Metody Sztucznej Inteligencji - Projekt.

Paweł Polski, Michał Włosek, Dariusz Szymula

28 kwietnia 2022

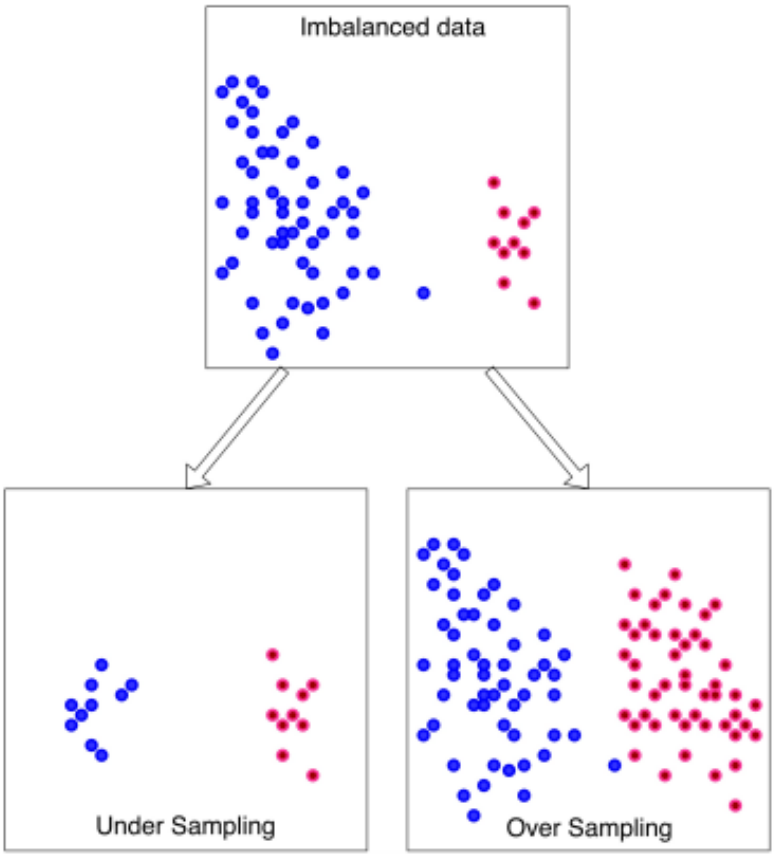
Streszczenie

Przetwarzanie wstępne w dużych zbiorach danych.

1 Opis problemu.

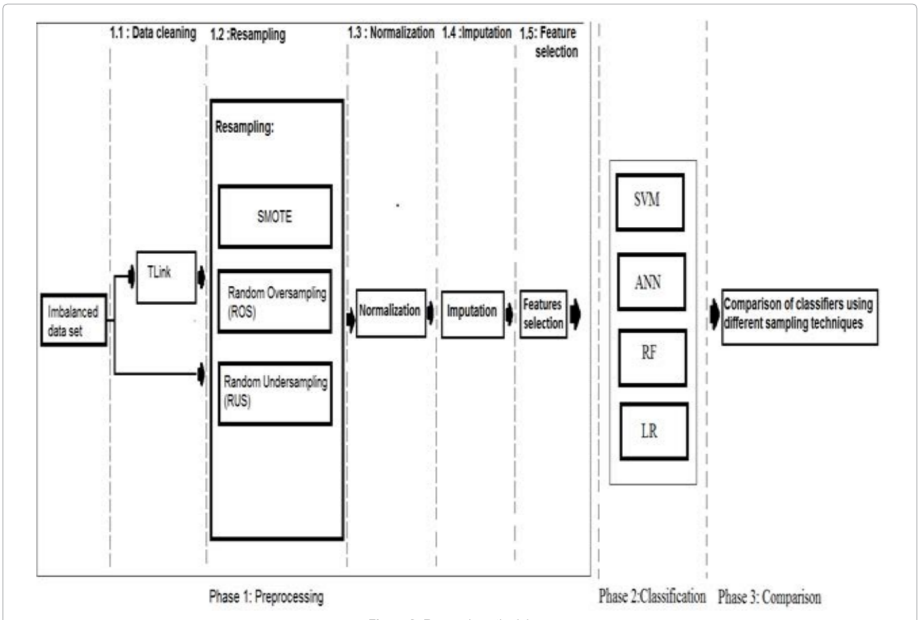
Pozyskiwanie wiedzy z rzeczywistych zbiorów danych w dzisiejszych czasach jest trudne, ponieważ dane te są często wielowymiarowe, wieloklasowe oraz niezrównoważone. W uczeniu nadzorowanym w celu poprawnego nauczenia modelu lub klasyfikatora potrzebujemy danych, które ten proces umożliwią. Aby przystosować obecnie gromadzone zbiory wprowadzono metody, które umożliwiają ich zrównoważenie. [10] Powszechnie stosowane są techniki próbkowania danych, czyli podpróbkowania (undersampling) lub nadmiernego próbkowania(oversampling)[9]. Powodują one zmniejszenie instancji klas w podejściu podpróbkowania oraz powstawania klas mniejszościowych w nadmiernym próbkowaniu. Każda z tych technik wnosi pewne zmiany do wcześniej posiadanych już zbiorów. Różnica pomiędzy undersamplingiem a oversamplingiem została przedstawiona na rysunku nr 1 i nr 2.

