Metody Sztucznej Inteligencji - Projekt
Analiza porównawcza algorytmów segmentacji

Mateusz Blicharski
235266
Wojciech Bodzek
235701

I. Wstęp

Segmentacja danych jest obecnie jednym z ważniejszych elementów statystycznej analizy danych. Z roku na rok obserwujemy coraz większą ilość baz danych, a wraz wzrostem ich objętości analiza skupień zwiększa swoje znaczenie. Zbiory danych zawierać mogą nawet miliony obiektów opisanych dziesiątkami cech, dlatego też metody analizy danych mają ciężkie zadanie w operacji na nich. Przykładami może być na pewno niewyobrażalna liczba zapytań w wyszukiwarkach internetowych czy choćby opinie wyrażane na różnych serwisach internetowych. Grupowanie jednostek jest zadaniem złożonym. Na uzyskane rozwiązanie wpływa wiele czynników. Do ważniejszych można zaliczyć: liczbę grupowanych jednostek, liczbę cech zmiennych opisujących daną jednostkę czy nawet zastosowane skale pomiarowe wszystkich cech. Każdy z tych czynników powoduje konieczność innego podejścia do problemu. Wynikiem tego jest potrzeba istnienia dużej liczby różnego rodzaju algorytmów segmentacji. W projekcie podjęto próbę porównania niektórych z nich.

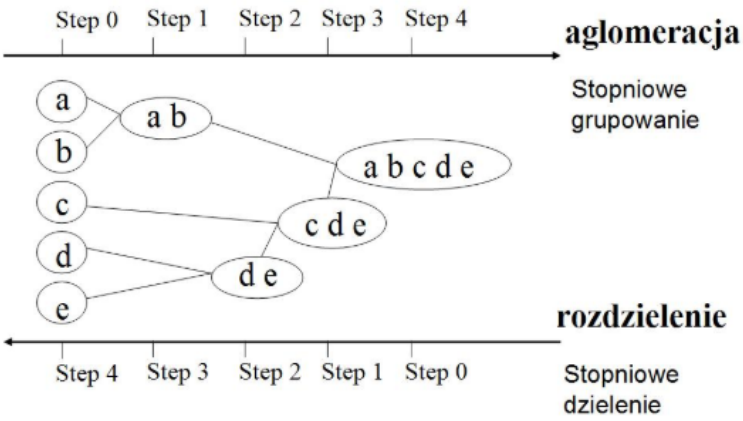
II. Wybór algorytmów oraz kryterium oceny

Segmentacja (analiza skupień, klasteryzacja, grupowanie) (ang. Clustering) jest pojęciem z zakresu eksploracji danych oraz uczenia maszynowego. Analiza skupień jest metodą tzw. Klasyfikacji bez nadzoru, zakłada brak obecności dokładnego, lub nawet przybliżonego wyjścia w danych uczących. Celem segmentacji jest ułożenie obiektów w grupy w taki sposób, by obiekty należące do tej samej grupy były ze sobą jak najbardziej powiązane, a jednocześnie były jak najmniej związane z obiektami z pozostałych grup. Wynikiem działania algorytmów grupujących są grupy obserwacji. Nie wszystkie obserwacje zostają jednak przydzielone do grup z powodu faktu, że na płaszczyźnie wielowymiarowej są one odległe od wszystkich pozostałych obserwacji. Obserwacje te, nazywamy obserwacjami nietypowymi. Grupowanie jest ściśle uwarunkowane źródłem danych oraz oczekiwaną postacią rezultatów. Podział algorytmów grupujących można dokonać na podstawie przyjętych przez nie metody grupowania:

1. Grupowanie hierarchiczne – Ma na celu zbudowanie hierarchii klastrów. Służy do dzielenia obserwacji na grupy (klastry) bazując na podobieństwie między nimi. Możemy je podzielić ze względu na podejście do budowy grup:

