AKADEMIA INNOWACYJNYCH ZASTOSOWAŃ TECHNOLOGII CYFROWYCH (AI TECH)

„Uczenie maszynowe" – laboratorium

Laboratorium 4

Algorytmy grupowania danych

data aktualizacji: 14.05.2025

Cel ćwiczenia

Celem ćwiczenia jest zapoznanie się z wybranymi algorytmami uczenia nienadzorowanego w ramach zadania grupowania (także: klasteryzacji, ang. *clustering*). Treść zadania obejmuje pracę z dwoma algorytmami grupowania: k-means (k-średnich) oraz DBSCAN (*density-based spatial clustering of applications with noise*). W ramach realizacji zdania zbadane zostaną różne konfiguracje algorytmów, użyte zostaną różne miary oceny jakości grupowania oraz metody wizualizacji danych.

Wprowadzenie

Uczenie nadzorowane (lub uczenie z nauczycielem) związane jest z tworzeniem modelu na podstawie danych, który dokładnie wie jakie jest wejście i wyjście modelu (np. w klasyfikacji mamy oznaczenie klasy dla każdego rekordu), a algorytm w procesie uczenia budując model wykorzystuje tę wiedzę. W uczeniu nienadzorowanym brakuje takiej informacji (brak nauczyciela…) i właśnie z taką sytuacją mamy do czynienia w zadaniu grupowania. Mamy dane (zbiór rekordów), dla których brak jest oznaczeń (klas), a zadanie sprowadza się do ich pogrupowania.

Przy zadaniu grupowania pojawiają się pytania. Jak pogrupować? Co znaczy dobre pogrupowanie danych? Czy powinno być dużo klastrów? Czy klastry powinny skupiać tylko podobne dane? Jak bardzo klastry powinny być „daleko" w przestrzeni danych? Na te pytania odpowiedzi są różne, w zależności od analizowanego zbioru danych oraz… przyjętej miary jakości klasteryzacji.

Co jeśli chcemy pogrupować dane, które mamy oznaczone (klasą)? Wtedy w zadania grupowania dodatkowo należy przeanalizować jak klasy rozłożyły się w klastrach. Jak klastry są jednorodne (miara *Purity*)? Ile (%) klas znajduje się w danym klastrze?

Algorytmy

Algorytm k-średnich: przy narzuconej liczbie klastrów *k* próbuje rozdzielić zbiór danych na *k* grup opisanych za pomocą *centroidów*, tj. punktów w przestrzeni stanowiących centra klastrów; algorytm minimalizuje *inercję*, tj. sumę kwadratów odległości punktów składających się na klaster od środka klastra.

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

DBSCAN: identyfikuje w zbiorze danych rejony o wysokiej gęstości, wyznaczając punkty stanowiące *rdzeń* klastra (ang. *core*) i budując klastry w sąsiedztwie tych punktów. Uwaga: algorytm DBSCAN nie ma trywialnej metody predykcji przynależności klastra dla nowych punktów (w odróżnieniu od k-means)!

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

Użyte zbiory danych

IRIS – tylko jako „sanity check"
SEEDS – zbiór „wdzięczny" do grupowania https://archive.ics.uci.edu/ml/datasets/seeds
GLASS – zbiór trudniejszy https://archive.ics.uci.edu/ml/datasets/glass+identification
Polish Companies Bankruptcy – zbiór bardzo trudny

Sugerowane narzędzia

Moduł `clustering` pakietu `sklearn` zawiera implementacje analizowanych algorytmów grupowania oraz miar oceny jakości. Sugerowana jest lektura dokumentacji i przewodnika dostępnego pod linkiem: https://scikit-learn.org/stable/modules/clustering.html

Miary grupowania – gdy nie znamy etykiet *ground truth*:

- Silhouette (w dokumentacji sklearn, sekcja 2.3.10)
- Davies-Bouldin Index (tamże)

Uwaga! Część miar jest minimalizowana, pozostałe są maksymalizowane. Warto zwrócić uwagę, jak poszczególne miary zachowują się w granicznych warunkach. Na przykład, co się dzieje gdy mamy tylko jeden klaster, lub co się dzieje jak mamy tyle samo klastrów co rekordów?