Rysunek 1: Różnica pomiędzy undersamplingiem a oversamplingiem [10].



Rysunek 2: Różnica pomiędzy undersamplingiem a oversamplingiem[2].

W zbiorach dużych danych dąży się do zmniejszenia danych bez utraty danych informacyjnych. Pozwala to zmniejszyć wymagania odnośnie systemów, które przetwarzają te dane. Z metod przetwarzania wstępnego te założenia spełnia undersampling i to na nim się skupimy w naszej pracy. Proces przetwarzania danych z zaznaczonym miejscem, w którym odbywa się przetwarzanie wstępne pokazano na rysunku nr 3.



Rysunek 3: Proces przetwarzania wstępnego.

1.1 Duże zbiory danych (Big Data).

1.1.1 Big Data

Terminem tym określa się zbiory danych tak duże, przy których metody analizy tradycyjnej nie zwracają oczekiwanych wyników bądź zastosowanie ich jest wręcz niemożliwe. Big Data to również dane, które nie mogą być obsługiwane i przetwarzane przez większość obecnych systemów lub metod informatycznych, ponieważ oprócz swojego rozmiaru uniemożliwiające załadowanie ich na pojedyńczą instancję urządzenia, to również wiele tradycyjnych metod analizy danych stowrzonych dla scentralizowanego procesu przetwarzanua ich mogą być niemożliwe w użyciu. Dane wieloskalowe wykorzystuje się zarówno w środowisku komercyjnym, ale również w niekomercyjnym. Na przykładzie firm komercyjnych analiza dużych zbiorów danych może posłużyć do zwiększenia wiedzy na temat personelu, procesów produkcyjnych, produktów oraz klientów[8][6][12]. Z drugiej strony organizacje rządowe wykorzystują analizę Big Data m.in w celu wykrywania oszustw, zwiększenia płynności finansowych oraz bezpieczeństwa. Różne organizacje, które mają dostęp do dużych zbiorów danych na konkretny temat mogą wykorzystać wyniki ich analizy w celu opracowania strategii rozwoju na przyszłość. Istnieje definicja zwana również 3V, która wyjaśnia czym są Big Data: objętość, szybkość, różnorodność. Definicja 3V oznacza, że rozmiar

danych jest duży, dane będą tworzone szybko, a dane będą istniały odpowiednio w wielu typach i pochodzą z różnych źródeł. Później wykazano, że te pojęcia nie są wystarczające aby wyjaśnić czym są Big Data, dlatego dodano do nich prawdziwość, trafność, wartość, zmienność, miejsce, słownictwo i niejasność, aby uzupełnić wyjaśnienie dużych zbiorów danych [11].

Głównymi kategorami danych są m.in:

- **strukturalne** - to dane, które zależą od modelu danych i znajdują się w stałym polu w rekordzie,
- **niestrukturalne** - to dane, które nie są łatwe do dopasowania do modelu danych, ponieważ zawartość jest zależne od kontekstu lub zmienne,
- **język naturalny** - szczególny rodzaj danych nieustrukturyzowanych wymagający wiedzy zarówno na temat danych jak i ligwistyki,
- **dane generowane maszynowo** - to informacje, które są automatycznie tworzone przez komputer, proces, aplikacja lub inna maszyna bez interwencji człowieka,
- **oparte na grafach** - dane wskazują na matematyczną teorię grafów, czyli matematyczną strukturę do modelowania relacji między obiektami,
- **filmy, obrazy, dźwięk** - trudne w analizie dane, ponieważ łatwe w analizie dla człowieka rozpoznawanie obiektów stanowi trudne zadanie dla maszyny,
- **strumieniowane** - dane, które strumieniowo wpływają do systemu

Z uwagi, że tworzenie danych jest dużo łatwiejsze niż znajdowanie w nich przydatnych rzeczy wystąpiły problemy z analizowaniem danych wielkoskalowych.

- **Nieskalowalne i scentralizowane** - większość metod analizy danych nie jest przeznaczona do działania na dużych i złożonych zbiorach danych. Z tego powodu metody te nie mają atrybutu skalowalności. A przede wszystkim projektując je zakładano, że wszystkie dane znajdą się w pamięci maszyny.
- **Niedynamiczne** - większość tradycyjnych metod nie jest przystosowana do dynamicznej analizy danych wejściowych, dostosowywania się do różnych sytuacji.
- **O jednolitej strukturze danych** - większość problemów z analizą danych, zakłada, że format danych wejściowych będzie taki sam. W Big Data pojawia się problem różnorodności danych wejściowych czyli ich niezbalansowania.