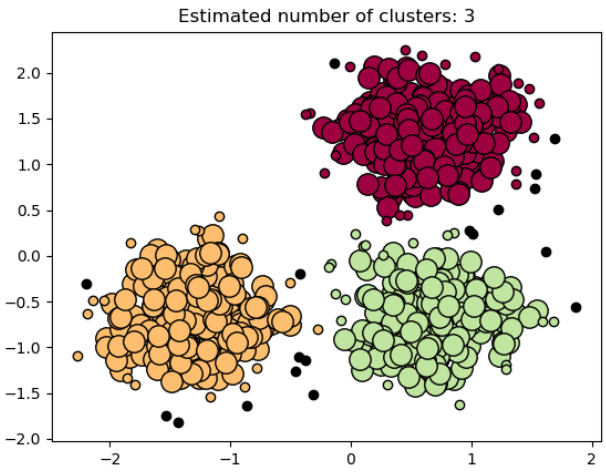
Aglomeracyjne – każda obserwacja tworzy na początku jednoelementowy klaster. Pary klastrów są scalane, a następnie w każdej iteracji algorytmu łączone ze sobą są dwa najbardziej zbliżone klastry. Tworzone są tzw. „aglomeracje". Podczas tworzenia klastrów poruszamy się w górę hierarchii.

Deglomeracyjne – zaczynają od skupienia obejmującego wszystkie obiekty, a następnie dzielą je na mniejsze i bardziej jednorodne skupienia. Podziały wykonywane są rekursywnie. W czasie tworzenia poruszamy się w dół hierarchii.



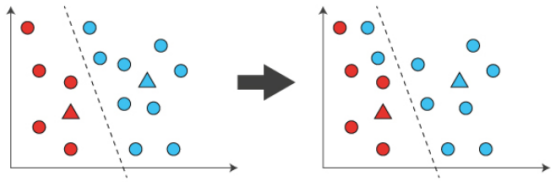
Rysunek 1. Grupowanie hierarchiczne

2. Grupowanie gęstościowe- metody oparte o zdefiniowany, maksymalny zbiór gęstościowo połączonych punktów. Klastry reprezentowane są jako obszary o większej gęstości niż pozostała przestrzeń obiektów. W tej metodzie grupowania, wykorzystywana jest koncepcja gęstościowej osiągalności, gdzie każdy punkt należący do klastra musi posiadać zadaną minimalną liczbę sąsiadów w swoim otoczeniu.



Rysunek 2. Grupowanie gęstościowe

3. Grupowanie iteracyjno-optymalizacyjne – algorytmy wstępnie wykonują podział na zadaną liczbę klas, a następnie w sposób iteracyjny poszczególne obserwacje migrują pomiędzy klasami celem zapewnienia możliwie najmniejszej wariancji.



Rysunek 3. Grupowanie iteracyjne

Celem naszego projektu jest porównanie algorytmów segmentacji reprezentujących wyżej opisane grupy. Do badania wykorzystaliśmy 4 algorytmy.

A) K-means
Jest metodą należącą do grupy algorytmów iteracyjno-optymalizacyjnych. Reprezentuje ona grupę algorytmów niehierarchicznych. Przy pomocy algorytmu tworzone jest k różnie możliwie odmiennych skupień. Wymaga określenia apriori oczekiwanej liczby podziału na grupy. Algorytm polega na przenoszeniu obiektów ze skupienia do skupienia tak długo aż zostaną zoptymalizowane zmienności wewnątrz skupień oraz pomiędzy skupieniami. Podobieństwo w skupieniu powinno być jak największe, zaś osobne skupienia powinny się od siebie jak najbardziej różnić.

Algorithm 1: Algorytm K-means
wejście: **P** - zbiór przykładów P,
$K \in \Re$ – liczba skupień.
wyjście: $C = \{C_1, ...C_K\}$ - zbiór skupień.
begin
  while *są zmiany w skupieniach* C_k **do**
    Podziel zbiór danych na dowolnych k rozłącznych zbiorów;
    Dla każdej grupy wyznacz środek ciężkości;
    Podziel przykłady do najbliższego środka;
end

