Strona czasopisma: http://analit.agh.edu.pl/

K-Nearest Neighbours oraz K-Means: Zrozumienie zasad działania oraz zalet i wad

K-Nearest neighbours and K-Means: Understanding How It Works & Its Advantages and Disadvantages

Natalia Koziara

AGH Akademia Górniczo-Hutnicza, Wydział Inżynierii Materiałowej i Ceramiki, al. Mickiewicza 30, 30-059 Kraków

ABSTRAKT: Uczenie maszynowe jest metodą analizy danych, polegającą na automatyzacji modeli analitycznych, dzięki któremu możliwe jest uzyskanie dokładniejszych wyników. Wyróżnia się cztery rodzaje algorytmów – nadzorowane, półnadzorowane, nienadzorowane oraz wzmocnione, do których zalicza się między innymi algorytm k-najbliższych sąsiadów (K-Nearest Neighbors – KNN) oraz algorytm k-średnich (K-Means). Pierwszy z nich jest nieparametryczny, nadzorowanym klasyfikatorem uczenia się, natomiast drugi zaliczany jest do uczenia maszynowego bez nadzoru. Algorytm k-najbliższych sąsiadów używany jest w przypadku klasyfikacji oraz regresji, podczas gdy algorytm k-średnich stosowany jest w zadaniach grupowych. Oba algorytmy, dzięki wielu zaletom znajdują szerokie zastosowanie w różnorodnych dziedzinach.

ABSTRACT: Machine learning is a method of data analysis that involves automating analytical models to produce more accurate results. There are four types of algorithms - supervised, semi-supervised, unsupervised and enhanced, which include the K-Nearest Neighbors (KNN) algorithm and the k-means (K-Means) algorithm. The former is a non-parametric supervised learning classifier, while the latter is classified as unsupervised machine learning. The k-nearest neighbor algorithm is used for classification and regression, while the k-means algorithm is used for clustering tasks. Both algorithms, thanks to their many advantages, are widely used in a variety of fields.

Słowa kluczowe: algorytm, uczenie maszynowe, k-najbliższych sąsiadów, k-średnich

1. Uczenie maszynowe

Uczenie maszynowe to metoda analizy danych, której głównym zadaniem jest automatyzacja tworzenia modeli analitycznych, na podstawie których uzyskujemy dokładniejsze wyniki. W uczeniu maszynowym istotne jest zrozumienie zasady działania algorytmu i jego podstawowej klasyfikacji. Pomaga to uniknąć nieporozumień i innych błędów z tym związanych. Wyróżnia się cztery główne rodzaje algorytmów uczenia maszynowego: nadzorowane, przez wzmocnienie, nienadzorowane i wzmocnione. W celu uzyskania jak najlepszego modelu, należy wybrać odpowiedni algorytm uczenia maszynowego zgodnie ze sformułowaniem problemu [4].

2. Algorytmy

Algorytm to ciąg jasno zdefiniowanych i uporządkowanych czynności, które konieczne są do wykonania zadanej czynności. Jest to instrukcja postępowania, która na podstawie danych wejściowych (input), wytwarza dane wyjściowe (output) poprzez ich przetworzenie. Każdy algorytm posiada jasno zdefiniowany zestaw warunków, który służy do rozwiązania określonej klasy zadań. Jego opis składa się z opisu danych, które podlegają przetwarzaniu oraz opisu czynności, które należy wykonać [1].

Algorytm poprzez swoje zastosowanie do rozwiązywania problemów matematycznych oraz faktu, że właściwości algorytmów badane są za pomocą narzędzi matematycznych łączy informatykę

rozumianą jako algorytmiczne przetwarzanie zakodowanych informacji za pomocą określonego automatu (komputera) z matematyką [2].

Wyróżnia się dwa rodzaje algorytmów: leniwe (lazy algorithms) oraz gorliwe (eager learning algorithms). Algorytmy leniwego uczenia się to typy algorytmów, które przechowują dane podczas uczenia i wstępnie przetwarzają je w fazie testowania. Wymagają krótszego czasu uczenia i dłuższego czasu przewidywania. Natomiast algorytm gorliwego uczenia się przetwarza dane w fazie uczenia i jest szybszy niż algorytmy leniwego uczenia się w przewidywaniu obserwacji danych [3].

