

Raport: Predicția Soldului Total în Sistemul Energetic Național (SEN)

Bocăneț Raluca-Andreea

02.01.2025

1. Introducere

Scopul acestui proiect este de a prezice soldul total din Sistemul Energetic Național (SEN) pentru luna decembrie 2024 folosind două tehnici de învățare automată: **Decision Tree Regressor (ID3)** și **Bayes Naiv**. Datele utilizate pentru antrenarea și testarea modelelor provin dintr-un fișier Excel care conține informații despre consumul și producția de energie pe diferite surse de energie (carbune, hidrocarburi, eolian, etc.) și soldul de energie.

2. Procesarea datelor

2.1 Incarcarea și explorarea datelor

Datele au fost încărcate din fișierul Excel `Grafic_SEN.xlsx`. La început, au fost verificate informațiile generale despre date, precum tipurile de date și statistici descriptive, folosind următoarele comenzi:

```
print(data.info())  
print(data.describe())
```

2.2 Curatarea datelor

Coloanele numerice care conțin datele de consum și producție au fost curățate pentru a elimina caracterele non-numerice și a transforma valorile într-un tip numeric adecvat (`float`). Procesul a fost realizat cu ajutorul următorului cod:

```
for col in columns_to_clean:  
    data[col] = data[col].replace(r'[^\\d.-]', '', regex=True).astype(float)
```

2.3 Conversia datelor

Coloana `Data` a fost transformata într-un format de tip `data (datetime)`, iar valorile lipsa au fost eliminate pentru a asigura o baza de date curata si completa. Codul folosit pentru aceasta conversie este:

```
data['Data'] = pd.to_datetime(  
    data['Data'],  
    format='%d-%m-%Y %H:%M:%S',  
    errors='coerce'  
)  
data = data.dropna()
```

2.4 Divizarea datelor in seturi de antrenament si testare

Datele au fost impartite pe baza lunii. Setul de antrenament include datele din lunile diferite de decembrie, iar setul de testare include datele din luna decembrie 2024. Daca setul de date de antrenament este gol, s-a folosit o divizare aleatorie a datelor. Codul utilizat este:

```
train_data = data[data['Data'].dt.month != 12]  
test_data = data[data['Data'].dt.month == 12]
```

2.5 Selectarea caracteristicilor si a tinteii

Caracteristicile utilizate pentru a prezice soldul sunt variabilele legate de consumul si producția de energie. Ținta este soldul de energie (`Sold[MW]`). Codul pentru definirea caracteristicilor si a tinteii:

```
features = [  
    'Consum[MW]', 'Medie Consum[MW]',  
    'Productie[MW]', 'Carbune[MW]', 'Hidrocarburi[MW]', 'Ape[MW]',  
    'Nuclear[MW]', 'Eolian[MW]', 'Foto[MW]', 'Biomasa[MW]',  
]  
target = 'Sold[MW]'
```

3. Modele de invatare automata

3.1 Modelul Decision Tree Regressor (ID3)

Modelul ID3 a fost antrenat cu datele de antrenament si a fost folosit pentru a prezice soldul de energie. Performanta a fost evaluata folosind doua metrice: **Root Mean Squared Error (RMSE)** și **Mean Absolute Error (MAE)**.

```
id3_model = DecisionTreeRegressor(max_depth=5, random_state=42)
id3_model.fit(X_train, y_train)
y_pred_id3 = id3_model.predict(X_test)
```

Rezultatele obtinute pentru modelul ID3 au fost:

RMSE: 109.52

MAE: 85.41

3.2 Modelul Bayes Naiv

Pentru modelul Bayes Naiv, datele de antrenament si testare au fost discretizate in intervale de valori (binning) pentru a se potrivi cu natura algoritmului.

```
bins = np.linspace(X_train.min().min(), X_train.max().max(), 10)
X_train_binned = np.digitize(X_train, bins=bins)
X_test_binned = np.digitize(X_test, bins=bins)
bayes_model = GaussianNB()
bayes_model.fit(X_train_binned, y_train)
y_pred_bayes = bayes_model.predict(X_test_binned)
```

Performanța modelului Bayes Naiv a fost evaluata cu urmatoarele rezultate:

RMSE: 295.08

MAE: 228.37

4. Compararea performantei

O comparatie vizuala intre predictiile celor doua modele a fost realizata printr-un grafic de dispersie, unde valorile reale ale soldului au fost comparate cu valorile prezise de ambele modele. Graficul a fost generat folosind urmatorul cod:

```
plt.figure(
    figsize=(10, 6))sns.scatterplot(x=y_test,
    y=y_pred_id3, label='ID3', color='blue', s=100, marker='o',
    alpha=0.6)sns.scatterplot(x=y_test, y=y_pred_bayes,
    label='Bayesian', color='red', s=100,
```

```

        marker='^', alpha=0.6)
for i in range(len(y_test)):
    plt.text(y_test.iloc[i], y_pred_id3[i],
            f'{data["Data"].iloc[i].strftime("%d-%m-%Y")}',
            fontsize=8, alpha=0.6)
plt.xlabel('Valoare reala')
plt.ylabel('Valoare prezisa')
plt.title('Compararea predictiilor cu Sold si Data')
plt.legend()
plt.show()

```

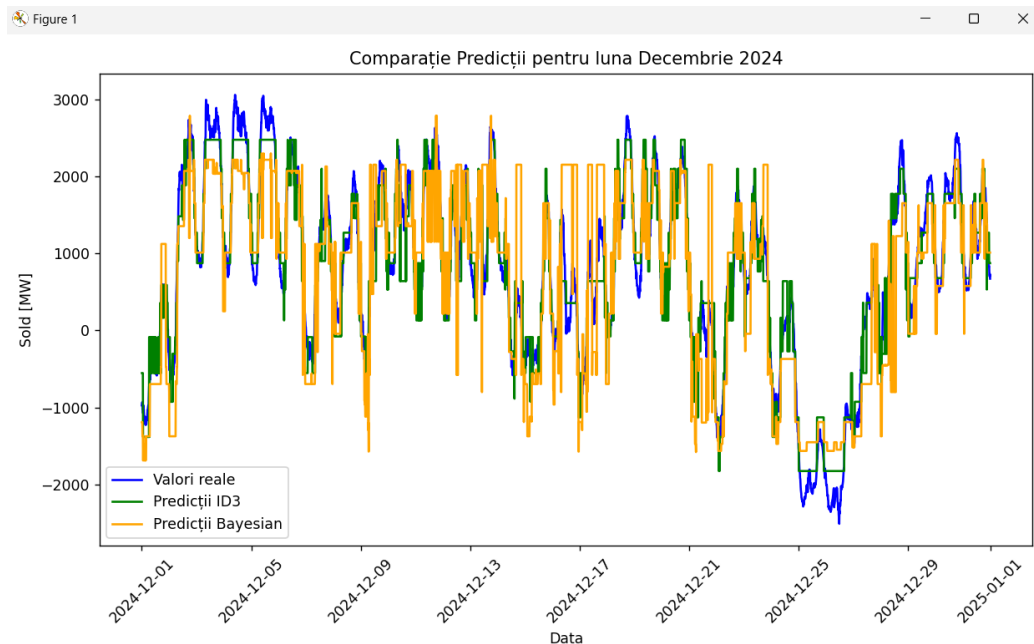


Figure 1: Compararea predictiilor intre modelele ID3 si Bayes Naiv

5. Cel mai mare sold din decembrie 2024

Cel mai mare sold pentru luna decembrie 2024 a fost gasit prin compararea valorilor din setul de date de testare. Valoarea maximă a soldului a fost de:

Max Sold: 3058.0 MW

Data corespunzătoare acestui sold maxim este:

Data: 04-12-2024 09:48:09

Pentru a obtine aceste informatii, s-a folosit urmatorul cod:

```
max_sold_december = test_data['Sold[MW]'].max()
max_sold_date =
    test_data[test_data['Sold[MW]'] ==
max_sold_december]['Data'].iloc[0]

print(f"Cel mai mare sold pentru luna decembrie 2024 este
    {max_sold_december} MW, la data
    {max_sold_date.strftime('%d-%m-%Y %H:%M:%S')}")
```

6. Concluzii

Modelul **Decision Tree Regressor (ID3)** a oferit o performanta mai buna decat modelul Bayes Naiv, avand valori RMSE și MAE mai mici. In ciuda acestui fapt, ambele modele au avut performanțe rezonabile, avand in vedere complexitatea datelor si natura lor.

Este recomandat sa se testeze si alte tehnici de invatare automata, cum ar fi **Random Forest** sau **XGBoost**, pentru a observa daca o performanta mai buna poate fi obtinuta.