Rysunek 4. Algorytm k-means

Zasada działania:
1. Określamy liczbę skupień.
2. Ustalamy wstępne środki skupień (np. wybór k pierwszych obserwacji)
3. Obliczamy odległości obiektów od środków skupień
4. Przypisujemy obiekty do skupień
5. Ustalamy nowe środki skupień
6. Powtarzamy kroki 3,4,5 do momentu aż warunek zatrzymania zostanie spełniony.
Niestety algorytm k-średnich ma wiele wad. Już na wstępie konieczne jest zdefiniowanie liczby grup a zwykle nie wiadomo, jak wiele grup występuje w przetwarzanym zbiorze. Metoda jest zbieżna do lokalnego optimum, a jednokrotne wykonanie algorytmu zazwyczaj nie daje w wyniku optymalnego podziału analizowanego zbioru. Pomimo swoich wad jest to jedna z najczęściej wykorzystywanych metod iteracyjnych, głównie z powodu prostoty.

B) Agglomeration
Jest reprezentantem metod aglomeracyjnych należących do grupy algorytmów hierarchicznych. Opiera się na miarach odległości między skupieniami. Przy zadanym grupowaniu wstępnym, w celu zredukowania liczby skupień łączą dwa skupienia, które są najbliżej siebie. Za każdym razem łączone są dwa najbliższe skupienia aż do momentu, gdy istnieje tylko jedno skupienie zawierające wszystkie elementy danych.

Algorithm 2: Aglomeracyjny algorytm grupowania danych
wejście: **P** - zbiór przykładów P,
$D(C_i, C_j)$ – funkcja do mierzenia odległości między dwoma skupieniami C_i i C_j.
wyjście: Dendogram skupień.
begin
  while *pozostało więcej niż jedno skupienie* **do**
    Niech C_i i C_j będą skupieniami minimalizującymi odległość $D(C_i, C_j)$ między dwoma skupieniami;
    $C_i = C_i \bigcup C_j$;
    usuń skupienie C_j;
end

Rysunek 5. Algorytm k-means

C) Spectral

Metoda spectra wykonuje niskopoziomowe osadzanie macierzy powinowactwa między próbkami, po którym następuje K- means w przestrzeni o niskim wymiarze. Algorytm tak jak w przypadku algorytmu K-średnich wymaga podania liczby klastrów. Metody spektralne są stosowane w nieliniowej redukcji wymiarowości, tzw. Manifold learning: odkrywaniu niskowymiarowej rozmaitości, na której leżą wysokowymiarowe dane.

Problemem grupowania spektralnego jest podzielenie zbioru obserwacji X na k rozłącznych grup. Zasada działania:

1. Utwórz symetryczną macierz wag:

$$W : X \times X \longrightarrow \mathbb{R}$$

2. W oparciu o W utwórz graf podobieństwa G(V,E), gdzie . Niech S będzie uogólnioną macierzą sąsiedztw.

$$V \equiv X$$

3. Oblicz laplasjan L, i wyznacz jego rozkład spektralny

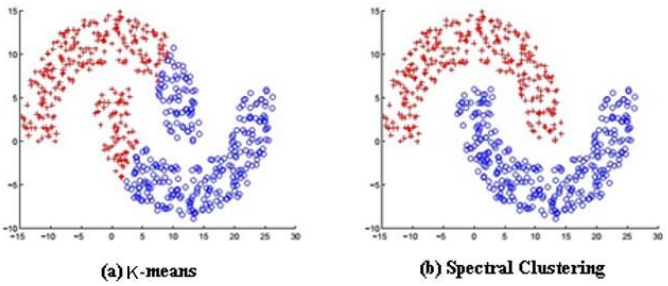
$$\mathbb{L} = Y \wedge Y^T$$

4. Utwórz odwzorowanie spektralne

$$V \ni v_i \mapsto y_i^* = (y_{i1}, \ldots, y_{ik})$$

5. Podziel zbiór , stosując dowolny algorytm grupowania.

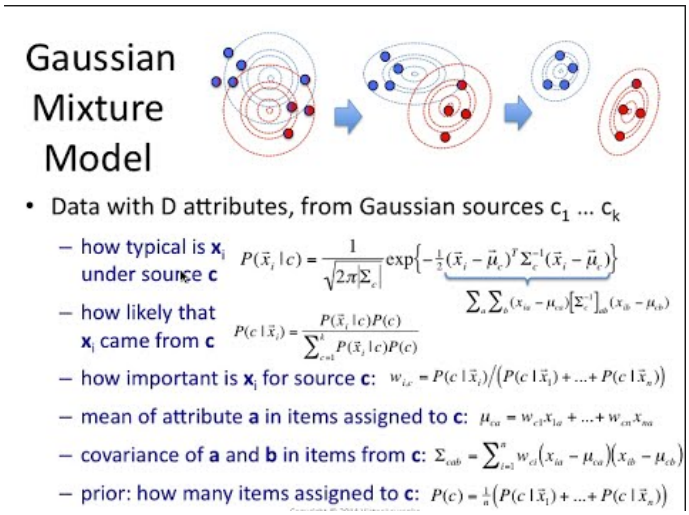
$$Y^* = y_1^*, \ldots, y_i^*$$



Rysunek 6. Spectral a K-means

D) Gaussian Mixture

Gaussian Mixture jest konkretyzacją, znanego w statystyce matematycznej algorytmu EM (Expectation-Maximalization). Algorytm GMM wymaga podania a priori liczby klastrów. Liczba klastrów określa liczbę komponentów. Klastrowanie GMM może pomieścić klastry, które mają różne rozmiary i struktury korelacji w nich. Z tego właśnie powodu, GMM może być bardziej odpowiednie do zastosowania niż np. K-means. Algorytm ten potrafi być bardzo wrażliwy na inizjalizację parametrów, dlatego warto uruchamiać go kilka razy dla tych samych parametrów, co niestety jest kosztowne obliczeniowo.



Rysunek 7. Algorytm GMM

Istnieje wiele metod określania podobieństwa pomiędzy grupowaniami. Wszystkie te metody mają na celu rozwiązanie różnych problemów związanych z klasteryzacją, np. usprawnić znalezienie najbardziej reprezentatywnego grupowania.

Miary podobieństwa pomiędzy grupowaniami można podzielić na:

- Miary oparte na zliczaniu par
- Miary oparte na dopasowaniu
- Miary oparte na wzajemnej informacji

Na potrzeby naszego projektu skupimy się na dwóch miarach, które pozwolą nam na dokonanie porównania algorytmów. Pierwszym kryterium jest reprezentant miar opartych na zliczaniu par – Skorygowany indeks Randa. Wybraliśmy go ponieważ miary oparte na zliczaniu par są według nas najbardziej intuicyjnym podejściem mierzenia podobieństwa pomiędzy grupowaniami.

Skorygowany indeks randa

Jest miarą opartą na zliczaniu par. Jego celem jest obliczenie stosunku poprawnie zaklasyfikowanych obiektów lub odpowiednio błędnie zaklasyfikowanych do wszystkich elementów zbioru pomiędzy danymi dwoma grupowaniami. Porównanie odbywa się na

podstawie par i ich przynależności do grupy.

$$Rand_{\pi_1\pi_2} = \frac{a+d}{a+b+c+d}$$

Indeks zwraca wartość z przedziału $< 0,1 >$. Gdy porównywane grupowania są identyczne zwracana jest wartość 1. Wartość 0 zwracana jest, kiedy porównywane grupowania są zupełnie niepodobne. Niestety zwykły indeks Randa ma poważną wadę. Nie zwraca zawsze tej samej, stałej wartości dla losowych grupowań tego samego zbioru danych. Z tego powodu powstała skorygowana wersja indeksu Randa pozbawiona wspomnianej wcześniej wady.

$$AdjustedRand_{\pi_1\pi_2} = \frac{Rand_{\pi_1\pi_2} - E(Rand_{\pi_1\pi_2})}{Rand_{max} - E(Rand_{\pi_1\pi_2})}$$

gdzie:
Rand - wartość obliczona za pomocą indeksu Randa,
$Rand_{max}$ - maksymalna wartość indeksu Randa(Max = 1)
E(Rand) - wartość oczekiwana indeksu Randa

Współczynnik tak jak wersja podstawowa zwraca wartość z przedziału $< 0,1 >$. W sytuacji w której porównywane grupowania będą niezależne, zwraca wartość 0. W sytuacji w której porównywane grupowania są identyczne, zwracana jest wartość 1.