3. Algorytm KNN

Algorytm k-najbliższych sąsiadów (KNN – K-Nearest Neighbors) to nieparametryczny, nadzorowany klasyfikator uczenia się, który wykorzystuje bliskość do klasyfikacji lub przewidywania grupowania poszczególnych punktów danych. Jest to jeden z najpopularniejszych i najprostszych klasyfikatorów stosowanych zarówno do klasyfikacji jak i regresji, które używane są obecnie w uczeniu maszynowym. Należy on do grupy algorytmów leniwych (lazy algorithms), czyli takich, które nie tworzą wewnętrznej reprezentacji wiedzy, lecz poszukają rozwiązania po prezentacji każdego wzorca testowego przeszukując dane uczące.

3.1. Miara odległości

W algorytmie KNN bardzo ważne jest poprawne zdefiniowanie miary odległości pomiędzy obiektami. Służy ona do wyboru najbliższych sąsiadów. W tym celu najczęściej posługuje się odległością euklidesową, która dla dwóch przypadków A i B wyrażana jest wzorem:

$$d(A,B) = \sqrt{(A_1 - B_1)^2 + (A_1 - B_1)^2 + \cdots + (A_n - B_n)^2} \quad (1)$$

W przedstawionym równaniu, A_1 oraz B_1 oznaczają wartości i dla przypadku odpowiednio A i B.

Często zamiast odległości euklidesowej stosuje się odległość taksówkową (odległość Manhattan), która opiera się na sumie bezwzględnych różnic między wartościami [5]:

$$d(A,B) = |A_1 - B_1| + |A_2 - B_2| + \cdots + A_n - B_n \vee \quad (2)$$

Uogólnioną formą obu wymienionych metryk jest odległość Minkowskiego, wyrażona wzorem:

$$(\sum_{i=1}^{n} |x_i - y_i|)^{1/p} \quad (3)$$

Parametr p pozwala na utworzenie innych metryk odległości. Gdy jego wartość wynosi dwa - odległość ta odnosi się do odległości euklidesowej, natomiast gdy przyjmuje wartość jeden – odległości Manhattanu [10].

3.2. Klasyfikacja i regresja

Algorytm KNN używany jest zarówno do klasyfikacji jak i regresji. W pierwszym przypadku jest on wynikiem przynależności do danej klasy i przypisywany do tej występującej najczęściej wśród jego k najbliższych sąsiadów. Natomiast w przypadku regresji wynikiem jest wartości właściwości danego obiektu i jest średnią najbliższych sąsiadów [8].

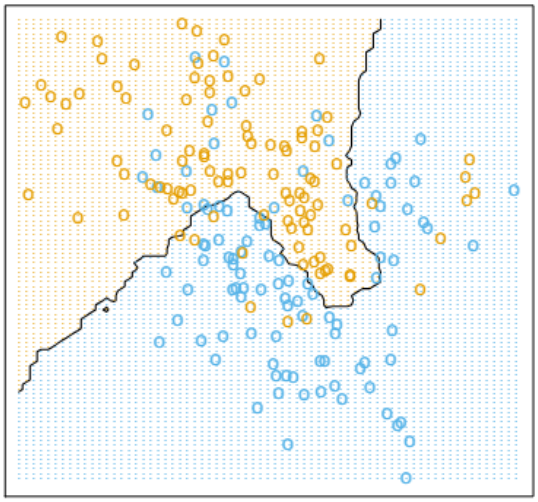
W klasyfikacji zmienna wyjściowa jest skategoryzowana i głównym zadaniem algorytmu jest przewidzenie klasy na podstawie danych dla nowego obiektu. Pierwszym krokiem jest wybór sposobu, w jakim należy zmierzyć określone sąsiedztwo, a następnie wybrać odpowiednią wartość k najbliższych sąsiadów. W kolejnym etapie należy znaleźć obiekty, które są najbardziej podobne do nowego przykładu, a więc te leżące najbliżej siebie i sprawdzić która z klasy występuje najczęściej wśród wszystkich znalezionych obiektów. Ostatnim krokiem jest przypisanie do obiektu klasy występującej najczęściej pośród wszystkich k-najbliższych sąsiadów [8].

Algorytm ten działa podobnie w przypadku regresji, czyli w sytuacji gdy zmienna wyjściowa jest ciągła i przewiduje wartość numeryczną dla nowego obiektu. Wtedy zamiast sprawdzenia najczęściej występującej klasy, oblicza się wartość średnią lub medianę dla k-najbliższych sąsiadów. Wynik ten uznawany jest za wartość, którą należy przypisać dla nowego przykładu [8].

