Warszawa 22.01.2020 r.

Wydział Elektroniki i Technik Informacyjnych

Podstawy Sztucznej Inteligencji

Projekt nr 2 Uczenie maszynowe PZ.U17

„Marketing bankowy”: Porównanie algorytmów: XGBoost (biblioteka wzmacniania gradientowego) i regresji logistycznej

Wykonawcy:

1. Sebastian Smoliński numer indeksu – opracowanie wyników, analiza danych i macierze 2.
2. Oleksandr Drobinin numer indeksu – algorytmy

Spis treści:

1. Opis projektu i cele

Celem projektu było wykonanie zadań klasyfikacji przy pomocy danych algorytmów (biblioteka XGBoost dla wzmacniania gradientowego oraz regresja logistyczna). W tym celu wykonano odpowiednią implementację w języku Python z wykorzystaniem środowiska Jupyter Notebook. Kod powinien korzystac z autmatycznego strojenia parametrów oraz odtworzenie wyników z raportu.

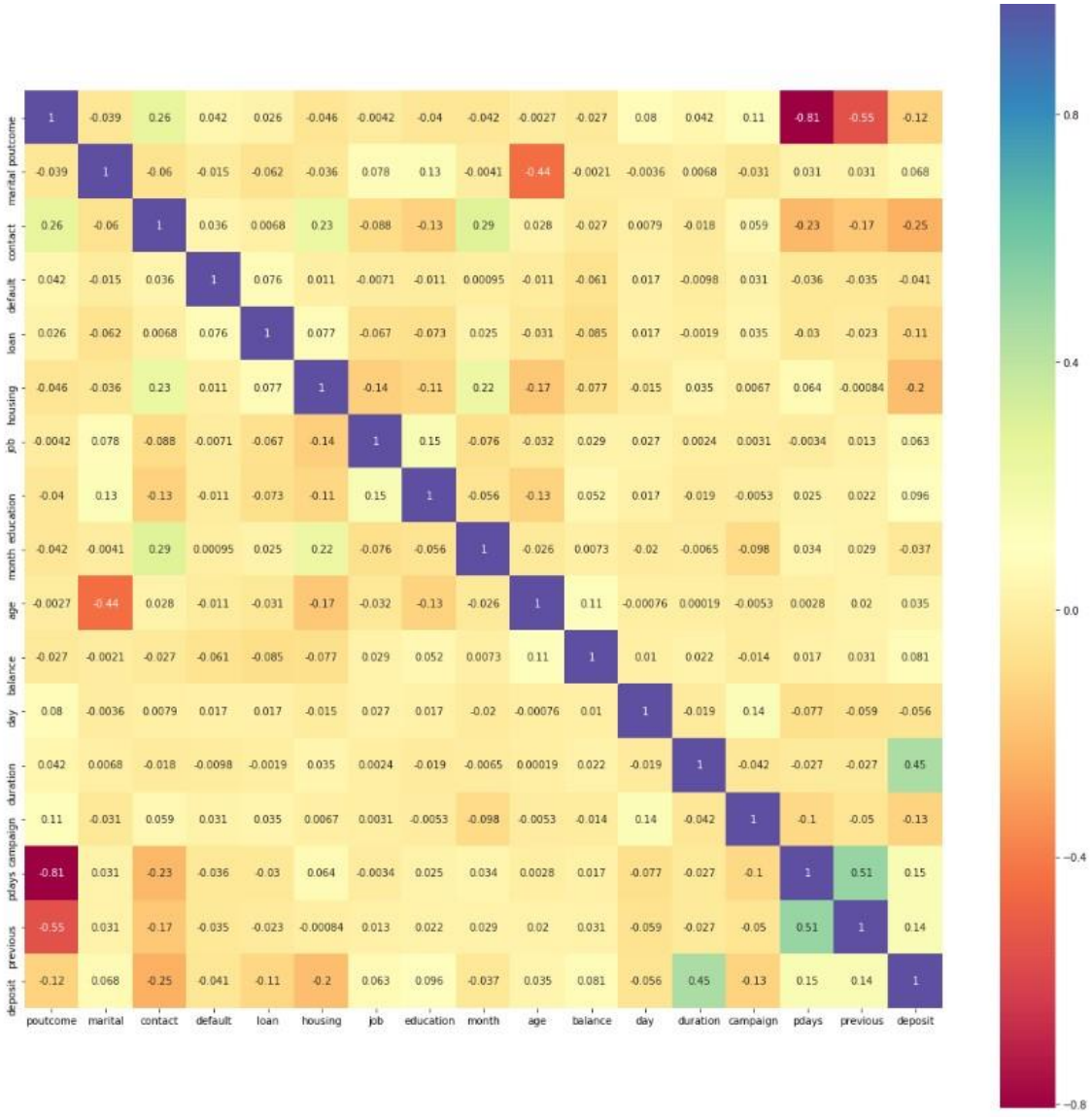
Założenia:

1. Dane wejściowe:

- były wprowadzane w postaci pliku w formacie .csv
- plik zawiera kilkanaście tysięcy rekordów opisanych przez 17 zmiennych parametrów

- najważniejszymi parametrami klasyfikacji w tym przypadku są dane opisowe, gdyż dane numeryczne nie mają tutaj aż tak dużego wpływu
- nie ma brakujących wartości w zbiorze, więc nie wypełniano żadnych pól ich wartościami średnimi itd.

Poniżej przedstawiono macierz korelacji dla poszczególnych parametrów w badanym zestawie:



Więcej wykresów i analiz można otrzymać z uruchomienia kodu.