Drugim kryterium które użyjemy w celach porównawczych będzie Indeks Silhouette. Nie bazuje on na testach statystycznych. Ułatwia graficzną prezentację stopnia przynależności poszczególnych obiektów do grup.

Polega na wyliczaniu średnich odległości wewnątrz grup i ich najbliższych sąsiadów. Sprowadza się do wyznaczenia współczynnika silhouette.

$$s = \frac{b-a}{max(a,b)}$$

Gdzie,
a – średnia odległość danej obserwacji od wszystkich innych obserwacji w grupie
b – średnia odległość danej obserwacji od wszystkich innych obserwacji w innej najbliższej grupie – większa wartość z wartości a i b.

Miara algorytmu silouette waha się od -1 do +1, gdzie wyższa wartość wskazuje że obiekt jest dobrze dopasowany do własnej grupy i słabo dopasowany do sąsiednich grup. Wartość współczynnika jest z zakresu $< -1,1 >$, gdzie wartość 1 oznacza, że dany obiekt jest przydzielony do najlepszej grupy, wartość -1 oznacza, że obiekt został źle sklasyfikowany. W przypadku gdy współczynnik wynosi 0, oznacza to że dany obiekt leży między dwoma grupami.

Globalny wskaźnik Silhouette jest równy wartości średniej Silhouette dla wszystkich obiektów, który otrzymujemy poprzez wyznaczanie kolejnych wartości Shilouette dla obiektów wejściowych zbioru danych.

$$GS = \frac{1}{N}\sum_{i=1}^{N} Silhouette_i$$

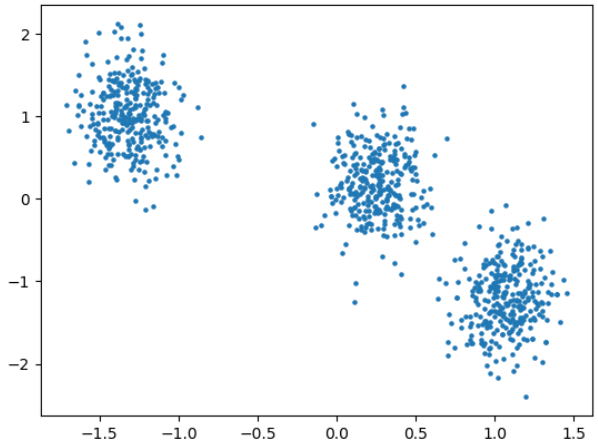
N - Liczba obiektów w zbiorze danych

III. Zbiory danych

Dane, które posłużyły nam do przeprowadzenia symulacji zostały stworzone przy pomocy wbudowanej funkcji Scikit-learn. Skorzystaliśmy z następujących generatorów:

- (make blobs)
- (make circles)
- (make moons)

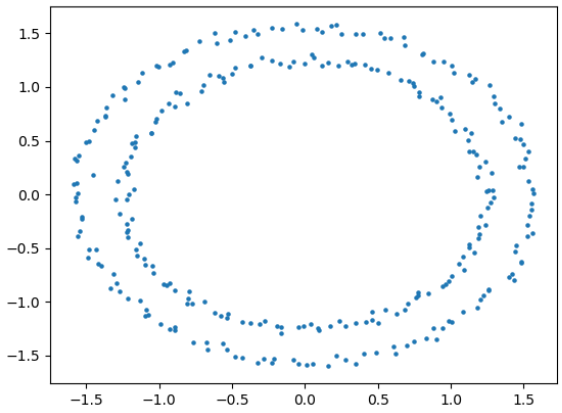
Funkcja make blobs generuje izotropowe Gaussowskie plamy. Do otrzymania danych skorzystano z parametrów:
- (sample)- docelowa ilość próbek
- (center)- ilość środków do wygenerowania, jeśli nie podane tworzone są 3
- (random state) - parametr umożliwiający otrzymywanie dokładnie takiego samego zbioru danych. Gdy nie podany wykorzystywana jest funkcja numpy.random dzięki czemu każde uruchomienie zapewnia inny zbiór danych