W celu rozwiązania tego problemu pojawiły się metody takie jak próbkowanie (podział metod próbkowania opisany jest w punkcie 1.2), kondensacja

danych, dziel i zwyciężaj, przetwarzanie rozproszone oraz wiele innych. Głównym zadaniem tych metod jest możliwość analizy dużych zbiorów danych w rozsądnym czasie w celu wydobycia interesującej wiedzy. Ważną kwestią jest wstępne przygotowanie danych do dalszej analizy. Część badań koncentruje się na zmniejszeniu złożoności danych wejściowych, ponieważ nawet najbardziej zaawansowana technologia komputerowa w większości przypadków nie jest w stanie wydajnie przetworzyć całych danych wejściowych przy użyciu jednej maszyny. Wykorzystanie wiedzy domenowej do zaprojektowania operatora przetwarzania wstępnego jest jednym z rozwiązań dla dużych zbiorów danych. Często też wykorzystuje się systemy chmurowe w celu wstępnego przetworzenia surowych danych [7].

Cecha	Small Data	Big Data
Objętość	Ograniczona-duża	Bardzo duża
Szybkość	Powolna, zamrożone ramki/pakiety	Szybka, ciagła
Różnorodność	Ograniczona	Szeroki zakres
Ograniczenia	Próbki	Cała populacja
Rozdzielczość i indeksykowalność	Ciągła i słaba, surowa i silna	surowa i silna
Relacyjność	Słaba-silna	Silna
Rozszerzalność i skalowalność	Mała-średnia	Duża

Tablica 1: Porównanie Small Data oraz Big Data

1.2 Metody próbkowania

1.2.1 Oversampling.

Metody oversamplingu polegają na zrównoważeniu rozkładu prawdopodobieństwa apriori pomiędzy klasami. Dzięki wiedzy na temat klas problemu możemy przy pomocy określonego algorytmu stworzyć dane syntetyczne dla klasy mniejszościowej, które rozkład apriori pomiędzy klasami. Poniżej znajdują się najpopularniejsze metody [4]:

- **Borderline-SMOTE** - algorytm ten wychodzi z założenia, że próbki znajdujące się daleko od granicy mogą w niewielkim stopniu zwiększyć powodzenie klasyfikacji. Technika ta identyfikuje próbki znajdujące się w pobliżu granicy.

- **AHC** - Ta metoda używa klasteryzacji do generowania danych syntetycznych do zrównoważenia rozkładu danych między klasami. Do tego celu został użyty algorytm centroidów.

- **ADASYN** - Główna idea tego algorytmu wywodzi się z wykorzystania rozkładu ważonego w zależności od rodzaju przykładów mniejszościowych zgodnie z ich zdolnością do uczenia się. Ilość danych syntetycznych dla każdego z nich jest związana z poziomem trudności każdego przykładu mniejszościowego.

- **DBSMOTE** - Ten algorytm opiera się klastrowaniu w oparciu o gęstość. Dane syntetyczne generowane są po najkrótszej ścieżce od każdej mniejszościowej instancji do pseudocentroidu klastra klasy mniejszościowej.

1.2.2 Undersampling.

Metody Undersamplingu mają za zadanie zmniejszyć ilość danych klasy mniejszościowej bez utraty istotnych informacji. Takie działanie jest korzystne w przypadku kiedy mamy do czynienia z dużymi zbiorami danych i ich przetwarzanie jest bardzo kosztowne obliczeniowo. Poniżej przedstawiamy najpopularniejsze metody:

- **Random under-sampling(RUS)** - w tej metodzie nieheurystycznej równoważenie zbiorów danych odbywa się poprzez losowe usuwanie niektórych próbek klas większościowych. Random under-smapling polega na losowym wybieraniu przykładów z klasy większości i usuwaniu ich ze zbioru danych uczących. W tej metodzie instancje klas większości są losowo odrzucane, aż do osiągnięcia bardziej zrównoważonego rozkładu. Obok random oversamplingu jest to druga metoda ńaiwnego próbkowania ponownego", ponieważ nie zakłada niczego na temat danych oraz nie są używane żadne heurystyki. Z tego powodu metoda ta jest prosta do wdrożenia oraz szybka do wykonania co jest ważną cechą w przypadku dużych i złożonych zbiorów danych. Metoda ta może być zastosowana do klasyfiakcji binarnej oraz problemów klasyfikacji wieloklasowej z jedną lub większą liczbą klas większościowych lub mniejszościowych. Zmiana rozkładu klas ma wpływ jedynie na zestaw danych uczących. Nie stosuje się go do testowego zestawu danych używanego do oceny wydajności modelu.

- **Condensed nearest neighbor rule(CNN)** - metoda polega na eliminacji próbek klasy większościowej, które są odległe od granicy decyzji, ponieważ te próbki możemy uznać za mniej znaczące w procesie nauki. Najpierw losowo wybierana jest próba z klasy większościowej i utworzony podzbiór z wszystkimi próbkami klas mniejszościowych. Następnie 1-NN jest używany w tym podzbiorze, aby sklasyfikować inne próbki z klasy większościowej. Każda błędnie sklasyfikowana próbka z klasy większościowej jest brana do ponownego utworzenia zestawu danych próbkowanych.

