**MINISTERUL EDUCAŢIEI ȘI CERCETĂRII AL REPUBLICII MOLDOVA**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică şi Microelectronică**

**Departamentul Ingineria Software și Automatică**

**Programul de studii: Tehnologia informației**

RAPORT

# LUCRARE DE LABORATOR NR. 1

# la Inteligența Artificială

**Tema: Regresie liniară***.*

A efectuat:

st. gr. TI-211 Popa Cătălin

A verificat: lect.dr. Mariana Rusu

UTM, Chișinău 2023

**Tema**

Unul din principalele capitole ale statisticii are în vedere posibilitatea de a face predicții. Prin intermediul regresiei se pot face predicții ale unei variabile, în funcție de valoarea alteia. Predicția este procesul de estimare a valorii unei variabile cunoscând valoarea unei alte variabile.

În continuare, ne vom referi doar la situația regresiei simple (o variabilă dependentă și una independentă) și liniare (relația dintre cele doua variabile poate fi descrisă printr-o dreaptă în cadrul norului de puncte).

Regresia se leagă foarte mult de conceptul de corelație. O asociere puternică între două elemente conduce la creșterea preciziei predicției unei variabile pe seama alteia.

**Pasul 1**

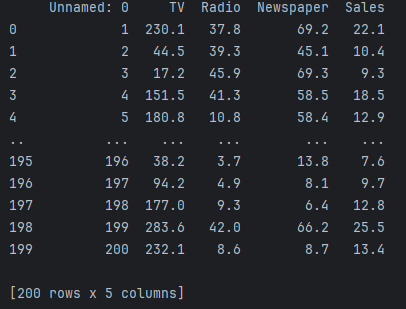
Importați panda, numpy, matplotlib și seaborn. Apoi setați %matplotlib inline (veți importa apoi sklearn după necesitate).

import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
from sklearn.model\_selection import train\_test\_split  
from sklearn.linear\_model import LinearRegression  
from sklearn.metrics import mean\_absolute\_error, mean\_squared\_error, r2\_score

**Pasul 2**

Citiți fișierul CSV ca DataFrame numit.

vanzari = pd.read\_csv("Advertising.csv")  
print(vanzari)

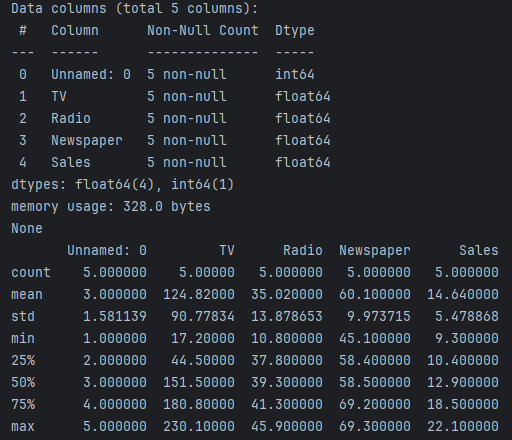


**Figura 3 – Afisare date csv.**

**Pasul 3**

Citiți primele linii (head) despre clienți și aflați informații despre ei folosinf funcțiile info() și describe().

def afisare\_info():  
 users = vanzari.head()  
 print(users.info())  
 print(users.describe())  
  
afisare\_info()

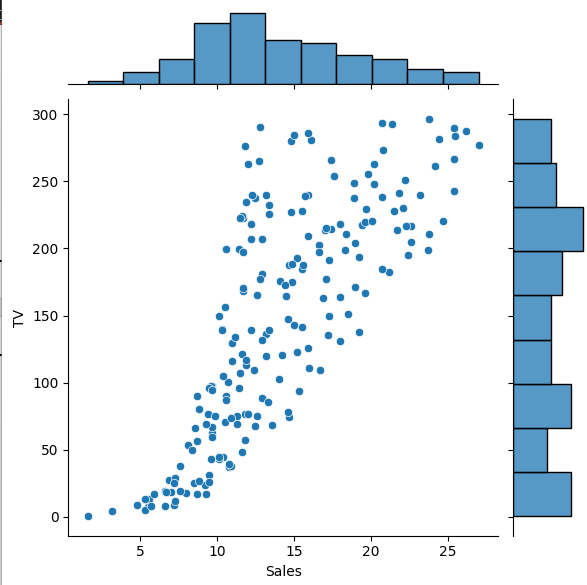


**Figura 5 – Afisare informatii primii clienti.**

**Pasul 3.1**

Utilizați seaborn pentru a crea o diagramă comună (**jointplot**) pentru a compara 2 coloane.

