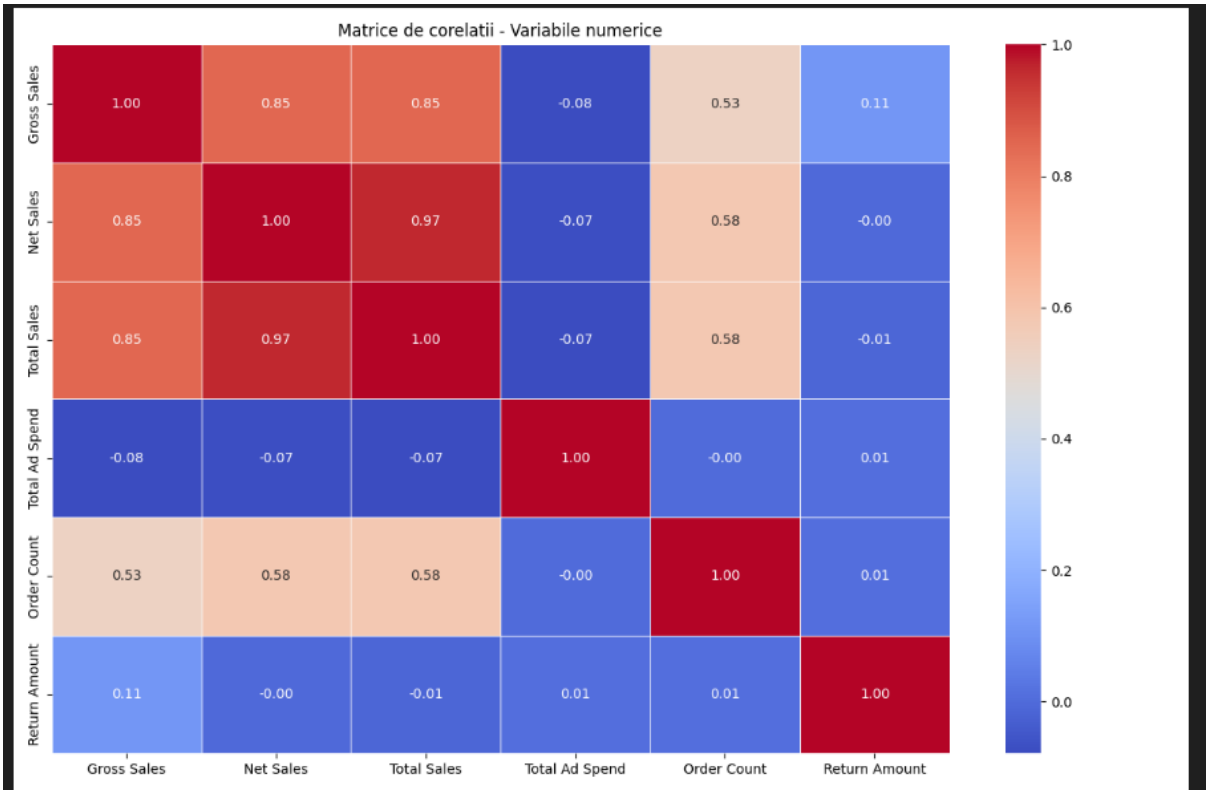




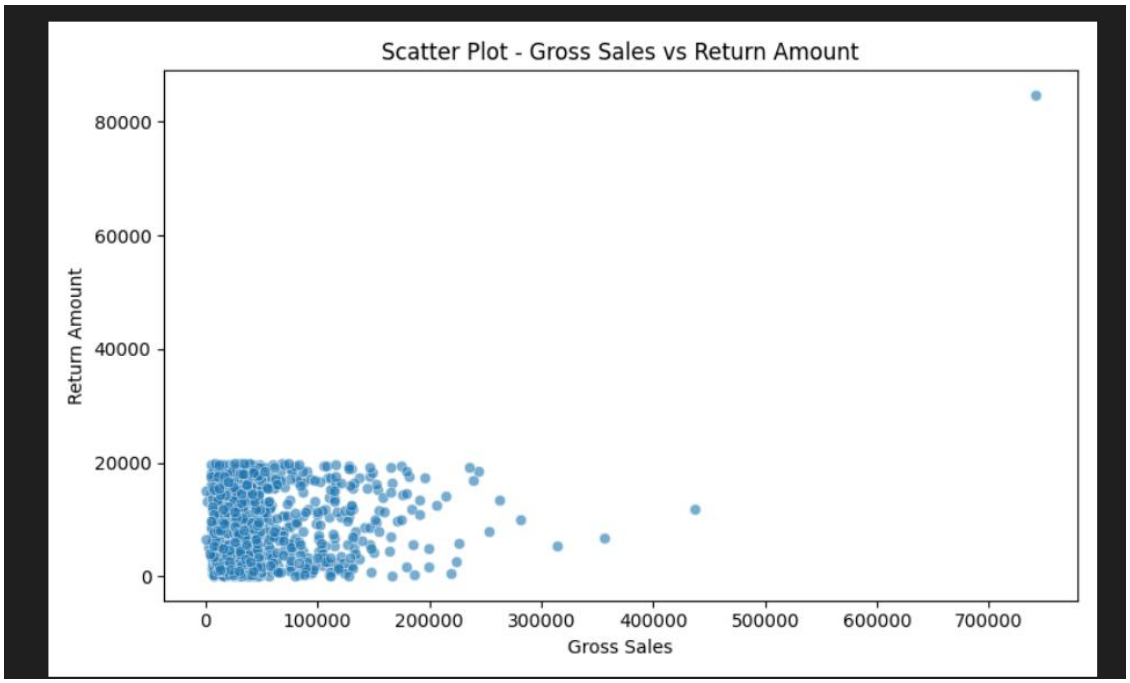
HEATMAP

Din matricea de corelații reiese că Gross Sales, Net Sales și Total Sales sunt foarte strâns legate între ele, având corelații de peste 0.85. Acestea exprimă practic același lucru sub forme diferite, așa că includerea tuturor în model poate fi redundantă. Order Count are o corelație moderată cu vânzările, ceea ce era de așteptat, fiindcă mai multe comenzi duc la vânzări mai mari.