2. Założenia projektowe

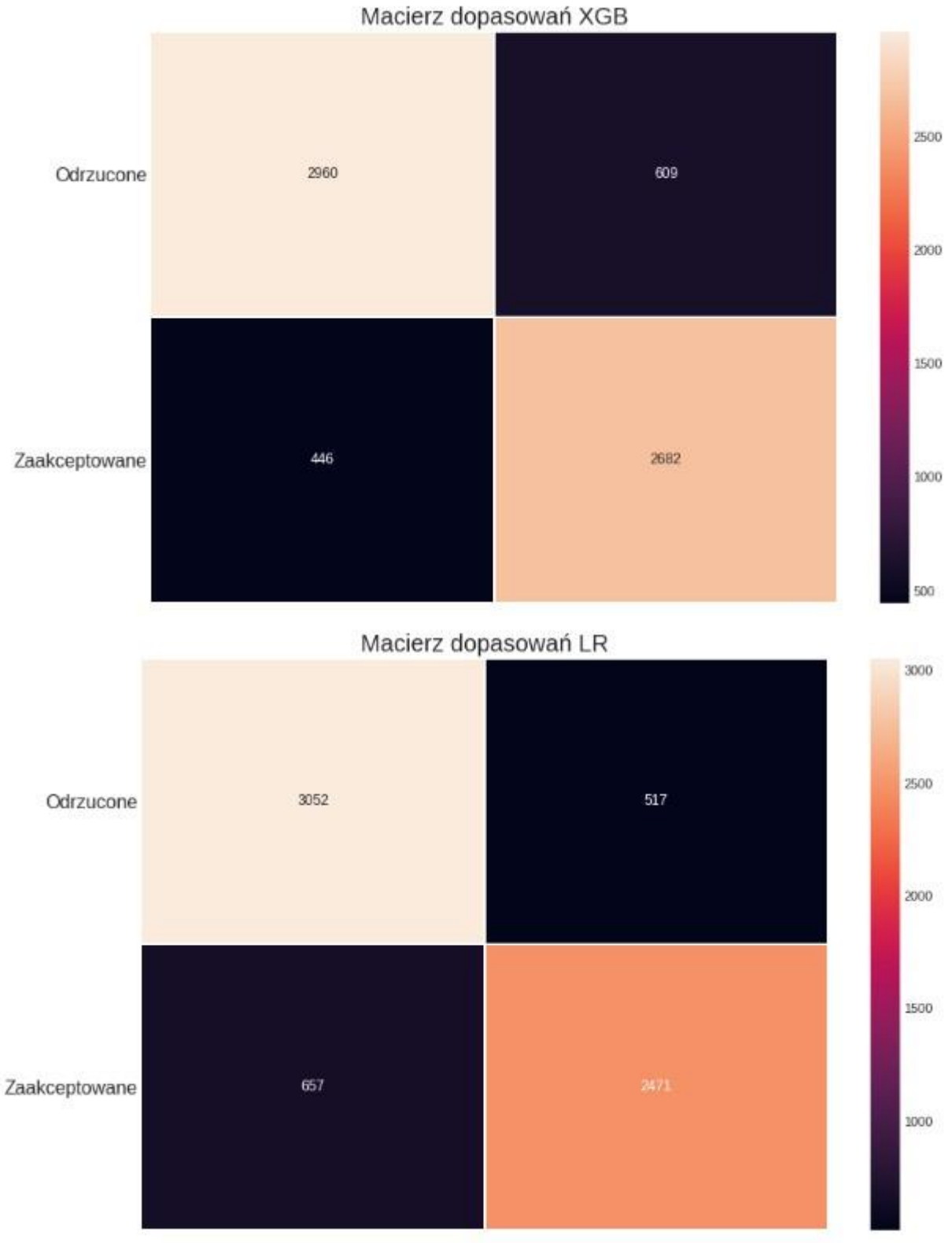
W ramach preprocesingu danych zauważylismy, że niektóre parametry mają bardzo nieregularny rozkład np. „loan” ma 87% odpowiedzi „no” i tylko 13% „yes”. Powoduje to konieczność zastosowania takiego podziału zbiorów, żeby ta proporcja była „mozliwie” dobrze zachowana. Dodatkowo przed analizą i uczeniem należało przekonwertować wartości w postaci opisowej np. „student” na wartości liczbowe i przypisać je odpowiednio do siebie. Ponadto wykonano częściowe czyszczenie zbioru np. z nieznanych form zatrudnienia „unknown”.

Podział zbioru danych na zbiory treningowy oraz testowy został wykonany w stosunku 8:2 tzn. zbiór testowy był 0.2 * zbiór pierwotny. Ponadto wprowadzono również walidację krzyżową poprzez podział zbioru na 20 podzbiorów.

Strojone parametry to:

- n_splits – ilość podzbiorów wytworzonych ze zbioru głównego, służące do walidacji krzyżowej
- test_size – wielkość zbioru testowego
- random_state – wartosc inicjująca wewnętrzne RNG
- cv – generator podziałów walidacji, zrównany z wartością n_splits

3. Podsumowanie pracy



Wyniki wskazały, że trochę lepiej poradził sobie XGBoost (wzmacnianie gradientowe), ale różnica przy przyjętych parametrach modelu oraz przy opracowywanym zbiorze danych jest niewielka.

Drugim ważnym spostrzeżeniem jest fakt, że parametr „duration” wraz z „contact” miały kluczowe znaczenie dla klasyfikatora.

Przy wykonaniu dokumentacji korzystaliśmy z:

1. Hands on Machine Learning with Scikit Learn and Tensorflow by Aurelien Geron