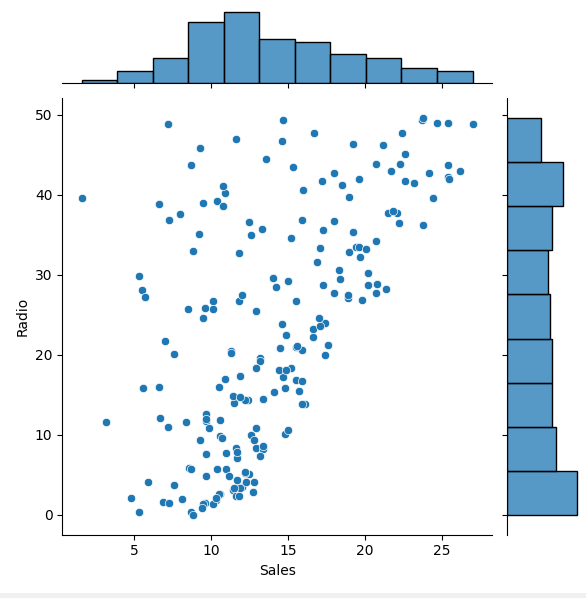
def analiza\_date():  
 sns.jointplot(x='Sales', y='TV', data=vanzari)  
 plt.show()  
  
analiza\_date()



**Pasul 4**

Faceți același lucru, dar cu alte coloane.

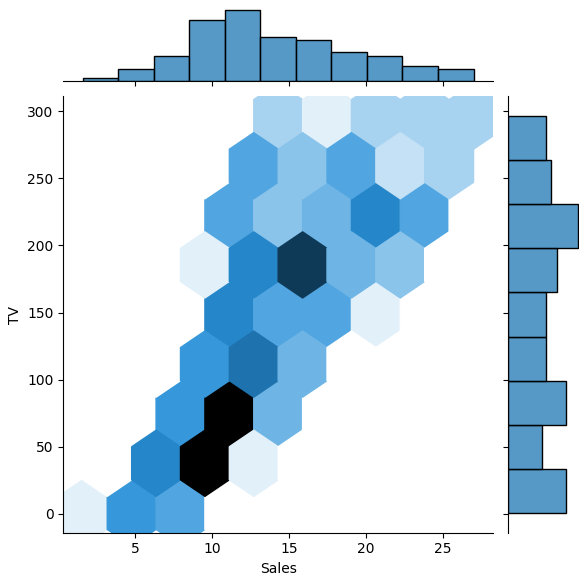
def analiza\_date2():  
 sns.jointplot(x='Sales', y='Radio', data=vanzari)  
 plt.show()  
  
analiza\_date2()



**Pasul 5**

Folosiți jointplot() pentru a crea o diagramă 2D cu coloanele luni conectat si numarul de reels.

