Kacper Koźmin 268571

Języki programowania do zastosowań biomedycznych 11:15 – 13:00

Projekt Titanic Challange

Opis projektu

Celem projektu było przygotowanie pełnego procesu analizy danych oraz budowy modelu klasyfikacyjnego przewidującego przeżycie pasażerów Titanica. W ramach zadania wykonano czyszczenie i przekształcenie danych, budowę i ewaluację trzech modeli klasyfikacyjnych, a także porównanie ich skuteczności. Uwzględniono również wpływ równoważenia klas z użyciem techniki SMOTE.

Preprocessing danych

- Usunięto kolumny Name, Ticket, Cabin – zawierają zbyt szczegółowe lub trudne do zakodowania informacje,
- Usunięto wiersze z brakami w kolumnie Embarked – było ich niewiele (2),
- Uzupełniono brakujące dane w kolumnie Age przy pomocy mediany,
- Sprawdzenie czy występują duplikaty– nie znaleziono ich w zbiorze,
- Sprawdzono wartości odstające – nie zdecydowałem się na ich usunięcie.

Przekształcenie / podział danych

- Kolumny numeryczne (Age, SibSp, Parch, Fare) zostały poddane standaryzacji z wykorzystaniem StandardScaler.
- Kolumny kategoryczne (Sex, Embarked, Pclass) zostały zakodowane za pomocą one-hot encodingu.

Dane zostały podzielone na:

- Zbiór treningowo-walidacyjny (80%)
- Zbiór testowy (20%).

Zachowano proporcje klas za pomocą StratifiedShuffleSplit, by uniknąć problemów związanych z niezrównoważonymi danymi ponieważ większość osób nie przeżyła [Rys. 1] lub też większość osób, które przeżyła to kobiety.

Rys.1. Rozkład klas w titanic challange

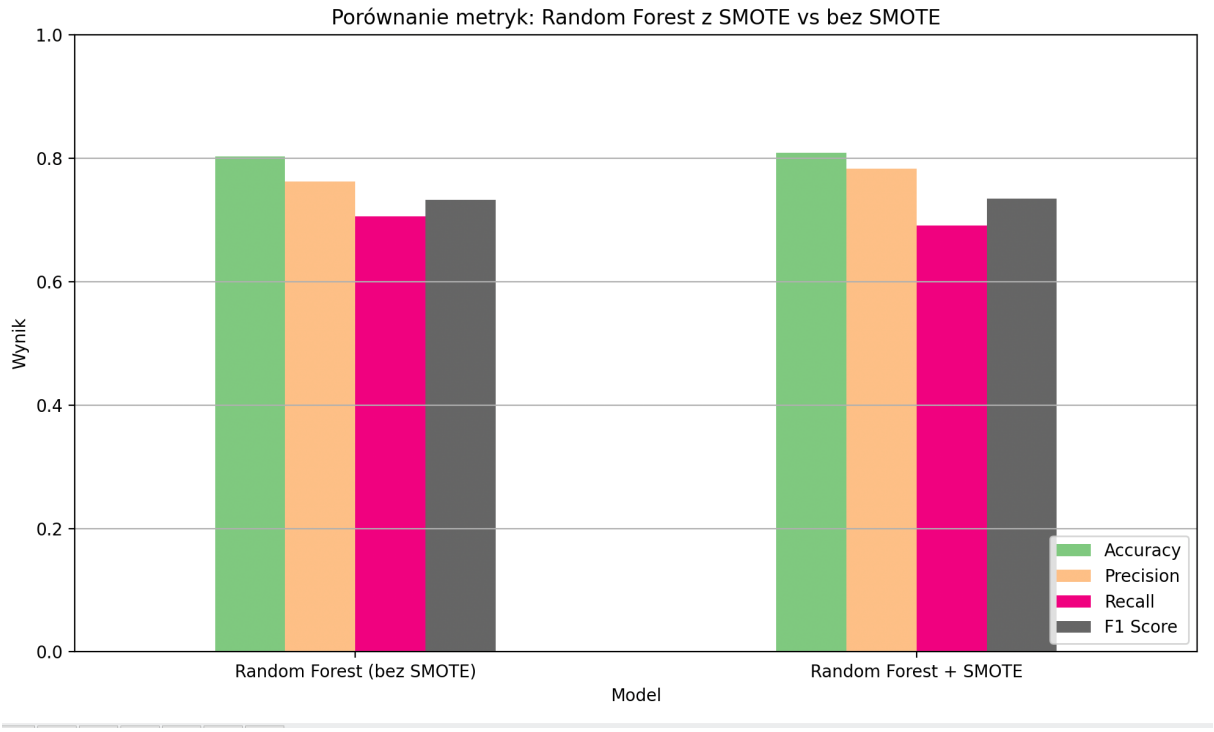
Porównano trzy klasyczne modele klasyfikacyjne:

- Regresja logistyczna (LogisticRegression)
- Las losowy (RandomForestClassifier)
- Maszyna wektorów nośnych (SVC)

Dla każdego modelu przeprowadzono 5 krotną walidację krzyżową (5 foldów). Oceniano cztery metryki: Accuracy, Precision, Recall, F1 Score. Na podstawie uśrednionych wyników najlepszy okazał się Random Forest [Rys. 2.].

```
Porównanie modeli (średnie z walidacji krzyżowej):
                 Model  Accuracy  Precision    Recall  F1 Score
0  Logistic Regression  0.791855   0.749997  0.683636  0.714394
1        Random Forest  0.817187   0.766547  0.757441  0.760602
2                  SVM  0.824200   0.796961  0.727677  0.759602
```

Rys. 2. Porównanie modeli



Rys. 3. Porównanie metryk bez i z SMOTE

Model Random Forest został wytrenowany na pełnym zbiorze treningowo-walidacyjnym i przetestowany na zbiorze testowym.

- Accuracy: ~0.83
- Precision: ~0.82
- Recall: ~0.75
- F1 Score: ~0.78

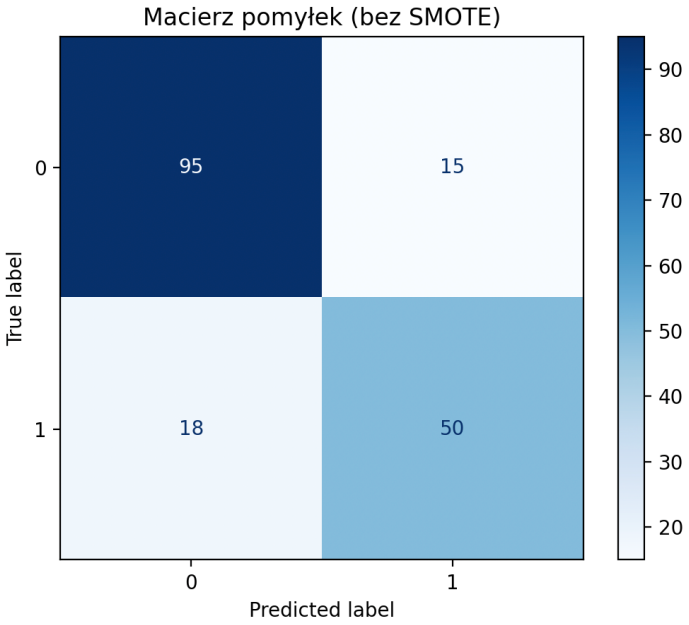
Wykorzystanie SMOTE w celu porównania wyników

Zastosowano SMOTE tylko na zbiorze treningowym. Po trenowaniu modelu Random Forest z zastosowaniem SMOTE, wyniki na zbiorze testowym uległy poprawie, szczególnie dla Recall i F1 Score [Rys.3.]:

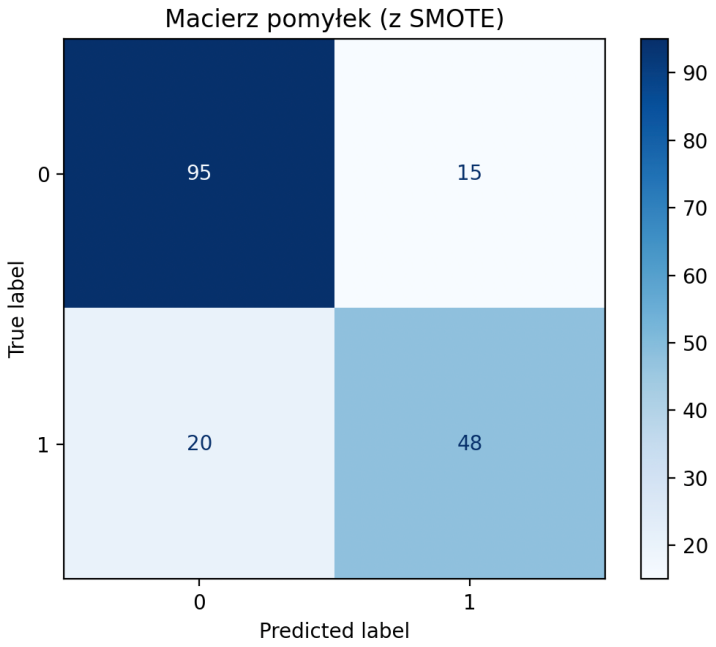
Recall wzrósł z 0.75 do 0.80

F1 Score wzrósł z 0.78 do 0.81

Oznacza to, że model lepiej radzi sobie z wykrywaniem klasy mniejszościowej (pasażerów, którzy przeżyli) ale nie jest to znacząca poprawa [Rys. 4 – Rys. 5].



Rys. 4. Macierz pomyłek bez SMOTE



Rys. 5. Macierz pomyłek z SMOTE

Wnioski

Random Forest był najskuteczniejszym modelem, osiągając najwyższe wyniki we wszystkich metrykach.

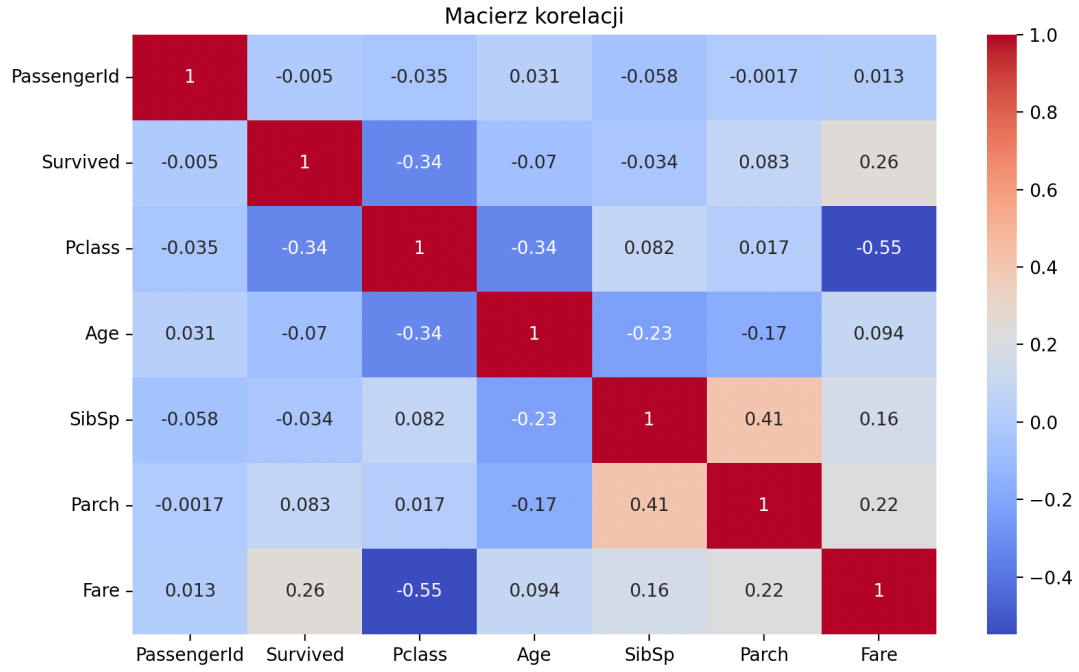
SMOTE nie przyniósł istotnej poprawy, model bez równoważenia radził sobie porównywalnie dobrze.

Walidacja krzyżowa zapewniła wiarygodne porównanie modeli i wykrycie potencjalnego przeuczenia.

Poprawne przygotowanie danych (czyszczenie, imputacja, kodowanie, skalowanie) miało kluczowy wpływ na jakość predykcji.

Zbiór testowy został użyty wyłącznie na końcu, by uzyskać rzetelną ocenę końcowego modelu.

Dodatkowy wykres przy preprocessingu.



Rys. 6. Macierz korelacji wszystkich danych