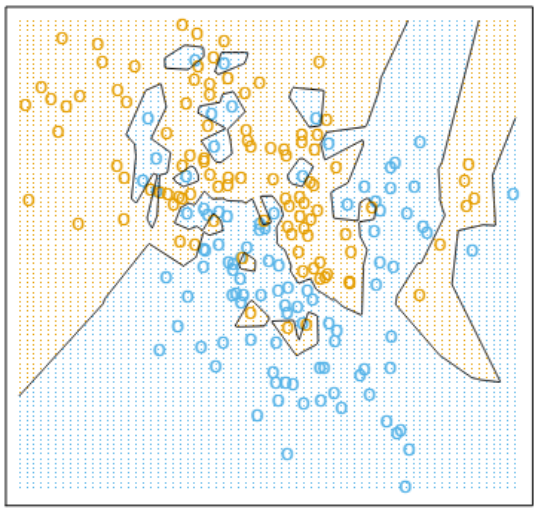
3.3. Wybór liczby K

Odpowiedni wybór liczby k ma ogromne znaczenie dla całego algorytmu. Jest to podstawowy parametr metody decydujący o jakości predykcji. Należy do hiperparametrów, przez co jego wartość musi zostać podana na początku wykonywania algorytmu [8].

Aby prawidłowo wybrać liczbę k, które będzie najbardziej odpowiednie dla danych, należy kilkukrotnie uruchomić algorytm KNN z różnymi wartościami k i ocenić, które z nich zmniejsza liczbę błędów. Jednocześnie algorytm nadal musi zachować swoją zdolność do dokładnego przewidywania. Jeśli zmniejszy się wartość k stają się one mniej stabilne, algorytm nie jest odporny na szumy, przez co jakości klasyfikacji jest niższa. Natomiast zwiększenie tej wartości przyczynia się do uzyskania dokładniejszych wyników, ponieważ rośnie czas działania algorytmu, zwiększa się zdolność obliczeniowa i maleją szumy [6].



Rysunek 1. Wartość k równa 15 [17].



Rysunek 2. Wartość k równa 1 [17].

Na **Rysunku 1** przedstawiono sytuację, w które wartość k najbliższych sąsiadów wynosi 15, natomiast na drugim (**Rysunek 2**) wartość ta wynosi 1. Większa wartość spowodowała zmniejszenie szumów i wygładzenie linii w porównaniu z mniejszą wartością k, a co za tym idzie – uzyskanie dokładniejszych wyników [7].

W celu jak najdokładniejszej i optymalnej wartości k należy zastosować metodę walidacji krzyżowej. Obejmuje on osiem etapów i zaczyna się od wstępnego wybrania wartości k i losowym podziale danych na określoną ilość rozłącznych zbiorów, z których pierwszy uznawany jest za zbiór testowy a pozostałe jako zbiory treningowe. Następnie dla przykładów znajdujących się w zbiorze testowym, należy dokonać predykcji za pomocą algorytmu KNN, porównać wyniki z wartościami rzeczywistymi i obliczyć dokładność w przypadku klasyfikacji lub błąd w przypadku regresji. Czynności te powtarza się dla kolejnych zbiorów i oblicza wartość średnią. Wszystkie kroki wykonuje się ponownie dla różnych wartości k i spośród wszystkich wybiera się tą, dla której uzyskano najlepszą średnią ocenę predykcji [8].

3.4. Zasada działania

Zasadę działania KNN obrazuje podobieństwo reprezentowane przez grupę posiadającą określoną liczbę najbardziej podobnych do siebie obiektów. Pierwszym krokiem jest opisanie podobieństwa między parami przykładów, a następnie dla nowego należy znaleźć określoną liczbę obiektów, które są najbardziej do niego podobne. Ostatnim już krokiem jest połączenie wybranych przykładów w celu uzyskania jednej wartości przypisywanej do nowego obiektu [8].

3.5. Zalety i wady algorytmu KNN

Algorytm KNN jest jednym z najpopularniejszych i najprostszych algorytmów uczenia maszynowego. Charakteryzuje się łatwością wdrożenia, ponieważ jest to jeden z pierwszym klasyfikatorów z jakim ma do czynienia analityk danych. Ponadto może być stosowany zarówno w klasyfikacji jak i regresji.

W miarę dodawania nowych obiektów, algorytm na bieżąca uzupełnia nowe dane, dzięki czemu łatwo się dostosowuje i przechowuje zebrane dane w pamięci.