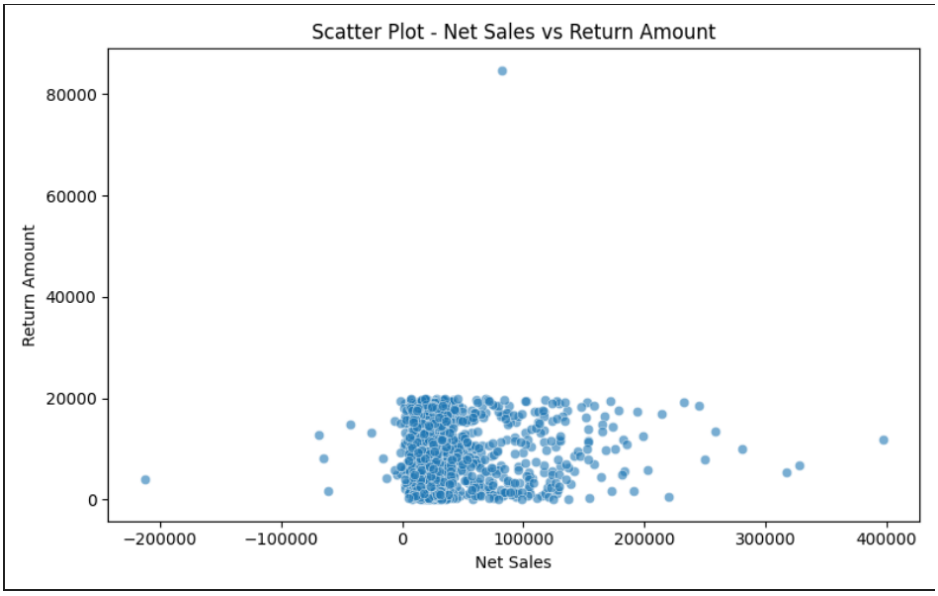
Pe de altă parte, Return Amount nu este corelată semnificativ cu nicio variabilă numerică, ceea ce sugerează că returnările depind de alți factori — probabil categorici sau externi. De asemenea, Total Ad Spend nu influențează vizibil nici vânzările, nici retururile, lucru care poate ridica semne de întrebare privind eficiența cheltuielilor de promovare sau calitatea datelor.



