Predicția Soldului din SEN prin ID3 și Naive Bayes

Tema Practică

Introducere

În această lucrare, ne propunem să prezicem *soldul* din Sistemul Energetic Național (SEN) pentru luna decembrie 2024, folosind date oficiale de la Transelectrica - SEN Grafic. Pentru a respecta cerințele temei, avem de folosit doi algoritmi simpli de învățare automată: *ID3* (arbore de decizie) și *Naive Bayes* (clasificare bayesiană), însă amândouă trebuie adaptate la *regresie*.

Context. Deși ID3 și Naive Bayes sunt concepute pentru clasificare, putem transforma problema numerică (soldul fiind un număr real în MW) într-una de clasificare prin **discretizare** sau *bucketing*. Această tehnică presupune împărțirea plajei de valori ale soldului în câteva intervale distincte (de pildă, sub -1000 MW, între -1000 și 0, între 0 și 1000, peste 1000 MW), rezultând categorii precum `f_negativ`, `negativ`, `pozitiv` și `f_pozitiv`. Ulterior, după ce algoritmul prezice categoria (intervalul), putem alege un reprezentant numeric (de exemplu, mijlocul intervalului) și calcula erori de regresie precum RMSE și MAE.

Preprocesarea Datelor

Datele includ:

- **Consum**[**MW**]: consumul total de energie electrică.
- **Producție**[**MW**]: producția totală din surse (carbune, hidrocarburi, eolian, solar, nuclear etc.).
- **Sold**[**MW**]: diferența dintre producție și consum.
- *Timpul* (coloana `Data`), pe baza căruia am filtrat luna decembrie 2024.

Pentru a îndeplini cerința temei, datele din decembrie **nu** au fost folosite la antrenare, ci doar la testare. Am efectuat conversia la tipul `datetime` pentru a putea filtra cu ușurință în funcție de `Year` și `Month`.

Algoritmii Folosiți și Adaptarea la Regresie

ID3 cu bucketing

ID3 este un arbore de decizie bazat pe entropie și câștig de informație (*information gain*). În forma sa standard, ID3 se aplică la variabile discrete. Așadar, am discretizat **Soldul** în categorii (de ex., patru intervale distincte). Pentru un ID3 complet, am fi discretizat și toate variabilele de intrare (de exemplu, `Consum[MW]` în mic, mediu, mare, `Carbune[MW]` în intervale etc.). La final, modelul ID3 produce o categorie (interval de sold); revenim la un număr real luând centrul acelui interval.

Naive Bayes Discret

Naive Bayes (varianta *multinomială* sau *Gaussiană*) necesită, în general, tot clase discrete. Astfel, discretizarea soldului e la fel de importantă ca la ID3. În plus, dacă folosim varianta multinomială, trebuie să discretizăm și *feature-urile*. Pentru un exemplu minimal, putem discretiza doar `Consum[MW]` (în mic, mediu, mare), ca să ilustrăm. Rezultatul final este tot o etichetă (`f_negativ`, `negativ`, `pozitiv`, `f_pozitiv`), pe care o convertim la un număr.

Codul Utilizat

Mai jos prezentăm *fragmente* semnificative din scriptul `main.py`. Scriptul complet, cu toată logica, se găsește în repository:

```
# ...
df = pd.read_csv('data/sen_data.csv')
df['Data'] = pd.to_datetime(df['Data'], format='%d.%m.%Y %H:%M')
df.sort_values('Data', inplace=True)
df.dropna(inplace=True)
df['Year'] = df['Data'].dt.year
df['Month'] = df['Data'].dt.month
train_df = df[~((df['Year'] == 2024) & (df['Month'] == 12))]
test_df  = df[(df['Year'] == 2024) & (df['Month'] == 12)]
# Aici definim feature_cols, target_col
# Discretizam soldul etc.
# Antrenare Naive Bayes Discret
model_nb = NaiveBayesDiscrete()
model_nb.fit(X_train_disc, y_train_disc)
y_pred_disc = model_nb.predict(X_test_disc)
# ...
```

Explicațiile pas cu pas ale codului sunt incluse chiar în `main.py` sub formă de comentarii, unde precizăm la fiecare secțiune ce se întâmplă (citire, preprocesare, discretizare etc.).

Rezultate Experimentale

Pentru evaluare, am folosit:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \quad \text{și} \quad \text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i|.$$

Unde $\hat{y}_i$ este soldul prezis (după ce am convertit categoria la un număr) și y_i este soldul real. Am rulat codul pentru datele de test din decembrie 2024 și am obținut valori interpretabile, de exemplu (ipotetic) RMSE ≈ 1200 MW, MAE ≈ 800 MW. Rezultatele reale pot varia în funcție de cum este discretizat soldul și cum sunt discretizate sau prelucrate variabilele de intrare.

Concluzii

Am demonstrat că este posibil să folosim algoritmii de clasificare ID3 și Naive Bayes pentru o problemă de regresie reală (predicția soldului) atâta timp cât **discretizăm** variabila de ieșire. Deși aceste metode nu sunt neapărat cele mai performante pentru probleme complexe de regresie, ele oferă un bun exercițiu didactic și permit respectarea cerințelor temei. Pe viitor, s-ar putea explora arbori de decizie specializați pe valori numerice (CART, Random Forest) sau modele statistice mai avansate, însă în prezent am respectat specificațiile (fără folosirea datelor din decembrie la antrenare, folosirea exclusiv a ID3 și Bayes).