

SPRAWOZDANIE 7

Podstawy Sztucznej Inteligencji

DBSCAN, K-means a algorytm hierarchiczny

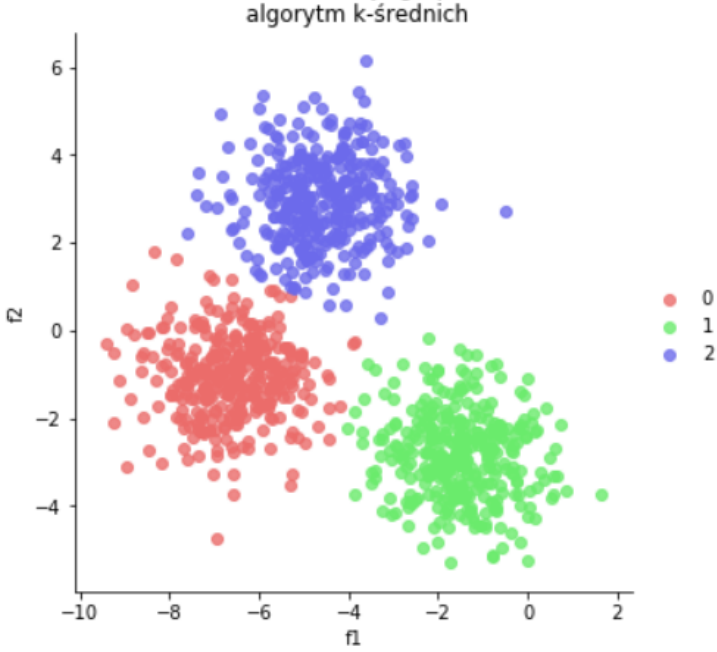
Program WEKA 7

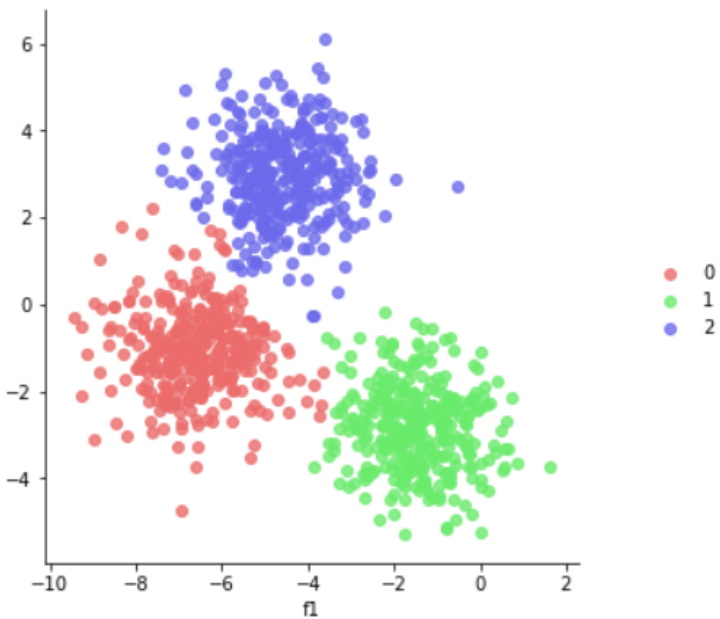
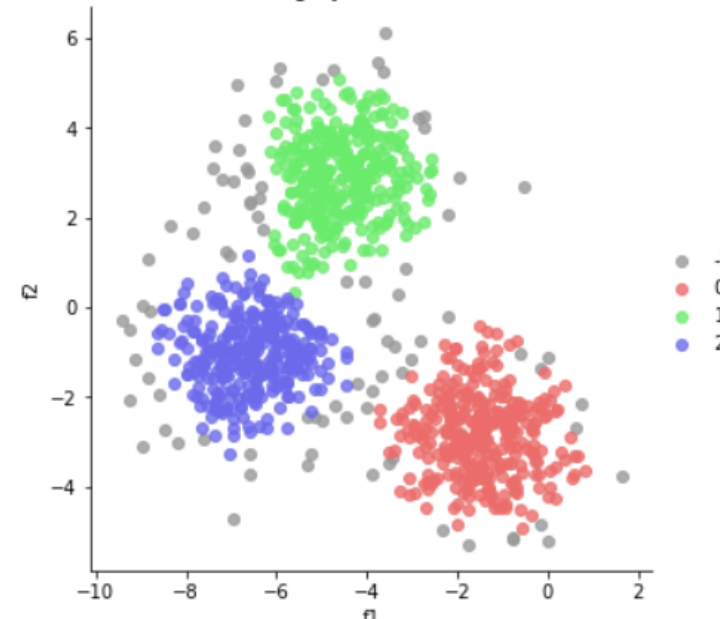
Natalia Gadocha 304165
Geoinformatyka III rok