Rysunek 8. Dane z funkcji (make blobs)

Funkcja make circles generuje 2 okręgi, jeden znajduje się wewnątrz drugiego.
Użyto następujące parametry:

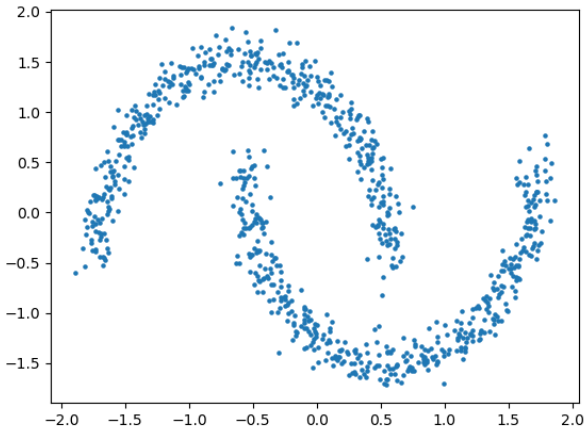
•(n samples)
•(noise)- odchylenie standardowe dodawane do każdej próbki, powoduje ono zaburzenie w ułożeniu danych
•(random state)



Rysunek 9. Dane z funkcji (make circles)

Funkcja make moons generuje 2 nachodzące na siebie półkola.
Użyto następujące parametry:

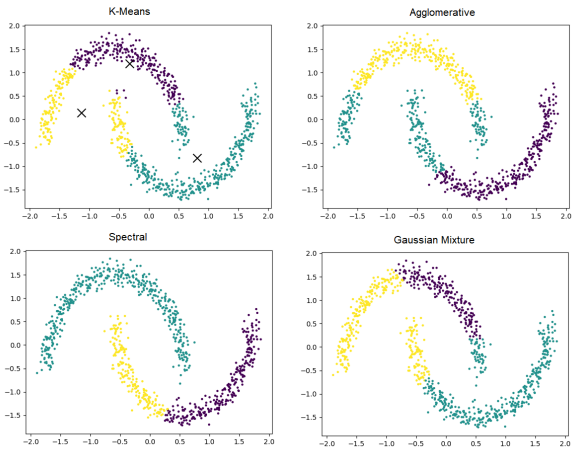
- (n samples)
- (noise)
- (random state)



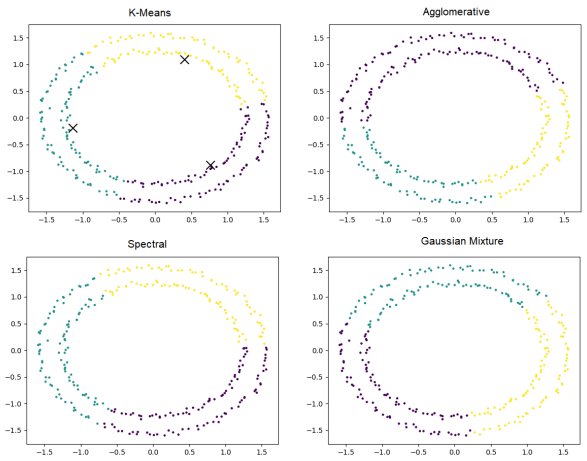
Rysunek 10. Dane z funkcji (make moons)

Przy ich pomocy wygenerowano 9 zbiorów danych, po 3 z każdej funkcji. Parametry wejściowe funkcji różniły się między sobą w obrębie funkcji ilością próbek, odchyleniem standardowym, zmienną determinującą generowane zbiory. Zapewniło to różnorodność zbiorów z jednoczesną możliwością porównania danych z tej samej funkcji różniących się parametrami. Podczas generowania danych kierowaliśmy się wrażeniami empirycznymi aby uzyskać jak najbardziej różnorodne zbiory.