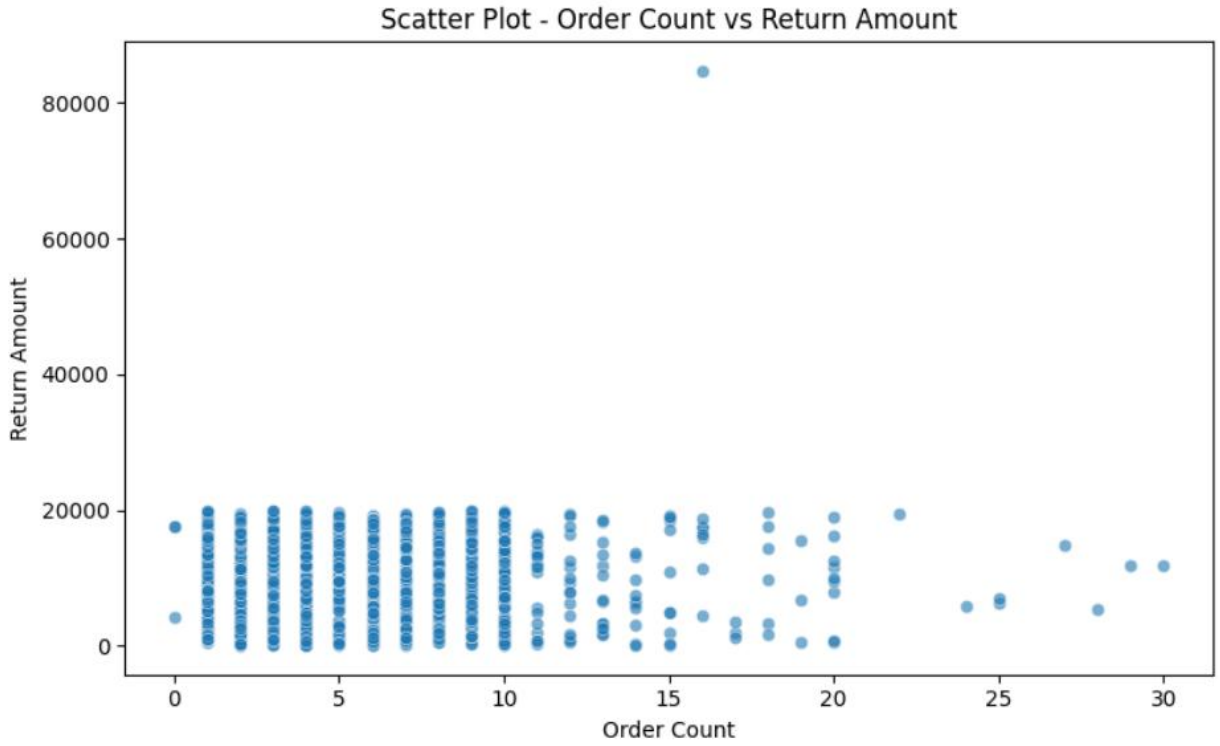
Majoritatea punctelor sunt grupate sub 100.000 la vânzări și sub 20.000 la returnări. Nu se observă o relație clară între Gross Sales și Return Amount. Valori mari de vânzări nu garantează returnări mari. Punctele sunt împrăștiate, deci legătura dintre ele este slabă.



