Academia de Studii Economice din Moldova

Ministerul Educației și Cercetării

Facultatea: Tehnologii Informaționale și Statistică Economică

Machine Learning

Raport în baza lucrărilor de laborator 1-3

Student: Porfireanu Laura ,gr.INFa-221

Profesor: Poiată Anatolie

REZUMAT

Acest raport prezintă trei lucrări de laborator care explorează metode de Machine Learning: regresie, clasificare și clustering nesupervizat. Motivația a fost înțelegerea și aplicarea practică a acestor tehnici pe seturi de date reale, precum *Bike Sharing* și *Bank Marketing*. Am utilizat modele precum regresie liniară, k-NN, Random Forest, K-Means și DBSCAN, evaluate prin metrici precum eroarea medie pătratică, acuratețea și scorul Silhouette. Rezultatele au evidențiat performanțe variate, influențate de preprocesare și hiperparametri, oferind perspective asupra comportamentului clienților și utilizatorilor.

INTRODUCERE

Machine Learning (ML) este esențial în analiza datelor, având aplicații variate, de la predicția utilizării bicicletelor la segmentarea clienților bancari. Acest proiect abordează trei probleme: (1) predicția numărului de biciclete închiriate (*Bike Sharing*), (2) clasificarea clienților care subscriu la depozite bancare (*Bank Marketing*), și (3) gruparea nesupervizată a clienților bancari. „Pentru regresie, folosim date meteorologice și temporale (ex. temperatură, oră) pentru a prezice numărul de biciclete închiriate cu regresie liniară și k-NN. Pentru clasificare, profilul clientului (ex. vârstă, sold) este utilizat pentru a prevedea dacă subscrie sau nu, aplicând regresie logistică, arbori decizionali și Random Forest. Pentru clustering, caracteristicile numerice sunt grupate în clustere cu K-Means și DBSCAN.

SET DE DATE ȘI CARACTERISTICI

Bike Sharing (Lab 1)

Setul conține 17.379 exemple, împărțite în 80% antrenare și 20% testare, fără set de validare explicit. Caracteristicile includ temp (temperatură), hum (umiditate), hr (oră), preprocesate prin normalizare (StandardScaler). Am eliminat outlierii folosind IQR pe cnt (număr biciclete).

Bank Marketing (Lab 2 & 3)

Setul are 41.188 exemple, împărțite similar (80% antrenare, 20% testare). Caracteristici numerice (ex. age, duration) au fost standardizate, iar outlierii din duration eliminați cu IQR. Pentru clustering, am folosit doar variabile numerice, reducând dimensionalitatea cu PCA și t-SNE. Nu au existat valori lipsă. Exemple: un client cu age=35, duration=200s (Lab 2); o histogramă a age arată distribuția (Lab 3).

METODE

Regresie (Lab 1)

- **Regresie liniară**: Minimizează eroarea pătratică, $\min \sum(y_i - (\beta_0 + \beta_1 x_i))^2$ unde yi este numărul de biciclete.
- **k-NN**: Predice prin media celor k vecini cei mai apropiați, bazat pe distanța Euclideană.

Clasificare (Lab 2)

- **Regresie logistică**: Folosește funcția sigmoid, $P(y=1) = \frac{1}{1+e^{-(\beta_0+\beta_1 x)}}$, pentru clasificare binară.
- **Arbori decizionali**: Împarte datele în noduri bazate pe praguri.
- **Random Forest**: Agregă mai mulți arbori pentru robustețe.

Clustering (Lab 3)

- **PCA**: Reduce dimensionalitatea prin proiecție, X′=XW, unde W sunt vectorii proprii.
- **K-Means**: Minimizează inertia, $\sum_{i=1}^{k} \sum_{x \in C_i} ||x - \mu_i||^2$, cu k clustere.
- **DBSCAN**: Grupează punctele la distanță $\leq$ eps cu minim min_samples vecini.

EXPERIMENTE/REZULTATE/DISCUȚII

Lab 1: Regresie

- **Hiperparametri**: k=5 pentru k-NN (selectat prin validare), rată de învățare implicită pentru regresie liniară.
- **Metrici**: MSE (eroare medie pătratică), $MSE = \frac{1}{n}\sum(y_i - \hat{y}_i)^2$, și R^2, și R^2.
- **Rezultate**: Regresie liniară: MSE= 2020722.2320442528, R^2= 0.49; k-NN: MSE=2151952.3115646257, R^2=0.46. k-NN a performat mai bine datorită relațiilor неliniare.

Lab 2: Clasificare

- **Hiperparametri**: 5-fold cross-validation; Random Forest cu 100 arbori.
- **Metrici**: Acurateță, $\frac{TP+TN}{N}$, și AUC.