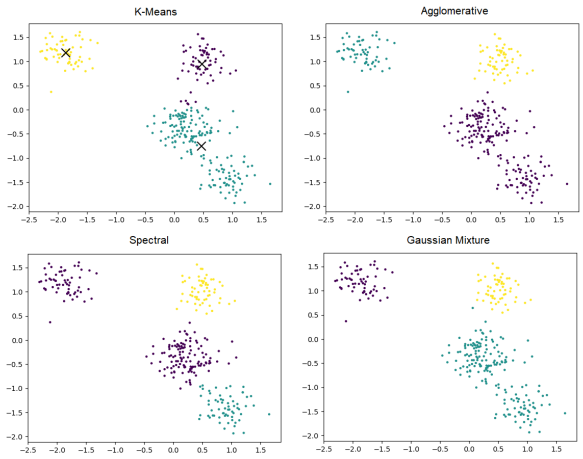
IV. Analiza wyników



Rysunek 11. Wizualizacja wyników uzyskanych z wygerenowanego zbioru funkcj make moons



Rysunek 12. Wizualizacja wyników uzyskanych z wygerenowanego zbioru funkcj make circles

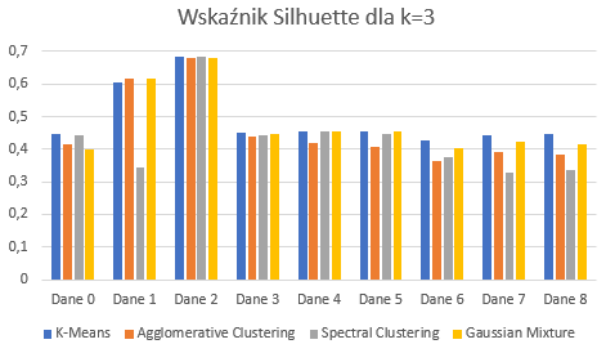


Rysunek 13. Wizualizacja wyników uzyskanych z wygerenowanego zbioru funkcj make blobs

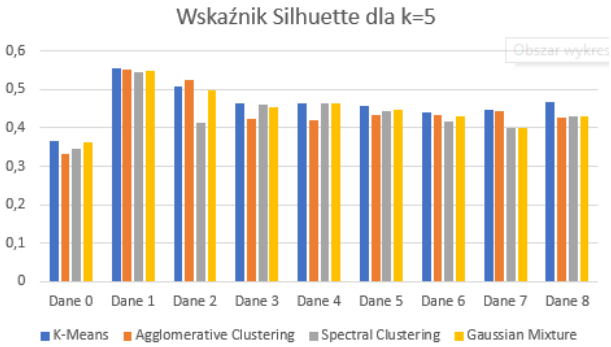
Powyżej przedstawione zostały wizualizacje otrzymanych przez nas wyników dla czterech algorytmów. Badania zostały przeprowadzone dla liczby klastrów równych 3 i 5 .Powyższej ilustracje przedstawione zostały dla k = 3. Wygenerowane wykresy są do siebie bardzo zbliżone, dlatego w celu dokładniejszej analizy zostały zastosowane dwie miary porównawcze. Metoda Silhouette oraz skorygowany indeks Randa. Poniżej przedstawiony został wykres kolumnowy zgrupowany dla wszystkich badanych 9 zbiorów danych.

Dla przedstawionych w projekcie wyników, interesujące nas kolumny to odpowiednio

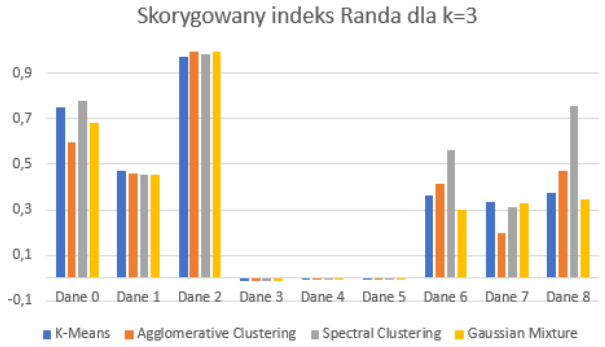
- Dane 8 - make moons
- Dane 4 - make circles
- Dane 1 - make blobs



