

PSZT - Uczenie Maszynowe

Stawczyk Przemysław 293153, Piotr Zmysłony 268833

Contents

1	Opis zagadnienia	2
1.1	Treść zadania	2
1.2	Narzędzia	2
2	Opis preprocesingu i modelowania	2
2.1	Opis danych wejściowych	2
2.2	Analiza zbioru danych	2
2.2.1	Brakujące dane	2
2.2.2	Zbalansowanie danych	3
2.3	Przepływ Danych	3
2.3.1	Wizualizacja Przepływu Danych	3
3	Modele	3
3.1	Parametry Modeli	4
4	Wyniki Eksperymentu	4
4.1	Wykresy	4
4.2	Interpretacja	4

1 Opis zagadnienia

1.1 Treść zadania

Przedstawić wyniki analizy zbioru *Bankruptcy*, opisać procedurę eksperymentalną uczenia maszynowego z wykorzystaniem algorytmów *random forest* i *k-najbliższych sąsiadów* oraz opisać wyniki strojenia parametrów powyższych algorytmów.

1.2 Narzędzia

Skrypty oraz algorytm zostały zaimplementowane w Pythonie 3. Wykorzystano biblioteki: *imblearn.over_sampling.SMOTE*, *sklearn*, *numpy*, *matplotlib*, *scipy.io*, *impyute*.

2 Opis preprocessingu i modelowania

2.1 Opis danych wejściowych

Jako dane wejściowe posiadamy 5 plików *.arff*, z których każdy zawiera ekonomiczne wskaźniki z systemu EMIS na temat polskich firm i ich klasyfikację względem tego, czy firmy zbankrutowały po *n* latach od roku, w którym zostały zebrane dane. Liczba *n* lat jest różna dla każdego z plików, od 1 do 5, a każda firma opisana jest przez 64 atrybuty, od *X1* do *X64*.

2.2 Analiza zbioru danych

2.2.1 Brakujące dane

Zaczęliśmy od analizy brakujących danych w wierszach. Jak widać w poniższych wynikach w większości zbiorów około połowa wierszy ma brakujące pola.

	1 rok	2 lata	3 lata	4 lata	5 lat
długość	7027	10173	10503	9792	5910
pełne wiersze	3194	4088	4885	4769	3031
wiersze wybrakowane	3833	6085	5618	5023	2879

Następnie przeprowadziliśmy analizę rozkładu brakujących danych w kolumnach i wierszach korzystając z biblioteki pythona *missingno* [fig 1-5]