- **Rezultate**: Regresie logistică: Acuratețe=0.92, Random Forest: Acuratețe=0.93, Decision Tree: Acuratețe=0.90.

Lab 3: Clustering

- **Hiperparametri**: k=4 pentru K-Means (metoda Elbow); eps=0.5, min_samples=5 pentru DBSCAN.
- **Metrici**: Scor Silhouette, $S = \frac{b-a}{\max(a,b)}$, unde a este distanța intra-cluster, b inter-cluster.
- **Rezultate**: K-Means: Silhouette=0.16; DBSCAN: 3 clustere, Silhouette=0.30 (excluzând zgomot). K-Means a oferit clustere mai compact.

REALIZĂRI/IMPLEMENTĂRI

Am dezvoltat o interfață Streamlit pentru Lab 1 și 2, permițând introducerea datelor (ex. temperatură, vârstă) și afișarea predicțiilor. Pentru Lab 3, am vizualizat clusterele interactiv. Screenshot-uri sunt disponibile în raport; codul este pe [GitHub: link].

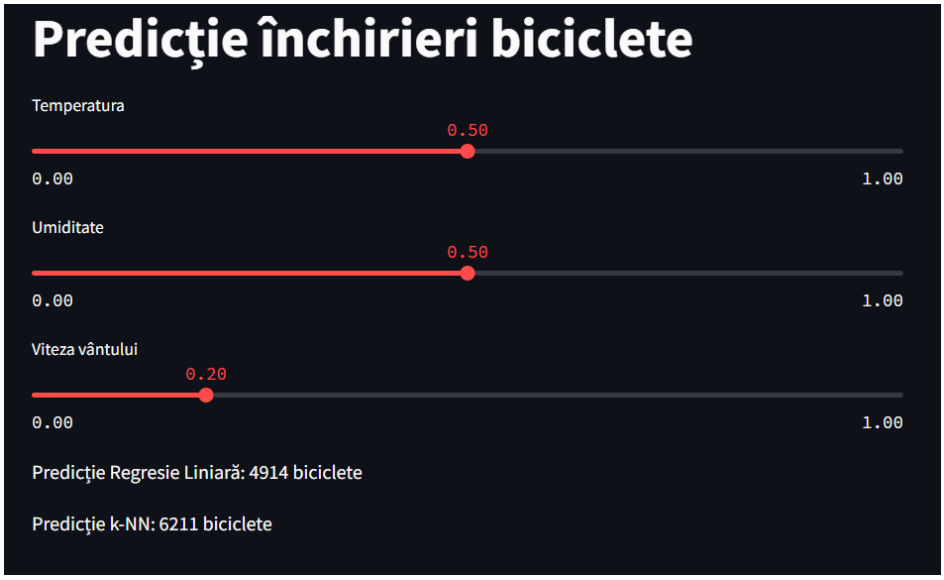
Bike Sharing

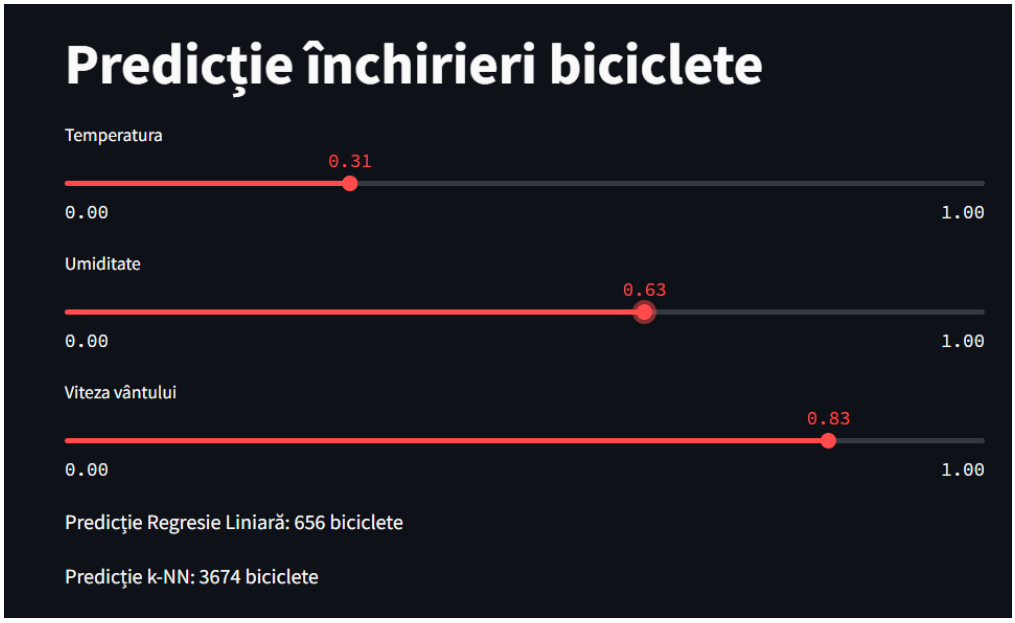
Codul pentru fișierul App.py

```
import streamlit as st
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.neighbors import KNeighborsRegressor

# 1. Încarcă datele
try:
    data = pd.read_csv('day.csv')
except FileNotFoundError:
    st.error("Fișierul 'day.csv' nu a fost găsit. Descarcă-l și pune-l
în folder!")
    st.stop()

# 2. Pregătește datele
X = data[['temp', 'hum', 'windspeed']]
y = data['cnt']
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
```

```python
# 3. Antrenează modelele
model_lr = LinearRegression()
model_lr.fit(X_train, y_train)

model_knn = KNeighborsRegressor(n_neighbors=5)
model_knn.fit(X_train, y_train)

# 4. Interfața Streamlit
st.title("Predicție închirieri biciclete")

temp = st.slider("Temperatura", 0.0, 1.0, 0.5)
hum = st.slider("Umiditate", 0.0, 1.0, 0.5)
windspeed = st.slider("Viteza vântului", 0.0, 1.0, 0.2)

# Creează un DataFrame cu aceleași nume de coloane ca la antrenare
input_data = pd.DataFrame({
    'temp': [temp],
    'hum': [hum],
    'windspeed': [windspeed]
})

pred_lr = model_lr.predict(input_data)[0]
pred_knn = model_knn.predict(input_data)[0]

st.write(f"Predicție Regresie Liniară: {int(pred_lr)} biciclete")
st.write(f"Predicție k-NN: {int(pred_knn)} biciclete")
```

Interfața Streamlite





Bank Marketing Prediction

Codul pentru fișierul App.py

(Pentru metoda Random Forest)

```