Algorytm ten nie potrzebuje wielu hiperparametrów, ponieważ wymaga jedynie zastosowania określonej wartości k oraz wybrania odpowiedniej metryki odległości, co w porównaniu z innymi algorytmami uczenia maszynowego jest bardzo korzystne.

Jednak mimo wielu zalet, algorytm ten posiada także wiele niedogodności, do których zalicza się między innymi duże zapotrzebowanie na pamięć. KNN musi przechowywać informację o wszystkich danych ze zbioru uczącego.

Istotnym problem w algorytmie KNN jest konieczność podania wartości k najbliższych sąsiadów, poprzez kilkukrotne uruchomienie programu. Wiąże się to także z odpowiednim doborem wartości k, co jest kolejną wadą algorytmu. Zmniejszenie wartości k powoduję, że algorytm staje się mniej odporny na szumy, dane stają się mniej stabilne, przez co jakość klasyfikacji maleje. Natomiast wyższe wartości powodują zwiększenie zdolności obliczeniowej, zmniejszenie szumów i uzyskanie dokładniejszych wyników.

Kolejną wadą algorytmu jest czas klasyfikacji, który zwiększa się wraz z powiększeniem zbioru danych.

3.6. Zastosowanie

Algorytm poprzez swoją prostotę i duża dokładność znajduje zastosowanie w różnorodnych dziedzinach, wykorzystując do tego klasyfikację lub regresję.

Ciekawym i bardzo przydatnym zastosowaniem algorytmu KNN jest określenie czy dany odpad można poddać recyklingowi organicznemu czy materiałowemu [9]. Wykorzystywany jest także do klasyfikacji obrazów na podstawie ich cech, np. do rozpoznawania twarz znajdujących się na zdjęciach lub obrazach, a także do klasyfikacja obiektów, które przedstawione są na różnych obrazach medycznych. Algorytm ten ma także zastosowanie w przypadku rozpoznawania mowy i klasyfikacji dźwięków wykorzystując do tego celu akustykę [10].

Często spotykanym zastosowaniem algorytmu jest automatyczne rekomendowanie treści użytkownikom korzystającym ze stron internetowych. Innym ważnym obszarem w którym wykorzystuję się KNN jest medycyna. Algorytm ten potrafi przewidzieć ryzyko ataków serca czy raka prostaty [10].

4. Algorytm K-Means

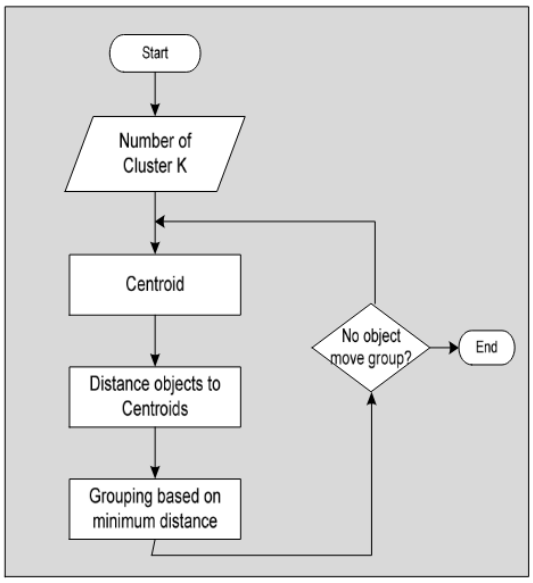
Metody grupowania danych dzieli się na metody niehierarchiczne i hierarchiczne. Do pierwszych z nich zalicza się metodę k-średnich (K-Means), która jest bardzo popularnym i powszechnie używany

algorytm grupowania danych. Opiera się na obliczaniu odległości pomiędzy każdym elementem danych, a więc wektorem cech, a każdym centroidem (klastrem) [11]. Algorytm ten zaliczany jest do uczenia maszynowego bez nadzoru i służy do grupowania podobnych punktów danych [12].

4.1. Zasada działania k-średnich

Głównym zadaniem klasteryzacji metodą k-średnich jest maksymalizacja podobieństwa w danym skupieniu i różnic pomiędzy nimi. Osiąga się to w wyniku przenoszenia danych obiektów pomiędzy skupieniami aż do uzyskania optymalizacji zmienności występujących wewnątrz i pomiędzy skupieniami. Głównym celem tego algorytmu jest minimalizacja sumy odległości pomiędzy punktami a odpowiadającym im środkiem ciężkości skupień [12].

