Uczenie maszynowe w Python

Członkowie grupy:

1. Yahor Koval, ykoval@edu.cdv.pl
2. Tymur Popovych, tpopovych@edu.cdv.pl
3. Anton Talmachou, atalmachou@edu.cdv.pl

Uczenie maszynowe jako narzędzie wczesnego wykrywania chorób wątroby

Wybranym problemem badawczym jest automatyczna klasyfikacja pacjentów pod kątem chorób wątroby na podstawie parametrów biochemicznych krwi. Celem projektu było stworzenie modelu uczenia maszynowego, który na podstawie danych takich jak *wiek(*`age`*)*, *płeć*(`gender`) czy stężenie enzymów wątrobowych (`SGOT, SGPT`) i *bilirubiny* (`tot_bilirubin,direct_bilirubin`), potrafi odróżnić osoby *zdrowe od chorych*(`is_patient`).

1. Opis i uzasadnienie przeprowadzonych operacji

Proces został zaprojektowany w paradygmacie programowania obiektowego (OOP), co pozwala na zachowanie czystości kodu i łatwą reużywalność poszczególnych modułów.

- **Wstępne przetwarzanie danych:** Dane surowe z pliku CSV zawierały zmienne tekstowe oraz brakujące wartości. Uzasadnieniem mapowania płeć (`gender`) i zmiennej docelowej (`is_patient`) na wartości numeryczne jest wymóg algorytmów uczenia maszynowego, które operują na macierzach liczb.
- **Analiza eksploracyjna:** Wykorzystanie macierzy korelacji pozwala zidentyfikować, które wskaźniki biochemiczne (np. `bilirubina`, `albuminy`) są ze sobą silnie powiązane, co pomaga w zrozumieniu struktury danych.
- **Logarytmizacja danych:** W metodzie `boxplot` zastosowano transformację `log1p` dla cech o dużej asymetrii rozkładu, takich jak `tot_bilirubin` czy `alkphos`. Uzasadnia się to chęcią lepszej wizualizacji danych i ograniczenia wpływu wartości odstających na czytelność wykresów pudełkowych.
- **Podział danych i balansowanie klas:** Zastosowano podział 80/20 z zachowaniem proporcji klas (`stratify=y`), co jest kluczowe przy nierównomiernym rozkładzie pacjentów chorych i zdrowych w zbiorze. W samym modelu użyto parametru `class_weight='balanced'`, aby algorytm przywiązywał większą wagę do rzadziej występującej klasy.
- **Wybór modelu:** Wykorzystano **Random Forest Classifier** z *300* drzewami decyzyjnymi. Wybór ten jest uzasadniony odpornością tego algorytmu na przeuczenie oraz jego zdolnością do radzenia sobie z nieliniowymi zależnościami w danych medycznych.

2. Opis funkcji i działań wewnątrzklasowych

1. **`__init__(self, file_path)`**: Inicjalizuje obiekt, przechowując ścieżkę do pliku i rezerwując miejsce na ramkę danych oraz model.
2. **`load_and_preprocess(self)`**:
   - Wczytuje plik CSV.
   - Mapuje `'Male'` na 1 i `'Female'` na 0.
   - Mapuje zmienną `is_patient`: 1 (chory) na 1, 2 (zdrowy) na 0.
   - Czyści zbiór z brakujących rekordów.
3. **`exploratory_analysis(self)`**: Oblicza korelację między wszystkimi cechami i rysuje mapę ciepła (*heatmap*), co pozwala na wizualną ocenę zależności.
4. **`boxplot(self)`**: Tworzy wykresy pudełkowe dla kluczowych parametrów, stosując skalę logarytmiczną dla zmiennych o szerokim zakresie wartości, co pozwala na identyfikację rozkładów statystycznych.
5. **`prepare_datasets(self, test_size=0.2)`**: Przygotowuje dane do uczenia, konwertując je na format NumPy i dzieląc na zbiory `X` (cechy) oraz `y` (etykiety).
6. **`train_random_forest(self, n_estimators=300)`**:
   - Konfiguruje klasyfikator Random Forest Classifier.
   - Trenuje model na zbiorze treningowym.
   - Zapisuje wytrenowany model do pliku binarnego `liver_model.pickle`.
7. **`evaluate_model(self)`**:
   - Generuje predykcje dla danych testowych.
   - Oblicza wskaźniki trafności i generuje macierz pomyłek (Confusion Matrix).

3. Analiza wyników (Poprawność modelu)

Na podstawie wygenerowanych raportów, model uzyskał następujące wyniki na zbiorze testowym:

- **Accuracy (Dokładność): 70.69%** – Oznacza to, że ponad 70% wszystkich diagnoz (zarówno zdrowych, jak i chorych) było poprawnych.
- **Recall (Czułość): 92.77%** – Jest to najważniejszy parametr w medycynie. Wskazuje on, że model poprawnie zidentyfikował prawie 93% wszystkich osób rzeczywiście chorych na wątrobę. Wysoki recall minimalizuje ryzyko przeoczenia choroby (mało wyników fałszywie ujemnych).
- **F1-Score: 81.91%** – Średnia harmoniczna między precyzją a czułością, wskazująca na ogólną solidną jakość klasyfikacji modelu.

Wnioski: Model wykazuje się bardzo wysoką zdolnością do wykrywania chorych pacjentów (`Recall`), co przy ogólnej dokładności na poziomie 71% czyni go użytecznym narzędziem do wstępnych badań przesiewowych. Macierz pomyłek potwierdza te statystyki, pokazując wysoką skuteczność w klasie `'Sick'`.