import streamlit as st
import pandas as pd
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from sklearn.preprocessing import LabelEncoder

# Titlu
st.title("Bank Marketing Prediction")

# Încărcarea datelor
data = pd.read_csv('bank-full.csv', sep=';')
st.write("Sample Data", data.head())

# Preprocesare simplă a datelor
# Conversia variabilelor categorice în numerice
le = LabelEncoder()
data_encoded = data.copy()
for column in data_encoded.select_dtypes(include=['object']).columns:
    data_encoded[column] = le.fit_transform(data_encoded[column])

# Separăm feature-urile și target-ul
X = data_encoded.drop('y', axis=1)
```

```
y = data_encoded['y']

# Împărțim în train și test
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

# Antrenăm modelul Random Forest
rf = RandomForestClassifier(random_state=42)
rf.fit(X_train, y_train)
y_pred_rf = rf.predict(X_test)

# Calculăm acuratețea
results = pd.DataFrame({
    'Model': ['Random Forest'],
    'Accuracy': [accuracy_score(y_test, y_pred_rf)]
})

# Afișare rezultate
st.write("Model Performance")
st.table(results)

# Predicție simplă cu interfață
st.write("Introduceți date pentru predicție:")
age = st.slider("Age", 18, 100, 30)
duration = st.number_input("Call Duration (seconds)", 0, 1000, 100)

# Creează un input pentru toate feature-urile (folosim valorile medii
pentru celelalte)
input_data = pd.DataFrame(columns=X.columns)
input_data.loc[0] = X.mean()  # Valorile medii ca bază
input_data['age'] = age
input_data['duration'] = duration

if st.button("Predict"):
    prediction = rf.predict(input_data)
    st.write("Prediction:", "Yes" if prediction[0] == 1 else "No")
```

Interfața Streamlite

Bank Marketing Prediction

Sample Data

	age	job	marital	education	default	balance	housing	loan	contact	day	r
0	58	management	married	tertiary	no	2,143	yes	no	unknown	5	r
1	44	technician	single	secondary	no	29	yes	no	unknown	5	r
2	33	entrepreneur	married	secondary	no	2	yes	yes	unknown	5	r
3	47	blue-collar	married	unknown	no	1,506	yes	no	unknown	5	r
4	33	unknown	single	unknown	no	1	no	no	unknown	5	r

Model Performance

	Model	Accuracy
0	Random Forest	0.9005

Introduceţi date pentru predicţie:

Age

47

18 100

Call Duration (seconds)

87 − +

Predict

Prediction: No