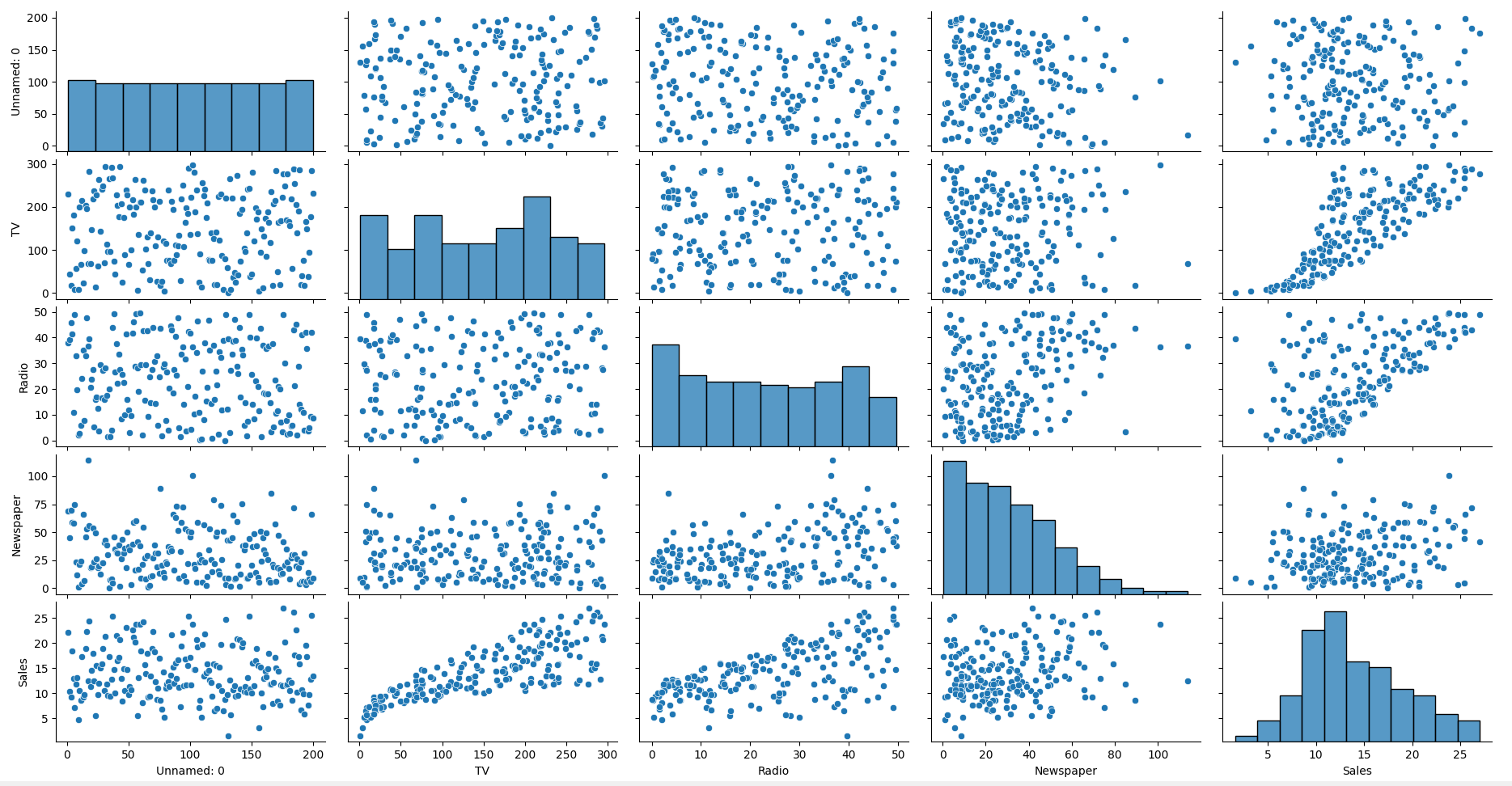
def d2d\_diagram():  
 sns.jointplot(data=vanzari, x="Sales", y="TV", kind="hex")  
 plt.show()



**Pasul 6**

Să explorăm aceste tipuri de relații în întregul set de date. Utilizați pairplot pentru a recrea graficul de mai jos.

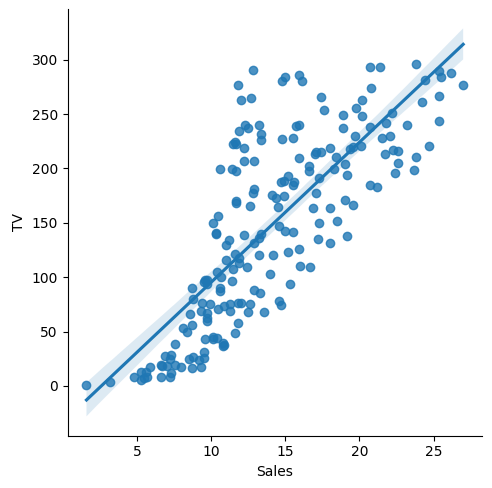
def airplot():  
 sns.pairplot(data=vanzari)  
 plt.show()  
  
airplot()



**Pasul 7**

Creați un model liniar grafic (folosind Implot din seaborn) al datelor: Urmaritori si timp inregistrare cont.

def model\_liniar():  
 sns.lmplot(data=vanzari, x="Sales", y="TV")  
 plt.show()



**Date de instruire și testare**

Să împărțim datele în seturi de antrenament și de testare

**Pasul 8**

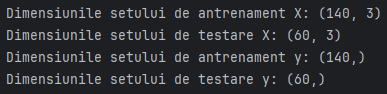
Setați o variabilă X egală cu caracteristicile numerice ale clienților și o variabilă y egală cu coloana Followers.