Algorytm CNN[5]:

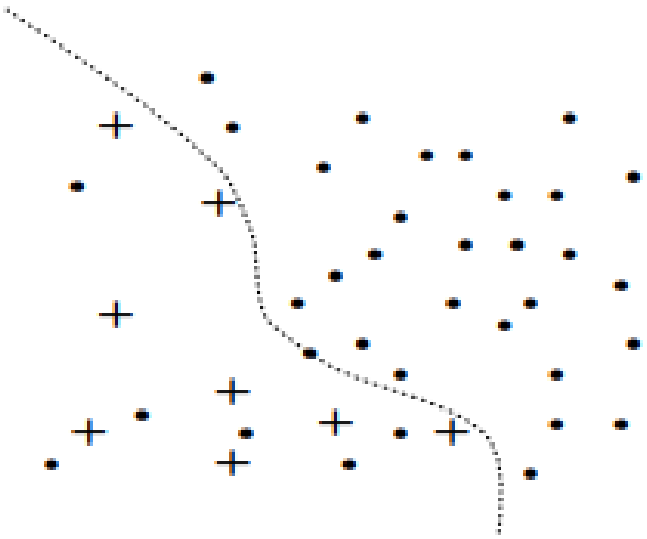
1. Pierwsza próbka jest umieszczana w zbiorze STORE.
2. Druga próbka jest klasyfikowana zgodnie z regułą NN, stosując jako odniesienie zawartość STORE. Jeśli druga próbka zostanie sklasyfikowana poprawnie wówczas jest umieszczana w zbiorze GRABBAG, jeśli nie to jest umieszczana w STORE.
3. Postępowanie indukcyjne, i-ta próbka jest klasyfikowana przez obecną zawartość STORE. Jeżeli sklasyfikuje poprawnie, to zostanie umieszczona w zbiorze GRABAG, w przeciwnym przypadku trafi do STORE.
4. Po jednym przejściu przez oryginalne próbki, procedura jest powtarzana w pętli aż do momentu kiedy zbiór GRABBAG zostanie wyczerpany, lub jeśli jedna iteracja przez zbiór GRABBAG nie powoduje przeniesienia próbek do zbioru STORE.
5. Finalna zawartość STORE służy jako punkty odniesienia dla reguły najbliższego sąsiada. Natomiast zawartość GRABBAG jest odrzucana.

- **Tomek links(TL)** - metoda jest przeciwieństwem metody CNN. [3] Próbki graniczne mogą być traktowane jako niebezpieczne, ponieważ niewielka zmiana może spowodować przypisanie ich do niewłaściwej klasy.Każda próbka służy do znalezienia innej próbki, która ma minimalną odległość między nimi. Jeżeli te dwie próbki znajdą się w różnych klasach, próbka z klasy większościowej zostanie usunięta. Metoda ta może spowodować wzrost obszaru decyzyjnego. Metoda ta jest rozszerzeniem metody Nearest-Neighbour Rule (NNR).
  Działanie algorytmu:

  1. Niech x będzie instancją klasy A a y instancją klasy B.
  2. Niech d(x,y) będzie odległością między x i y.
  3. (x,y) to T-link, jeśli w każdym przypadku z, $d(x,y)<d(x,z)$ lub $d(x,y)<d(y,z)$
  4. Jeśli jakiekolwiek dwa przykłady to T-link, to jeden z nich to szum inaczej oba przykłady znajdują się na granicy klas.

  Metoda T-link może być stosowana jako metoda kierowanego undersamplingu.

- **One-sided selection(OSS)** - metoda stosuje metody Tomek links, a następnie Condensed nearest neighbor rule. [13] Dzięki zastosowaniu tych dwóch technik, pozostałe próbki klasy większościowej są bardziej przydatne do nauki.



Rysunek 4: Rozkład danych niezbalansowanych

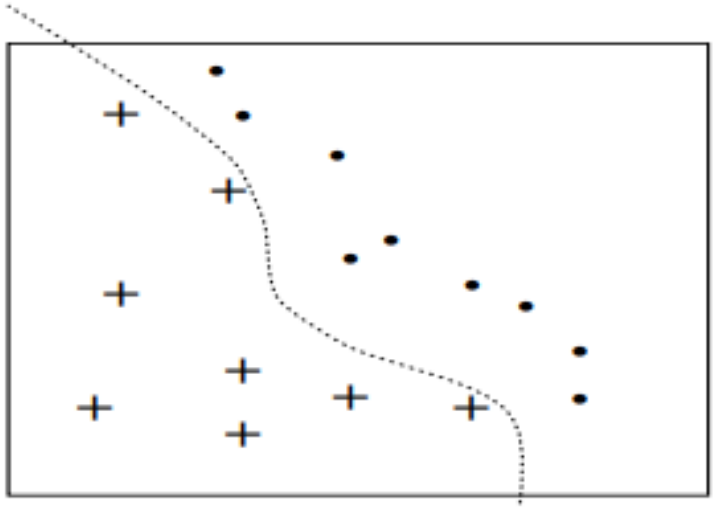
Rysunek nr 3 pokazuję, iż negatywne próbki mogą być podzielone na 4 części. 1. Próbki, które cierpią z powodu szumu etykiety klasy, na przykład próbka znajdująca się w lewym dolnym rogu.