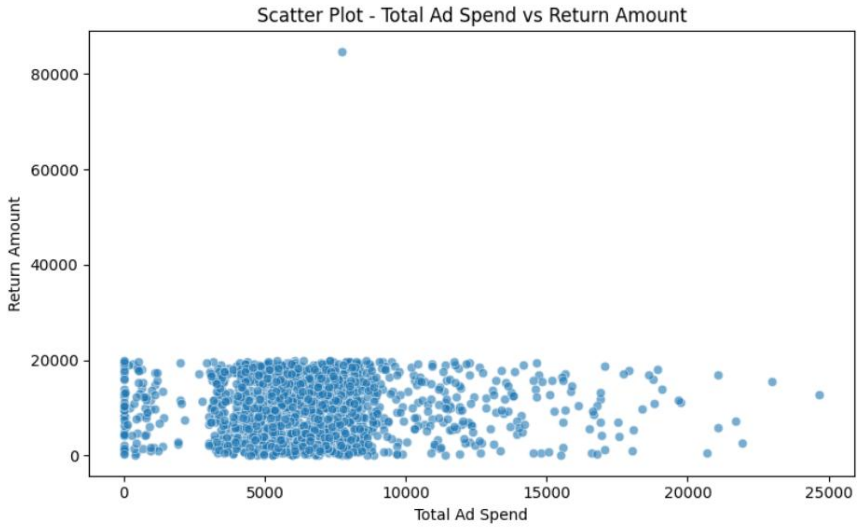
Majoritatea punctelor sunt concentrate în jurul valorilor mici și pozitive. Relația dintre Net Sales și Return Amount este slabă, cu date dispersate și fără o direcție clară. Apar și câteva valori negative sau extreme, dar sunt rare.



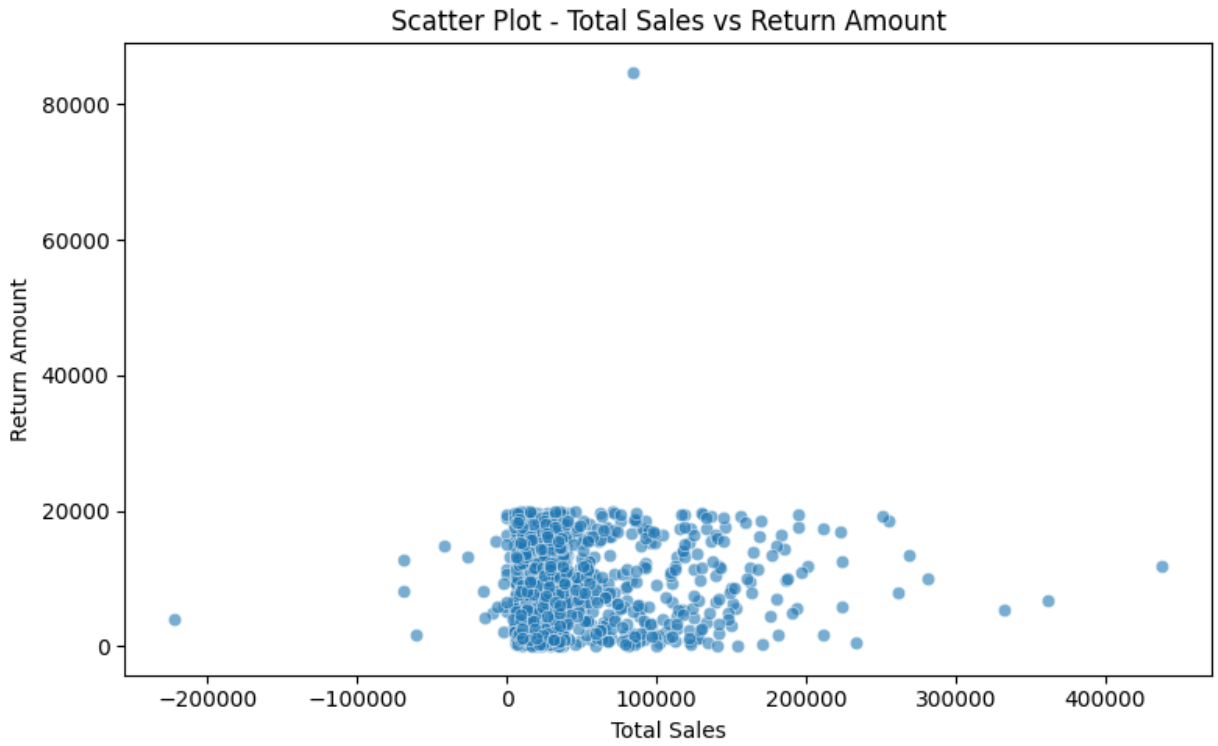
Aici, majoritatea punctelor sunt concentrate între 1 și 15 comenzi, dar Return Amount variază mult indiferent de numărul de comenzi. Nu se observă o relație clară între Order Count și returnări. Punctele sunt răspândite, iar legătura e foarte slabă.