X = vanzari[['TV', 'Radio','Newspaper']]  
Y = vanzari['Sales']

**Pasul 9**

Utilizați model\_selection.train\_test\_split din sklearn pentru a împărți datele în seturi de antrenament și de testare. Setați test\_size=0,3 și random\_state=101

X\_antrenamente, X\_test, Y\_antrenamente, Y\_test = train\_test\_split(X, Y, test\_size=0.3, random\_state=101)  
  
print("Dimensiunile setului de antrenament X:", X\_antrenamente.shape)  
print("Dimensiunile setului de testare X:", X\_test.shape)  
print("Dimensiunile setului de antrenament y:", Y\_antrenamente.shape)  
print("Dimensiunile setului de testare y:", Y\_test.shape)



**Antrenarea modelului**

**Pasul 10**

Importați LinearRegression din sklearn.linear\_model.

from sklearn.linear\_model import LinearRegression

**Pasul 11**

Creați o instanță a unui model LinearRegression() numit lm.

lm = LinearRegression()

**Pasul 12**

Antrenați lm pe datele de antrenament. LinearRegression(copy\_X=True, fit\_intercept=True, n\_jobs=1, normalize=False)

LinearRegression(copy\_X=True, fit\_intercept=True, n\_jobs=1,positive=False)

lm.fit(X\_antrenamente, Y\_antrenamente)

**Pasul 13**

Imprimați coeficienții modelului.

print("Coeficienții modelului sunt:")  
print("Coeficienții:", lm.coef\_)



**Prezicerea datelor de testare**

**Pasul 14**

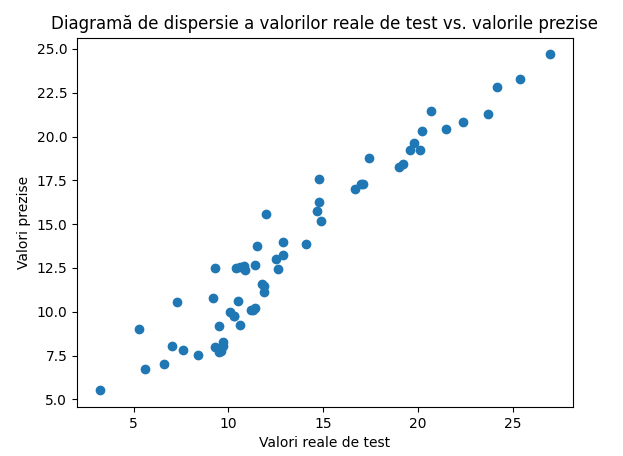
Utilizați lm.predict() pentru a prezice setul X\_test de date.

predictie = lm.predict(X\_test)

**Pasul 15**

Creați o diagramă de dispersie a valorilor reale de test față de valorile prezise.

plt.scatter(Y\_test, predictie)  
plt.xlabel("Valori reale de test")  
plt.ylabel("Valori prezise")  
plt.title("Diagramă de dispersie a valorilor reale de test vs. valorile prezise")  
plt.show()



**Evaluarea modelului**

Să evaluăm performanța modelului nostru calculând suma reziduală a pătratelor și scorul de varianță (R^2).

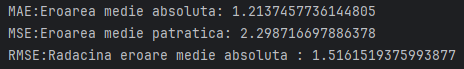
**Pasul 16**

Calculați eroarea medie absolută, eroarea medie pătratică și eroarea medie pătratică.

Consultați prelegerea sau Wikipedia pentru formule.

eroarea\_medie\_absoluta = mean\_absolute\_error(Y\_test, predictie)  
eroarea\_medie\_patratica = mean\_squared\_error(Y\_test, predictie)  
radacina\_eraore\_medie\_absoluta = np.sqrt(eroarea\_medie\_patratica)

print("MAE:Eroarea medie absoluta:", eroarea\_medie\_absoluta)  
print("MSE:Eroarea medie patratica:", eroarea\_medie\_patratica)  
print("RMSE:Radacina eroare medie absoluta :", radacina\_eraore\_medie\_absoluta)