2. Próbki będące na pograniczu mogą być niewiarygodne ponieważ niewielki szum może spowodować ich błędne sklasyfikowanie.

3. Próbki, które są zbędne a ich część może być reprezentowana przez inne punkty. Takim przykładem są punkty w prawym górnym rogu.

4.Bezpieczne przykłady które warto zachować do dalszych etapów klasyfikacji.

Zbędne przykłady nie wpływają negatywnie na poroces klasyfikacji, lecz zwiększają jej koszt. Rysunek nr 5 pokazuje usuwanie zbędnych próbek negatywnych.

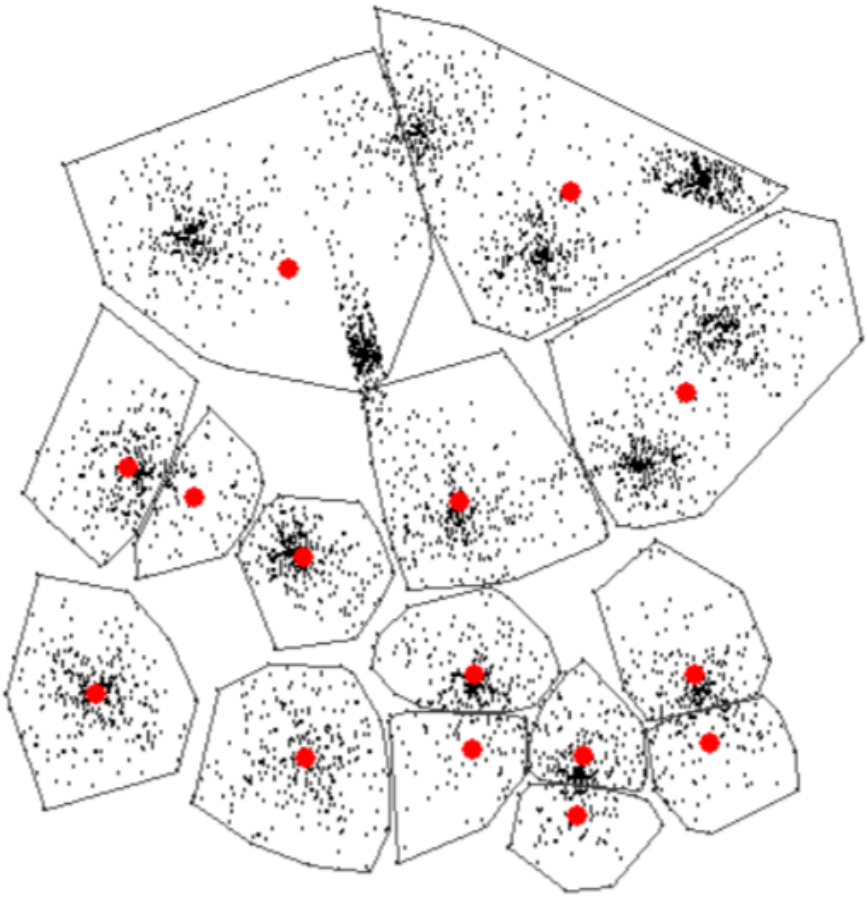


Rysunek 5: Po usunięciu zbędnych danych negatywnych

Korzystając z koncepcji Tomek links, bierzemy dwie próbki x i y, o różnych etykietach oraz oznaczamy dystans pomiędzy nimi.Para x i y jest oznaczana jako "Tomek links"jeżeli nie ma takich przykładów że,
(x,z)<(x,y) lub (y,z)<(y,x)
Próbę zmniejszenia liczby zbędnych przykładów można potraktować jako zadanie stworzenia spójnego podzbioru C zbioru uczącego S. Z definicji zbiór C zawierający się w S jest zgodny z S, jeśli użyty przez regułę 1-NN poprawnie sklasyfikuje przykłady w S. Należy zauważyć, iż każdy zbiór jest spójny sam w sobie. Nie zależy nam jednak na stworzeniu najmniejszego zbioru C. Wystarczy jedynie, iż zbiór wartości negatywnych wystarczająco się skurczy. W tym celu można użyć np. techniki Hart. Zaczynamy z jedną negatywną próbką oraz wszystkimi pozytywnymi próbkami umieszczonymi w C. Następnie za pomocą reguły 1-NN z przykładami w zbiorze C w celu ponowej reklasyfikacji zbioru S. Wtedy próbki, które zostały wcześniej błędnie pominięte zostaną dodane.

- **Neighborhood Cleaning Rule(NCL)**- wykorzystuję edytowaną regułę najbliższego sąsiada Wilsona (ENN), [13] aby usunąć niektóre próbki klasy większościowej. Początkowo, odnajdywanych jest trzech najbliższych sąsiadów.Jeżeli wybrana próbka należy do klasy większościowej, ale algorytm trzech najbliższych sąsiadó błednie je zakwalifikował, taka próbka zostanie usunięta. Jeżeli wybrana próbka należy do klasy mniejszościowej, ale trzej wybrani sąsiedzi do klasy większościowej, najbliżsi sąsiedzi zostaną usunięci[1].