Miary grupowania – gdy mamy dostęp do etykiet:

- Rand Index (w dokumentacji sklearn)
- Purity – niedostępna *bezpośrednio* w sklearn (ale można ją wyliczyć na podstawie Contingency Matrix), patrz wzór z: https://www.rdocumentation.org/packages/funtimes/versions/7.0/topics/purity

Przebieg ćwiczenia

1. Sanity check:
   - wczytanie danych,
   - wstępne uruchomienie k-means na IRIS,
   - wizualizacja,
   - obliczenie metryk jakości.
2. Przebadanie algorytmu k-means na zbiorach SEEDS i GLASS. Zbadanie jak zmieniają się wartości miar przy zmianie parametrów:
   - n_clusters – liczba klastrów (sprawdź najpierw na IRIS)
   - n_init – liczba restartów
   - max_iter – maksymalna liczba iteracji
3. Wizualizacja wyników klasteryzacji algorytmem k-means (dla wybranych konfiguracji hiperparametrów) oraz uzyskanych wartości miar jakości.

4. Przebadanie algorytmu DBSCAN na zbiorach SEEDS i GLASS. Zbadanie jak zmieniają się wartości miar przy zmianie parametrów:
    - eps – rozmiar sąsiedztwa punktu (sprawdź najpierw na IRIS)
    - min_samples – liczba punktów w sąsiedztwie potrzebna by uznać punkt za „core”
    - metric – rodzaj metryki odległości
5. Wizualizacja wyników klasteryzacji algorytmem DBSCAN (dla wybranych konfiguracji hiperparametrów) oraz uzyskanych wartości miar jakości.
6. Analiza wyników:
    - porównanie działania algorytmów k-means i DBSCAN,
    - porównanie użytych miar jakości grupowania.
7. [dla chętnych i nie bez zastanowienia] Przetestowanie wybranego przez siebie algorytmu na zbiorze PCB, wizualizacja i obliczenie miar.

Punktacja

Przy realizacji zadania możesz otrzymać **max 10 punktów** wedle poniższej tabeli.

1	Wczytanie danych, „sanity check”, wizualizacja
3	K-means – zbadanie algorytmu w funkcji 3 parametrów dla zbiorów SEEDS i GLASS
3	DBSCAN – zbadanie algorytmu w funkcji 3 parametrów dla zbiorów SEEDS i GLASS
1	Analiza jakości działania obu algorytmów w kontekście miar opartych o etykiety ground truth i tych nie uwzględniających etykiet (porównanie)
2	Analiza porównawcza obu algorytmów w oparciu o wizualizację klastrów

W zadaniu można narysować milion wykresów i milion tabelek. Założenie sprawozdania jest ukazanie tylko pewnego aspektu przeprowadzonych badań (nie wszystkie!). W tym celu podawane są powyższe punkty realizacji zadania oraz pytania pomocnicze.

Przy realizacji tego zadania wystarczy notebook Jupytera, z wykresami i tabelkami.

Pytania pomocnicze

1. Czy przy grupowaniu potrzebna jest normalizacja/standaryzacja danych?
2. Co różni oba algorytmy z punktu widzenia reprezentacji klastra?

   Jaką konsekwencję mają te różnice na możliwości wykonywania „predykcji”?
3. Który z algorytmów jest mniej odporny na szum i wartości odstające (ang. *outliers*)? Dlaczego?
4. Czy w zadaniu grupowania powinniśmy użyć walidacji krzyżowej?
5. Czy wyniki badanych algorytmów klasteryzacji powinny być powtarzane i uśredniane?
6. Co mierzą miary klasteryzacji podane w treści zdania?
7. Czy konieczne jest ustawianie liczby klastrów tak, by odpowiadała liczbie klas?

Literatura

1. Wykłady do przedmiotu autorstwa prof. H. Kwaśnickiej
2. Przewodnik użytkownika pakietu sklearn.clustering: https://scikit-learn.org/stable/modules/clustering.html
3. Cichosz P. ”Systemy uczące się”, WNT Warszawa
4. Zasoby Internetu: uczenie maszynowe (machine learning), data mining, grupowanie, klasteryzacja, clustering, k-means, dbscan