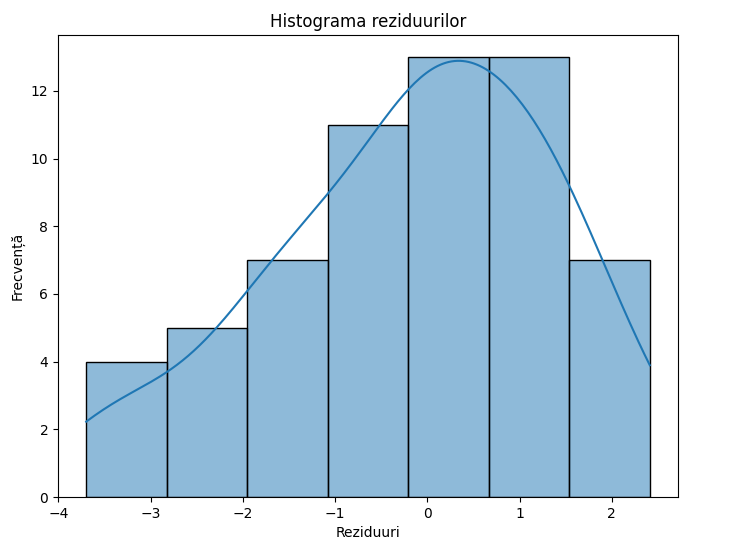
**Reziduri**

Să explorăm rapid reziduurile pentru a ne asigura că totul a fost în regulă cu datele noastre.

**Pasul 17**

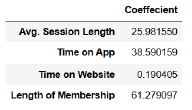
Trasați o histogramă a reziduurilor și asigurați-te că are o distribuție normală. Utilizați fie seaborn distplot, fie doar plt.hist().

reziduri = Y\_test - predictie  
  
plt.figure(figsize=(8, 6))  
sns.histplot(reziduri, kde=True)  
plt.title('Histograma reziduurilor')  
plt.xlabel('Reziduuri')  
plt.ylabel('Frecvență')  
plt.show()

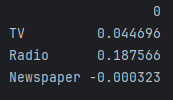


**Pasul 18**

Recreați setul de date de mai jos.



new\_datafreame = pd.DataFrame(lm.coef\_)  
new\_datafreame = new\_datafreame.rename(index=dict(zip(new\_datafreame.index, X)))  
print(new\_datafreame)



**Pasul 19**

Coeficienți de regresie: Pentru fiecare unitate de creștere a bugetului pentru publicitate TV, se estimează o creștere a vânzărilor de aproximativ 0.0447 unități. Pentru fiecare unitate de creștere a bugetului pentru publicitate la radio, se estimează o creștere a vânzărilor de aproximativ 0.1876 unități. Pentru fiecare unitate de creștere a bugetului pentru publicitate în ziare, se estimează o scădere foarte mică (-0.0003) a vânzărilor. Acest lucru poate sugera că publicitatea în ziare nu are un impact semnificativ asupra vânzărilor sau că impactul său este negativ, dar foarte mic.

**Pasul 20**

Bugetele pentru publicitate la televizor și radio par să aibă un impact semnificativ pozitiv asupra vânzărilor, în timp ce publicitatea în ziare nu pare să aibă un impact semnificativ sau poate chiar negativ.

**Concluzie:**

În concluzie, în funcție de aceste rezultate, o strategie de marketing care se concentrează pe publicitatea la televizor și radio ar putea fi mai eficientă în creșterea vânzărilor decât cea care se bazează pe publicitatea în ziare. Modelul de regresie liniară pare să se potrivească relativ bine datelor, cu un scor R^2 de aproximativ 0.9186, ceea ce înseamnă că explică aproximativ 91.86% din variația datelor de vânzare. Eroarea medie absolută (MAE) este de aproximativ 1.2137, iar eroarea medie pătratică (MSE) este de aproximativ 2.2987. Aceste valori indică o anumită eroare în predicțiile modelului, dar aceasta pare să fie acceptabilă, având în vedere dimensiunile seturilor de dat. Diagrama de dispersie a valorilor reale de test versus valorile prezise arată că există o corelație pozitivă între acestea, deși există unele discrepanțe, așa cum arată și reziduurile. Este important să se continue evaluarea și ajustarea modelului pentru a îmbunătăți performanța sa și pentru a lua decizii mai informate în ceea ce privește alocarea bugetelor de publicitate.