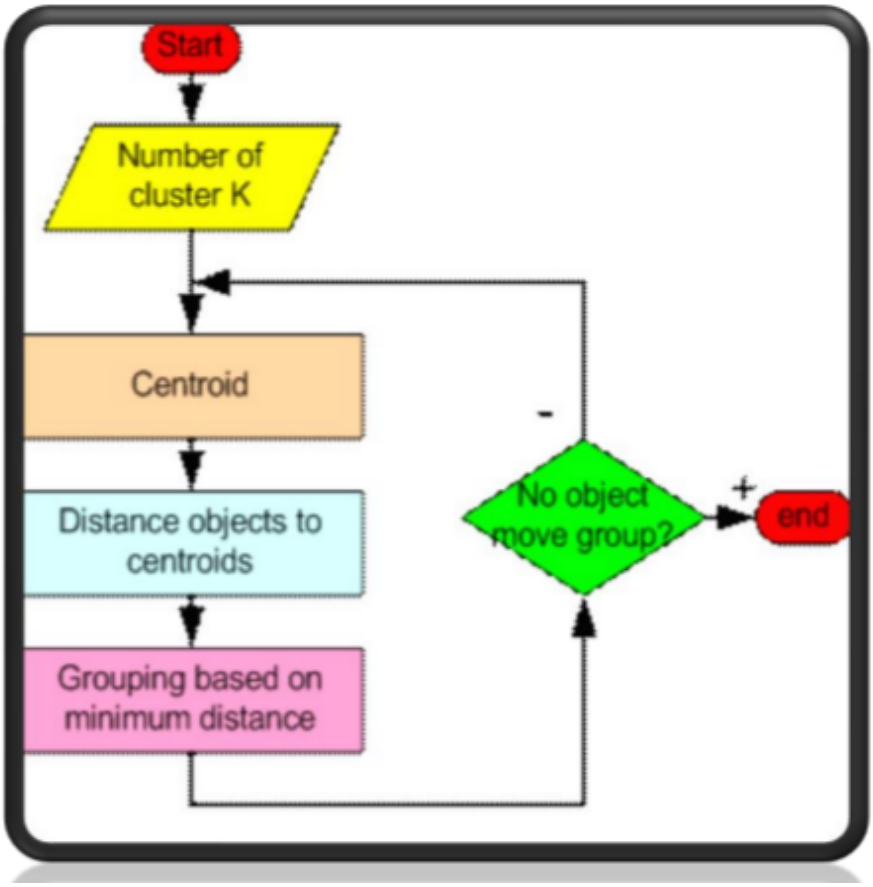
- **Cluster Centroid**- Klastrowanie jest to grupowanie obiektów o podobnych właściwościach w wyniku czego powstaje klaster.Celem działania algorytmu jest znalezienie skupienia dla znioru obiektów nie etykietowanych. Zasada działania algorytmu k-centroidów polega na znalezieniu k środków tak, aby suma odległości punktów do najbliższego centroida była jak najmniejsza [14]. Kroki postępowania algorytmu:
  1. W sposób losowy zostaje wybranych k centrum centroidów.
  2. Każdy z istniejących obiektów zostaje przydzielony do najbliższego centroida.



Rysunek 6: Przydzielenie do centroidów

3. Wyznaczony zostaje nowy układ centroidów.
4. Krok 2 zostaje powtórzony do momentu, aż przestanie być zauważalna poprawa jakości.

Opisany schemat algorytmu został przedstawiony na Rysunku nr 7.



Rysunek 7: Schemat działania algorytmu

2 Wybrane algorytmy.

Po dogłębnym przeanalizowaniu tematu uznaliśmy, że do dużych zbiorów danych pasuje idealnie metoda udersamplingu. Dzięki niej zmniejszymy ilość danych, i tak jest ich bardzo dużo. Redukcja danych może odbyć się bez większych strat informacyjnych ponieważ dane często się duplikują. Metody, które wybraliśmy to:

- Random Undersampling
- Cluster Centroids

Wybraliśmy dwie metody, które są zróżnicowane. RUS wybraliśmy z powodu prostoty działania. Metodę Cluster Centroids wybraliśmy ze wzgledu na to, że jest bardziej zaawansowana i powinna teoretycznie lepiej wybierać dane, które można usunąć. Metoda ClusterCentroids korzysta z algorytmu k-means w celu redukcji liczby próbek w klasie większościowej. Dodatkowo stworzymy własny algorytm, którego działanie będzie opierać się na tworzeniu klastrów przy pomocy algorytmu DBSCAN. Postaramy się ulepszyć sposób tworzenia klastrów. Sprawdzimy jak te algotrytmy radzą sobie na różnych zbiorach dużych danych oraz porównamy je ze sobą. Dane będą niezbilansowane, lecz stopień ich niezbilansowania będzię różny.

3 Hipoteza.

Zmniejszenie liczby danych w klasie większościowej w dużych zbiorach danych nie powoduje znacznego pogorszenia predykcji w procesie inferencji. Metoda pozwalająca w jednoznaczny sposób określić dane informatywne może w znaczący sposób zwiększyć skuteczność predykcji.

