



Politechnika
Wrocławska

POLITECHNIKA WROCŁAWSKA
WYDZIAŁ INFORMATYKI I TELEKOMUNIKACJI
KIERUNEK TELEINFORMATYKA

Metody Sztucznej Inteligencji - Projekt

Implementacja algorytmu oversamplingu ADASYN

Autorzy:
Kałwa Weronika 263876
Postawa Sandra 263826
Zych Zuzanna 263882

1 Wstęp i przegląd literatury

1.1 Omówienie działania algorytmu oversamplingu ADASYN

Algorytm ADASYN (Adaptive Synthetic Sampling - Adaptacyjne Syntetyczne Próbkowanie) jest zaawansowaną techniką oversamplingu, mającą na celu zwiększenie dokładności klasyfikatorów w problemach związanych z niebalansowanymi zbiorami danych. Główna idea algorytmu opiera się na generowaniu syntetycznych przykładów dla mniejszościowej klasy w taki sposób, aby równoważyć zbiór danych, skupiając się przede wszystkim na tych obszarach, gdzie granica decyzyjna między klasami jest niejasna. W porównaniu do innych metod, takich jak SMOTE, ADASYN dąży do bardziej zróżnicowanego i adaptacyjnego generowania próbek, poprawiając w ten sposób klasyfikację w trudniejszych obszarach [1].

1.2 Cel projektu

Celem projektu jest implementacja algorytmu ADASYN w kontekście balansowania zbioru danych w problemach klasyfikacji z niezrównoważonymi klasami. Projekt skupi się na zrozumieniu działania algorytmu, jego implementacji praktycznej oraz ocenie efektywności w porównaniu z innymi technikami oversamplingu.

1.3 Przegląd literatury

1.3.1 Omówienie istniejących metod

Oprócz ADASYN istnieją także inne metody oversamplingu, takie jak:

- SMOTE (Synthetic Minority Over-sampling Technique) [2]: Jest to technika statystyczna zwiększająca liczbę przypadków w zestawie danych w zrównoważony sposób. Składnik działa przez wygenerowanie nowych wystąpień z istniejących przypadków mniejszości, które są dostarczane jako dane wejściowe. Ta implementacja programu SMOTE nie zmienia liczby przypadków większościowych.
- BorderlineSMOTE [3]: Jest to modyfikacja metody SMOTE, ogranicza tworzenie nowych obiektów jedynie do granicy między przykładami z obydwu klas, co ma na celu zmniejszenie ryzyka przeuczenia.
- SVM-SMOTE [4]: Jest to metoda nadpróbkiwania danych, która używa wektorów nośnych z SVM do generowania syntetycznych próbek klasy mniejszościowej przez interpolację między wektorami nośnymi a ich najbliższymi sąsiadami tej samej klasy, zwiększając liczbę danych na granicy decyzyjnej.
- BAGGING [5] polega na wykorzystaniu zbioru trenującego algorytmu klasyfikacji. Tworzony jest zbiór klasyfikatorów, z których każdy wykorzystuje algorytm i trenowany jest na zbiorze trenującym. Zbiór trenujący powstaje poprzez wylosowanie przykładów (ze zwracaniem) ze zbioru. Losowanie odbywa się zgodnie z rozkładem jednostajnym. Liczności zbioru. Każdy klasyfikator przydziela kategorię przykładowi. Ostateczna kategoria jest tą, która najczęściej była proponowana przez klasyfikatory
- NearMiss [6] to technika undersamplingu. Jej celem jest zrównoważenie rozkładu klas poprzez losowe eliminowanie przykładów z klasy większościowej. Gdy instancje dwóch różnych klas są bardzo blisko siebie, usuwamy instancje z klasy większościowej, aby zwiększyć odstęp między tymi klasami. Aby zapobiec problemowi utraty informacji w większości technik undersamplingu, szeroko stosuje się metody bliskich sąsiadów.
- K-Nearest Neighbors [7] tworzy wyimaginowaną granicę, aby sklasyfikować dane. Gdy do przewidywania dodawane są nowe punkty danych, algorytm dodaje ten punkt do najbliższego punktu granicy

1.3.2 Metoda referencyjna

Głównym celem projektu jest zaimplementowanie algorytmu oversamplingu ADASYN za pomocą istniejących funkcji i implementacji. Do badania zostaną wykorzystane dane wygenerowane dane syntetyczne.

W projekcie metodą, do której porównywane będą wyniki, będzie zaimportowany ADASYN, SMOTE oraz BorderlineSMOTE.

2 Bibliografia

Literatura

- [1] <https://medium.com/@ruinian/an-introduction-to-adasync-with-code-1383a5ece7aa>
- [2] <https://learn.microsoft.com/pl-pl/azure/machine-learning/component-reference/smote?view=azureml-api-2>
- [3] <https://pb.edu.pl/oficyna-wydawnicza/wp-content/uploads/sites/4/2021/12/Modelowanie-i-optimalizacja-1.pdf>
- [4] <https://www.blog.trainindata.com/oversampling-techniques-for-imbalanced-data/>
- [5] https://repo.pw.edu.pl/docstore/download/WEiTI-ab4d82b3-a859-462f-a20b-2823dec969b1/pandrusz_Metauczenie+a+mo%C5%BClwo%C5%9B%C4%87+poprawy+skuteczno%C5%9Bci+klasyf
- [6] <https://www.geeksforgeeks.org/ml-handling-imbalanced-data-with-smote-and-near-miss-algorithm-in-python/>
- [7] <https://www.geeksforgeeks.org/regression-using-k-nearest-neighbors-in-r-programming/?ref=lbp>