Pierwszym krokiem jaki należy wykonać stosując ten algorytm jest ustalenie liczby skupień (liczby klas, na które podzielone są dane) oraz warunków zatrzymania, czyli liczby iteracji lub braku przesunięć obiektów pomiędzy danymi skupieniami. Kolejnym krokiem jest wybór metryki i ustalenie środków skupień (centroidów). W pierwszej iteracji dokonuje się to za pomocą losowego wyboru k obserwacji i doboru odpowiedniej odległości skupień. Natomiast w kolejnych środek skupień wyznacza się za pomocą średniej arytmetycznej współrzędnych punktów, które należą do danego skupienia. Następnie oblicza się odległość obiektów od wyznaczonych środków skupień i przypisuje się je do danego skupienia, dla którego środek występuje najbliżej. Jeżeli warunek zatrzymania jest spełniony, grupowanie jest zakończone, natomiast jeśli nie jest spełniony należy od nowa ustalić środki skupień [8]. Wszystkie etapy algorytmu przedstawione są na **Rysunku 3**.



Rysunek 3. Etapy algorytmu k-średnich [13].

Odległość obiektów od środków skupień oblicza się na podstawie odległości euklidesowej *odl(x,y)*, wyrażonej wzorek (1). Poprzez zmniejszenie błędu kwadratowego, metoda k-średnich umożliwia znalezienie optymalnego rozwiązanie, co przedstawione jest wzorem:

$$E = \sum_{i=1}^{k} \sum_{j=1}^{k} \left| odl\left(x_j, y_i\right) \right|^2 \tag{4}$$

gdzie: E jest funkcją celu, k – numerem klastru, n – liczbą punktów, x,y – określonymi przypadkami.

Błąd ten obliczany jest na podstawie odległości podniesionej do kwadratu między każdym punktem a środkiem danego klastra [13].

4.2. Zalety i wady algorytmu k-średnich

Najważniejszą zaletą algorytmu jest jego proste i łatwe w zrozumieniu działanie. Algorytm k-średnich poprzez mniejszą złożoność obliczeniową działa znacznie szybciej niż pozostałe algorytmy, bez względu na wielkość danego zbioru [12].

Algorytm ten wrażliwy jest na dobór odpowiednich k obserwacji, ponieważ losowo dobiera punkty startowe. Dlatego więc dokładność uzyskanych wyników zależy także od czynnika losowego.

Jednak główną wadą algorytmu k-średnich jest ukierunkowanie jedynie na sferyczne kształty skupień, przez co staje się on niewrażliwy na pozostałe kształty [12].

Literatura

[1] Michał Bąba, Algorytmy — nowy wymiar nadzoru i kontroli nad świadczącym pracę, t. LXI, nr 3/2020
[2] Paweł Stacewicz, O algorytmach i algorytmicznej dostępności wiedzy, Studia metodologiczne, nr 36, 2016
[3] https://www.analyticsvidhya.com/blog/2023/02/lazy-learning-vs-eager-learning-algorithms-in-machine-learning/
[4] https://www.sas.com/pl_pl/insights/analytics/machine-learning.html
[5] Przemysław Juszczuk, Sztuczna inteligencja: Algorytm KNN, Instytut Informatyki Uniwersytetu Śląskiego 23 kwietnia 2012
[6] Paweł Drzewiecki, Predykcja poziomu tlenu w piecu EAF z wykorzystaniem metod sztucznej inteligencji: regresji liniowej i metody najbliższych sąsiadów (k-NN), Procesy termiczna, piece przemysłowe & kotły I-II/2013
[7] Trevor Hastie,Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, Data Mining, Inference, and Prediction, Springer Series in Statistics
[8] Aleksandra Król-Nowak, Katarzyna Kotarba, Podstawy uczenia maszynowego pod redakcja Aleksandry Król-Nowak, Wydawnictwo AGH, Kraków, 2022
[9] https://ekordo.pl/klasyfikacja-odpadow-na-ulegajace-organicznemu-i-materialowemu-recyklingowi-za-pomoca-metody-k-najblizszych-sasiadow-ang-k-nearest-neighbors/
[10] https://www.ibm.com/topics/knn
[11] Tomasz Kuczyński, Algorytm k-średnich na wielordzeniowych procesorach CPU oraz akceleratorach GPU, Rozdział I
[12] https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/
[13] Alhamza Munther, Rozmie Razif, A Preliminary Performance Evaluation of K-means, KNN and EM Unsupervised Machine Learning Methods for Network Flow Classification, International Journal of Electrical and Computer Engineering (IJECE), Vol. 6, No. 2, April 2016, pp. 778~784