4 Plan eksperymentu.

4.1 Research questions.

- Czy za pomocą wybranych przez nas algorytmów możliwe jest zbalansowanie zbiorów danych, tak aby ograniczyć wielkość kolekcji danych bez straty na ich jakości?
- Jak modyfikacja klasteryzacji w procesie undersampligu wpłynie na jakość predykcji?
- Jak prostszy algorytm jakim jest RUS wypadnie w porównaniu z bardziej zaawansowanymi algorytmami undersamplingu?
- Czy istnieje możliwość redukcji danych w klasie większościowej, bez straty informatywności tych danych?
- Jak stopien niezbilansowania danych wpływa na działanie metod undersamplingu?

4.2 Plan eksperymentu.

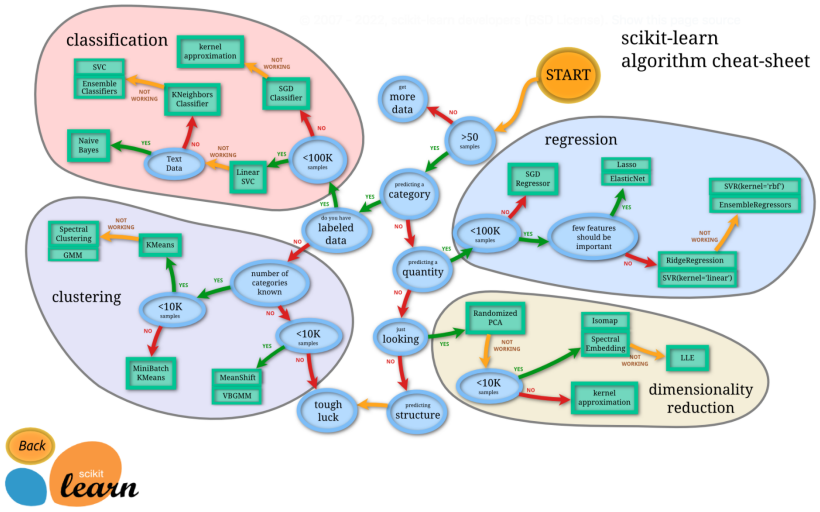
- Wybranie konkretnych algorytmów undersamplingu. W naszym przypdaku zostały wybrane dwa algorytmy, które zostały opisane we wcześniejszej części dokumentu.
  - **Random Undersampling.**
  - **Cluster Centrtoids.**

- Wybranie 10 rzeczywistych zbiorów dużych danych o różnym stopniu niezbalansowania, w celu przeprowadzenia na nich eksperymentu badawczego. Zbiory danych zostały wybrane sposród zbiorów publicznie dostępnych.

  Wybrane zbiory:

  - Hepmass Data Set - liczba instancji: 10500000, liczba atrybutów: 28, charakterystyka zbioru danych: wielowymiarowe, powiązane zadania: klasyfikacja
  - Bar Crawl: Detecting Heavy Drinking Data Set - liczba instancji: 14057567, liczba atrybutów: 3, charakterystyka zbioru danych: wielowymiarowe, szeregi czasowe, powiązane zadania: klasyfikacja, regresja
  - Heterogeneity Activity Recognition Data Set - liczba instancji: 43930257, liczba atrybutów: 16, charakterystyka zbioru danych: wielowymiarowe, szeregi czasowe, powiązane zadania: klasyfikacja, grupowanie
  - Human Activity Recognition from Continuous Ambient Sensor Data Data Set - liczba instancji: 13956534, liczba atrybutów: 37, charakterystyka zbioru danych: wielowymiarowe, sekwencyjne, szeregi czasowe, powiązane zadania: klasyfikacja
  - Detection of IoT botnet attacks N BaIoT Data Set - liczba instancji: 7062606, liczba atrybutów: 115, charakterystyka zbioru danych: wielowymiarowe, sekwencyjne, powiązane zadania: klasyfikacja, grupowanie
  - Kitsune Network Attack Dataset Data Set - liczba instancji: 27170754, liczba atrybutów: 115, charakterystyka zbioru danych: wielowymiarowe, sekwencyjne, szeregi czasowe, powiązane zadania: klasyfikacja, grupowanie, modelowanie przyczynowe (causal-discovery)
  - PPG-DaLiA Data Set - liczba instancji: 8300000, liczba atrybutów: 11, charakterystyka zbioru danych: wielowymiarowe, szeregi czasowe, powiązane zadania: regresja
  - SIFT10M Data Set - liczba instancji: 11164866, liczba atrybutów: 128, charakterystyka zbioru danych: wielowymiarowe, powiązane zadania: modelowanie przyczynowe (causal-discovery)
  - WESAD (Wearable Stress and Affect Detection) Data Set - liczba instancji: 63000000, liczba atrybutów: 12, charakterystyka zbioru danych: wielowymiarowe, szeregi czasowe, powiązane zadania: klasyfikacja, regresja
  - WISDM Smartphone and Smartwatch Activity and Biometrics Dataset Data Set - liczba instancji: 15630426, liczba atrybutów: 6, charakterystyka zbioru danych: wielowymiarowe, szeregi czasowe, powiązane zadania: klasyfikacja

- Wybranie konkretnych klasyfikatorów, na których zostanie przeprowadzony eksperyment badawczy.