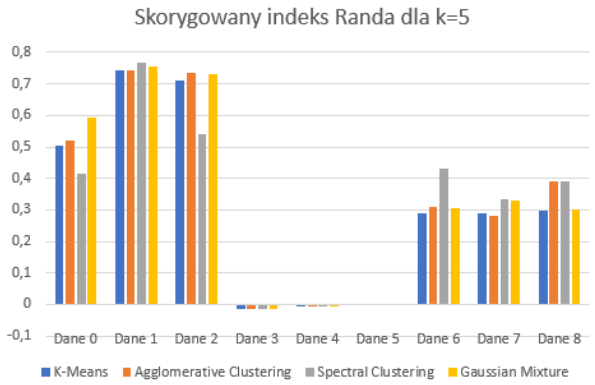
Rysunek 14. Wykres Silhouette k=3



Rysunek 15. Wykres Silhouette k=5



Rysunek 16. Wykres Skorygowanego Indeksu Randa k=3



Rysunek 17. Wykres Skorygowanego Indeksu Randa k=5

Na podstawie naszych wyników, najbardziej uniwersalnym algorytmem jest K-Means oraz Gaussian Mixture. Główne różnice w uzyskanych wynikach stanowiło ułożenie wygenerowanych danych. Zależnie od rodzaju skupień, każdy algorytm przyjmował inne założenia co do sposobu segmentacji, stąd widoczne różnicy dla kilku danych. Algorytmy Spectral Clustering oraz Agglomerative Clustering uzyskiwały widocznie niższe oceny według wybranych skal, jednak mogą one przewyższyć swoimi wynikami inne zbiory danych. Pracując na rzeczywistych danych i posiadając większą wiedzę i umiejętności w zakresie analizy skupień, możemy uzyskać wynik które mogą zdecydowanie odbiegać od tych które udało nam się uzyskać w projekcie. Wyniki badań symulacyjnych zwykle nie posiadają waloru dowodu takiego jak rozważania czysto formalne. Jednak dzięki nim możemy uzyskać cenne informację na temat badanych zagadnień w zakresie przewidzianym w projekcie. Warto również wspomnieć, że prezentowane przez nas algorytmy były badane na podstawie zaledwie dziewięciu zbiorów danych.

Literatura

[1] H. Kopka and P. W. Daly, A Guide to LaTeX, 3rd ed. Harlow, England: Addison-Wesley, 1999.

[2] Dr inż. P. Ksieniewicz Metody Sztucznej inteligencji Wykład1, Wykład 2, Wykład 3, Wykład 4

[3] Uniwersytet Warszawski, Wydział Matematyki, Informatyki i Mechaniki, Clustering- Efektywne metody grupowania danych

[4] Osama Mahmoud Abu Abbas, Comparisons Between Data Clustering Algorithms, IAJIT, Vol. 5, No. 3, 2008, p.p: 320-325. 1

[5] Polsko-Japońska Akademia Technik Komputerowych, Wykład 13 - Analiza skupień

[6] Krzysztof Rządca, Algorytmy grupowania danych

[7] Daniel Bieńkowski, Grupowanie konsensusowe danych otrzymanych z mikromacierzy

[8] J. Koronacki, J. Ćwik, Statystyczne systemy uczące się, wydanie drugie, Exit, Warsaw, 2008

[9] https://acadgild.com/blog/k-means-clustering-algorithm

[10] https://scikit-learn.org/stable/modules/clustering.html

[11] https://mateuszgrzyb.pl/wstep-do-problemu-grupowania/

[12] https://mateuszgrzyb.pl/k-srednich-praktyka/

[13] http://wazniak.mimuw.edu.pl/images/d/dc/ED-4.2-m10-1.01.pdf

[14] https://www.jakbadacdane.pl/silhouette-coefficient-czy-dobrze-pogrupowalem-obserwacje/

[15] https://nauka.metodolog.pl/metody-analizy-skupien-segmentacjagrupowanie/