Figure 1: 1 rok

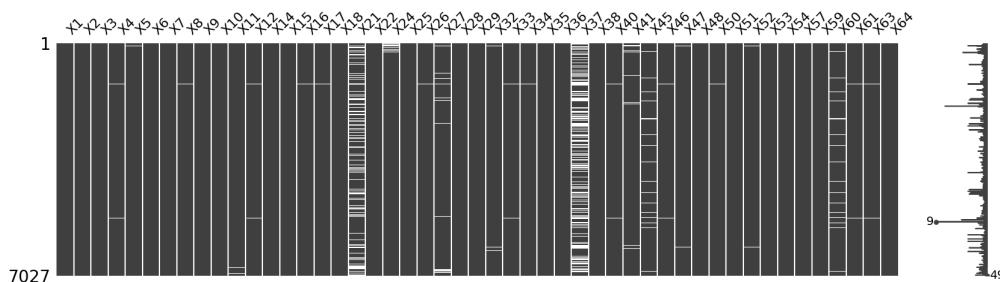


Figure 2: 2 lata

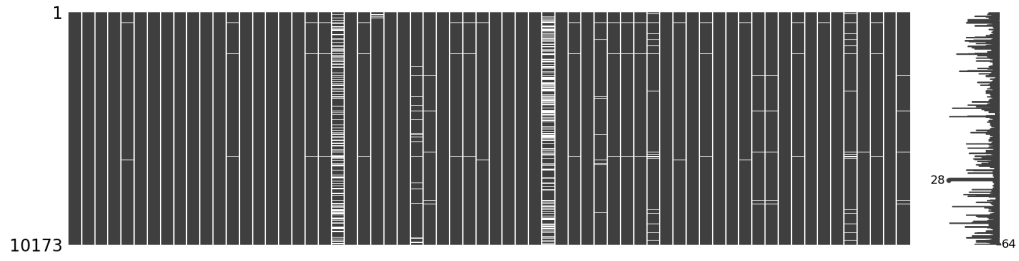


Figure 3: 3 lata

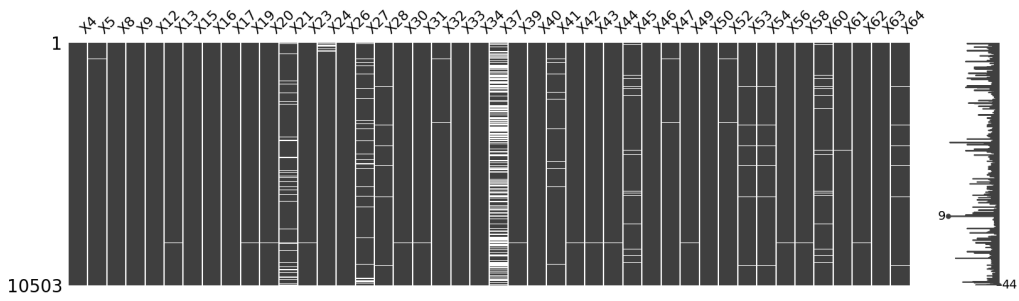


Figure 4: 4 lata

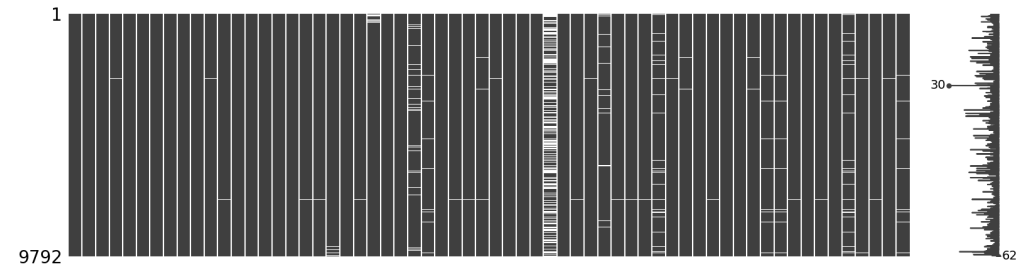
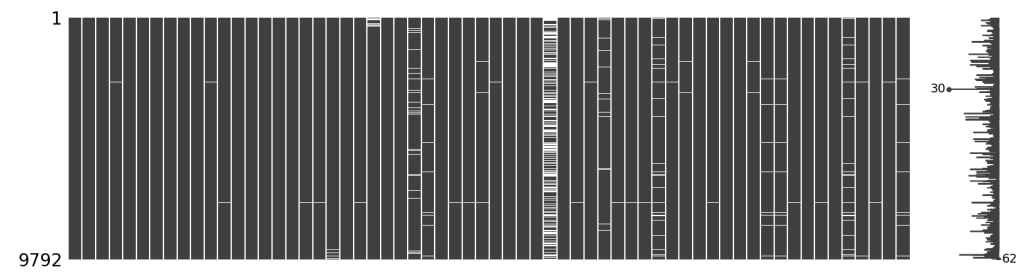


Figure 5: 5 lat



Jak widać większość brakujących danych jest w kolumnie *X37*. Kolumna *X21* ma brakujące w niektórych ale nie wszystkich latach.

Trudno nam było ocenić jaki charakter mają braki w tych danych, czy są skorelowane w wartościami w innych kolumnach czy zupełnie losowe. Wierszy z brakującymi danymi jest około połowy lub więcej. Aby nie odrzucać tak dużej liczby krotek zdecydowaliśmy się interpolować brakujące dane.