Rysunek 8: Klasyfikatory polecane w obrębie dużych zbiorów danych w bibliotece scikit-learn.

Zgodnie z rysunkiem nr 8 dla dużych danych zalecane klasyfikatory to:

  - **SVC**
  - **Naive Bayes**
  - **KNeighbours Classifier**
  - **Linear SVC**

Klasyfikatory będą potrzebne do dokonania predykcji na zbiorze danych po undersamplingu.

- Implementacja własnego algorytmu undersamplingu w języku Python:
  - W celu ulepszenia procesu klasteryzacji operającej się na algorytmie k-means spróbujemy wykorzystać algorytm DBSCAN w celu stworzenia klastrów. Algorytm k-means dobrze się sprawdza w przypadkach gdy dane są wyraźnie od siebie oddzielone, lecz w przeciwnym przypadku lepiej radzi sobie algorytm DBSCAN.
  - Modyfikacja algorytmu DBSCAN w celu ulepszenia metody tworzenia klastrów. Klastry w algorytmie DBSCAN tworzone są w oparciu o gęstość, stąd gdy zostanie stworzony klaster o małej gęstosci próbek wewnątrz, to podział go na mniejsze klastry może spowodować poprawę działania algorytmu.

- Implementacja wybranych algorytmów undersamplingu, klasyfikacji oraz określonych wyżej zbiorów danych oraz przeprowadzenie eksperymentu w języku programowania Python:
  - Podział danych na zbiór testowy i treningowy przy wykorzystaniu wielokrotnej stratyfikowanej walidacji krzyżowej. W związku z faktem, że operacje odbywają się na dużych zbiorach danych zaproponowano walidację krzyżową w konfiguracji 5x2 tzn. 5 powtórzeń i 2 foldy gdyż nie ma potrzeby wykorzystywać walidacji 5x5.
  - Zapisanie wyników z operacji klasyfikacji do macierzy numpy. Takie podejście umożliwia wczytanie wyników w dowolnym programie oraz udostępnienie wyników dalej do dalszych badań.
  - Przeprowadzenie testów rankingowych, które umożliwią porównanie ze sobą określonych metod undersamplingu oraz określenie, która z metod jest najlepsza na zbiorach, na których zostanie przeprowadzone doświadczenie.
  - Prezentacja na wykresach radarowych wybranych metryk dla poszczególnych zbiorów danych oraz klasyfikatorów.
- Dokonanie analizy otrzymanych wyników. Odpowiedz na postawione pytania badawcze, próba wyciągnięcia wniosków na przyszłe eksperymenty badawcze.

Literatura

[1] Khafidurrohman Agustianto and Prawidya Destarianto. Imbalance data handling using neighborhood cleaning rule (ncl) sampling method for precision student modeling. In *2019 International Conference on Computer Science, Information Technology, and Electrical Engineering (ICOMITEE)*, pages 86–89, 2019.

[2] Md. Yasir Arafat, Sabera Hoque, Shuxiang Xu, and Dewan Md. Farid. An under-sampling method with support vectors in multi-class imbalanced data classification. In *2019 13th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, pages 1–6, 2019.

[3] T Elhassan and M Aljurf. Classification of imbalance data using tomek link (t-link) combined with random under-sampling (rus) as a data reduction method. *Global J Technol Optim S*, 1, 2016.

[4] Alberto Fernández, Salvador García, Francisco Herrera, and Nitesh V. Chawla. Smote for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. *J. Artif. Int. Res.*, 61(1):863–905, jan 2018.

[5] Peter Hart. The condensed nearest neighbor rule (corresp.). *IEEE transactions on information theory*, 14(3):515–516, 1968.

[6] Tawfiq Hasanin and Taghi Khoshgoftaar. The effects of random undersampling with simulated class imbalance for big data. In *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, pages 70–79, 2018.

[7] Haibo He and Yunqian Ma. *Imbalanced Learning: Foundations, Algorithms, and Applications*. Wiley-IEEE Press, 1st edition, 2013.

[8] Rob Kitchin and Gavin McArdle. What makes big data, big data? exploring the ontological characteristics of 26 datasets. *Big Data & Society*, 3(1):2053951716631130, 2016.

[9] Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550, 2009.

[10] Roweida Mohammed, Jumanah Rawashdeh, and Malak Abdullah. Machine learning with oversampling and undersampling techniques: Overview study and experimental results. pages 243–248, 04 2020.

[11] F.J. Ohlhorst. Big data analytics: Turning big data into big money. 2012.

[12] Chun-Wei Tsai, Chin-Feng Lai, H. C. Chao, and Athanasios V. Vasilakos. Big data analytics: a survey. *Journal of Big Data*, 2:1–32, 2015.

[13] Ginny Y. Wong, Frank H. F. Leung, and Sai-Ho Ling. An under-sampling method based on fuzzy logic for large imbalanced dataset. In *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1248–1252, 2014.

[14] Show-Jane Yen and Yue-Shi Lee. Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, 36:5718–5727, 01 2006.