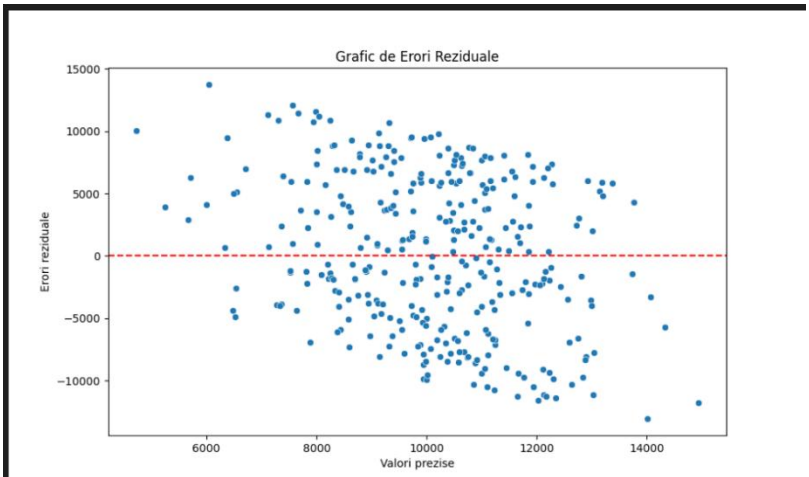
Punctele sunt foarte dispersate și nu indică o relație între Total Ad Spend și Return Amount. Chiar și cu bugete mari, returnările rămân variabile. Nu se observă un tipar clar, deci influența publicității asupra returnărilor pare neglijabilă.



Majoritatea valorilor sunt concentrate aproape de origine, fără o tendință clară. Total Sales nu pare să aibă o relație directă cu Return Amount. Punctele sunt răspândite, iar returnările rămân variabile indiferent de vânzări.



În acest grafic de erori reziduale, punctele sunt răspândite destul de uniform în jurul liniei roșii (care reprezintă eroarea zero), ceea ce e un semn bun. Nu se observă un tipar clar sau o tendință, ceea ce sugerează că modelul nu face erori sistematice. Totuși, există o ușoară variație mai mare a erorilor pentru valorile prezise din zona centrală, dar per total distribuția pare rezonabilă pentru un model ca Random Forest.



EXPLICARE COD (main.py)

Acest cod implementeaza un flux complet de analiza si modelare a datelor pentru a prezice valoarea returnarilor (Return Amount) in functie de mai multe variabile comerciale si de marketing. Initial, sunt importate bibliotecile necesare pentru manipularea datelor, antrenarea modelului si vizualizarea graficelor. Setul de date este incarcat dintr-un fisier CSV folosind un encoding special pentru a interpreta corect caracterele speciale. Se face o corectie in coloana Country, unde abrevierea "SA" este inlocuita cu denumirea completa "South Africa".

Pentru a simula variabilitatea reala, se introduce zgomot aleator de tip gaussian in coloana Net Sales, proportional cu deviata sa standard. Sunt alese coloanele relevante care vor fi folosite ca input (X) si coloana tinta (y) care trebuie prezisa. Datele sunt apoi impartite in seturi de antrenament si test.