W tym celu wybraliśmy 4 metody:

1. Wstawianie średniej w danej kolumnie (*Jako punkt odniesienia*)
2. K najbliższych krotek
3. Spodziewanej Maksymalizacji (*Expected Maximalisation*)
4. Algorytm MICE

2.2.2 Zbalansowanie danych

Dokonałiśmy analizy ile z poszczególnych rekordów należy do klas klasyfikacyjnych

<i>Czy zbankrutowano:</i>	<i>rok 1</i>	<i>rok 2</i>	<i>rok 3</i>	<i>rok 4</i>	<i>rok 5</i>
<i>Nie</i>	6756	9773	10008	9277	5500
<i>Tak</i>	271	400	495	515	410
<i>procent większości</i>	3.857 %	3.932 %	4.713 %	5.259 %	6.937 %

Dane w zbiorach są mocno niezbalansowane dlatego zdecydowaliśmy się na interpolację korzystając z metody *SMOTE* (*Synthetic Minority Over Sampling Technique*)

2.3 Przepływ Danych

Po powyższej analizie zdecydowaliśmy o następującym przepływie oryginalnych danych do konstrukcji modeli.

Walidacji modeli planujemy dokonać korzystając K-krotnej walidacji krzyżowej.

2.3.1 Wizualizacja Przepływu Danych

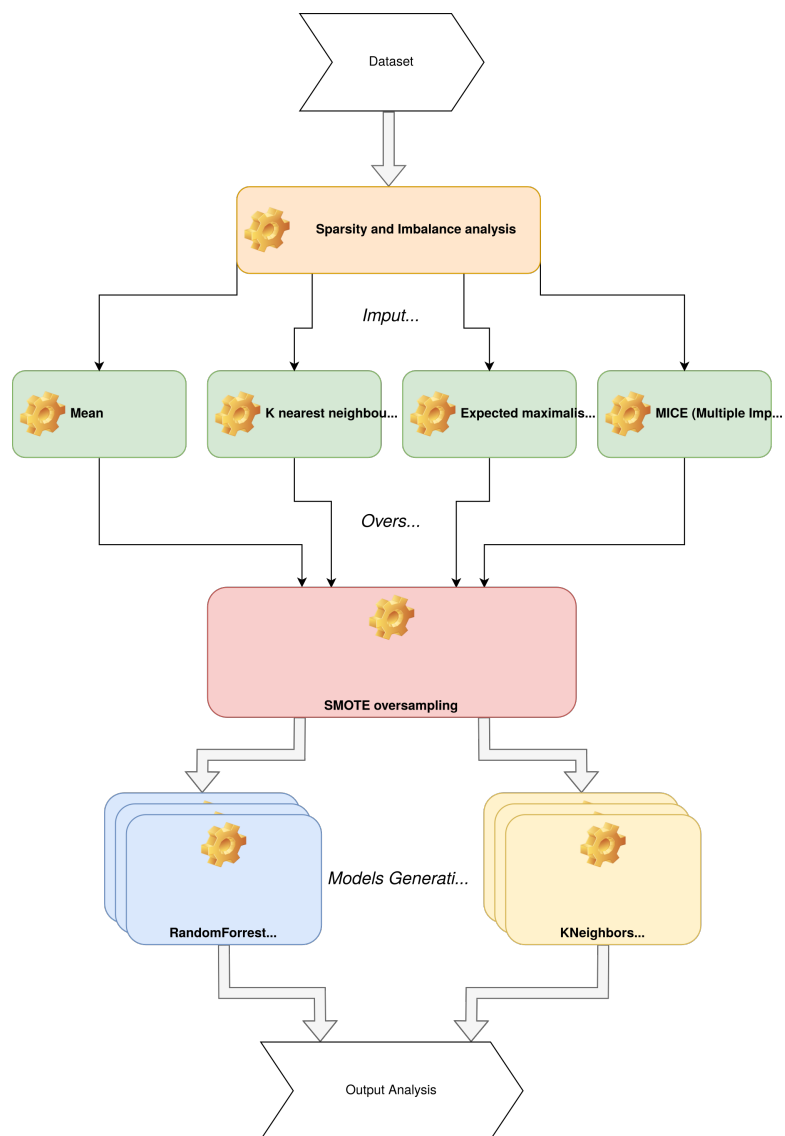
3 Modele

Zgodnie z poleceniem wykorzystaliśmy algorytmy tworzenia modeli :

- Las Losowy [*RF - Random Forrest*]
- K Najbliższych sąsiadów [*KNN - K Nearest Neighbors*]

Implementacje wymienionych algorytmów pochodzą z biblioteki *sklearn*.

Figure 6: Przepływ Danych



3.1 Parametry Modeli

4 Wyniki Eksperymentu

4.1 Wykresy

4.2 Interpretacja