Urmeaza o analiza exploratorie a datelor (EDA), unde se afiseaza informatii generale despre set, se verifica existenta valorilor lipsa si se genereaza statistici descriptive. corupte.

Se genereaza vizualizari grafice pentru a intelege mai bine distributia si relatiile dintre variabile: histograme, countplot-uri pentru variabile categorice, scatter plot-uri fata de variabila tinta si o matrice de corelatii (heatmap). Sunt identificati si afisati outlierii folosind metoda IQR pentru fiecare variabila numerica.

Dupa analiza, coloana Brand Name este codificata numeric folosind LabelEncoder, iar Country este transformata in variabile binare prin One-Hot Encoding. Coloanele numerice sunt standardizate cu StandardScaler pentru a avea media 0 si deviatie standard 1, o practica comuna in modelele de invatare automata.

Datele sunt apoi reimpartite si salvate in fisiere CSV separate pentru antrenare si testare. Modelul ales este RandomForestRegressor, care este antrenat pe setul de antrenament. Dupa antrenare, se fac predictii, iar performanta este evaluata folosind eroarea medie absoluta (MAE), care in acest caz a fost de aproximativ 5224.29. Pentru o analiza vizuala a erorilor, este salvat si un grafic al reziduurilor (diferenta dintre valori reale si prezise).

In final, modelul antrenat si transformarile aplicate (scalerul, codificatorul de brand si lista caracteristicilor) sunt salvate in fisiere .pkl, astfel incat sa poata fi reincarcate si utilizate in viitoare aplicatii fara a fi nevoie de reantrenare.

Proiectul este astfel pregatit pentru integrarea intr-o interfata grafica.

EXPLICARE COD (interfata_grafica.py)

În acest fișier Gradio, am aplicat aceleași preprocesări ca în main.py, folosind modelul deja antrenat și salvat, împreună cu encoderul pentru branduri, scalerul pentru standardizarea variabilelor numerice și ordinea corectă a coloanelor. Scopul a fost să păstrez consistența între modul în care modelul a fost antrenat și cum primește noile date pentru predicție.

Ce am adăugat nou aici este o funcție numită predict(), care primește valorile completate de utilizator în interfață și le transformă corespunzător, pentru a putea fi introduse în model. Înainte de asta, am inclus câteva verificări utile: dacă utilizatorul introduce toate valorile numerice ca fiind zero sau dacă valorile sunt vizibil prea mici, programul oprește execuția și oferă un mesaj clar de eroare.

După generarea predicției, funcția construiește și un grafic scatter. Acesta compară predicțiile făcute pe datele din test.csv cu valorile reale și marchează predicția nouă a utilizatorului cu un punct roșu, astfel încât să fie ușor de observat în context.

Pentru a face modelul ușor de folosit, am creat o interfață vizuală cu Gradio. Aceasta include dropdown-uri pentru alegerea brandului și a țării, precum și câmpuri numerice pentru introducerea datelor de vânzări și comenzi. Interfața este lansată în browser și oferă în timp real atât valoarea estimată de returnare, cât și graficul care o plasează în raport cu restul datelor. Interfața construită cu Gradio folosește modelul deja antrenat în main.py. Nu se reantrenează nimic – toate transformările realizate în timpul procesului de antrenare (inclusiv codificarea brandului, scalarea variabilelor numerice și ordinea caracteristicilor) sunt încărcate din fișierele salvate (model.pkl, scaler.pkl, brand_encoder.pkl, model_features.pkl).

Această interfață permite utilizatorului să introducă datele într-un mod intuitiv, iar în fundal, acele date sunt procesate exact ca în timpul antrenării. Predicția returnată este făcută de modelul antrenat anterior, iar utilizatorul primește atât valoarea estimată a returnărilor, cât și un grafic comparativ cu valorile reale din setul de test.

Pentru a realiza acest proiect m-am folosit de laboratoare în mare parte și m-am uitat și pe internet pentru a vedea cum să implementez anumite părți din cod.