Wstęp

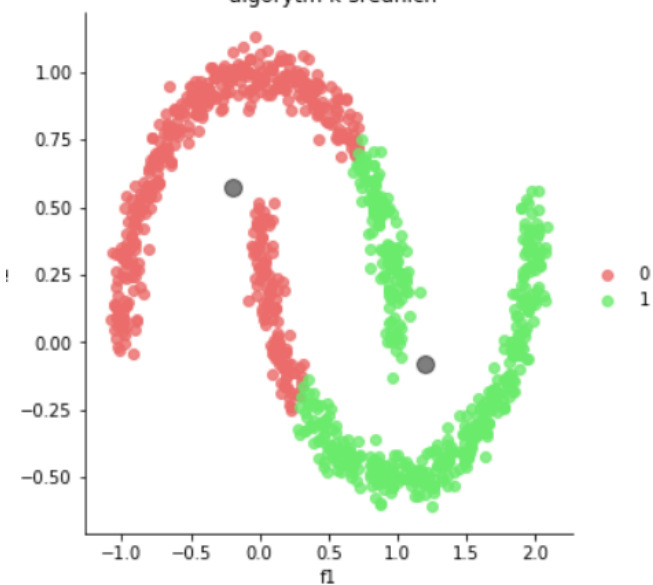
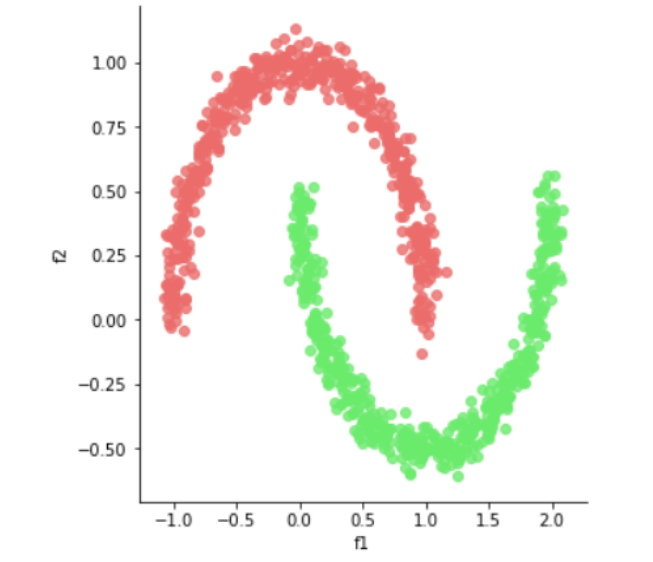
W poniższej analizie zostaną badane trzy zbiory danych: blobs (trzy zbiory punktów tworzące zwarte koła), moons (dwa zbiory punktów tworzące półksiężycy) oraz circles (zbiór tworzący pierścień kołowy). Porównamy również na nich trzy algorytmy klastrujące - DBSCAN, K-means i algorytm hierarchiczny. Klasteryzacja będzie przeprowadzona z użytą opcją *use training set*.

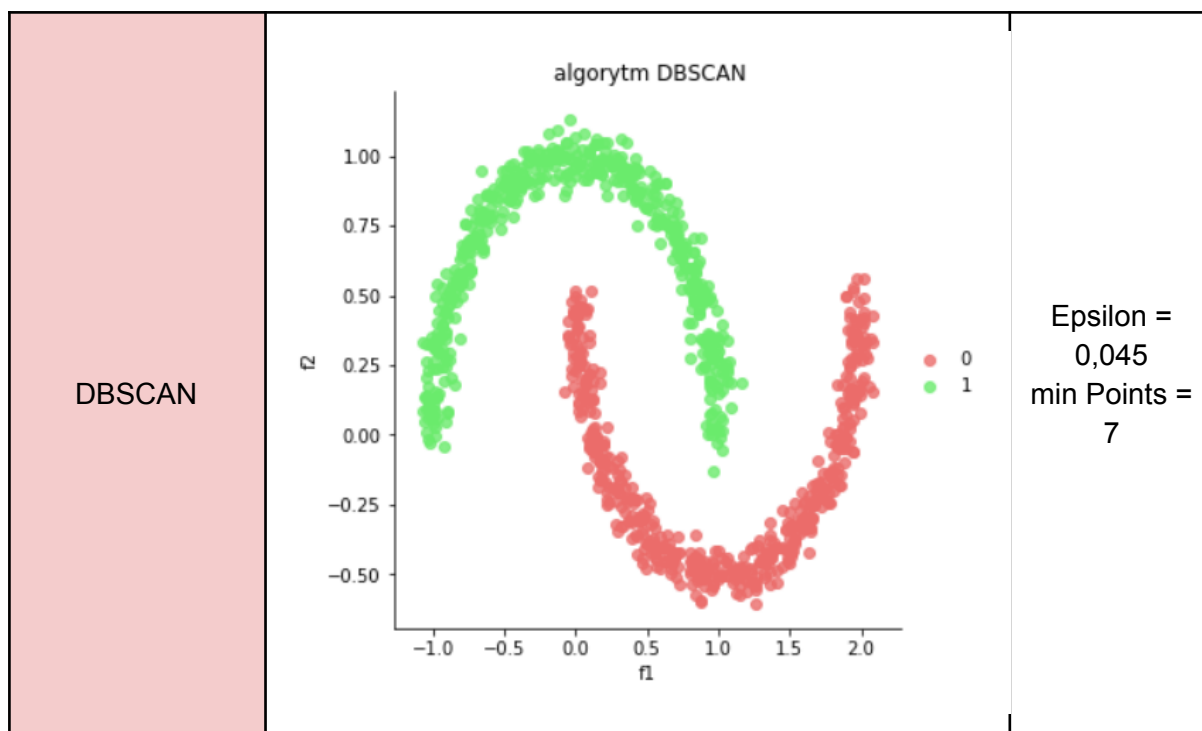
Trzy kształty koliste - Blobs

Algorytm	Wykres	Parametry
k-średnich		k = 3

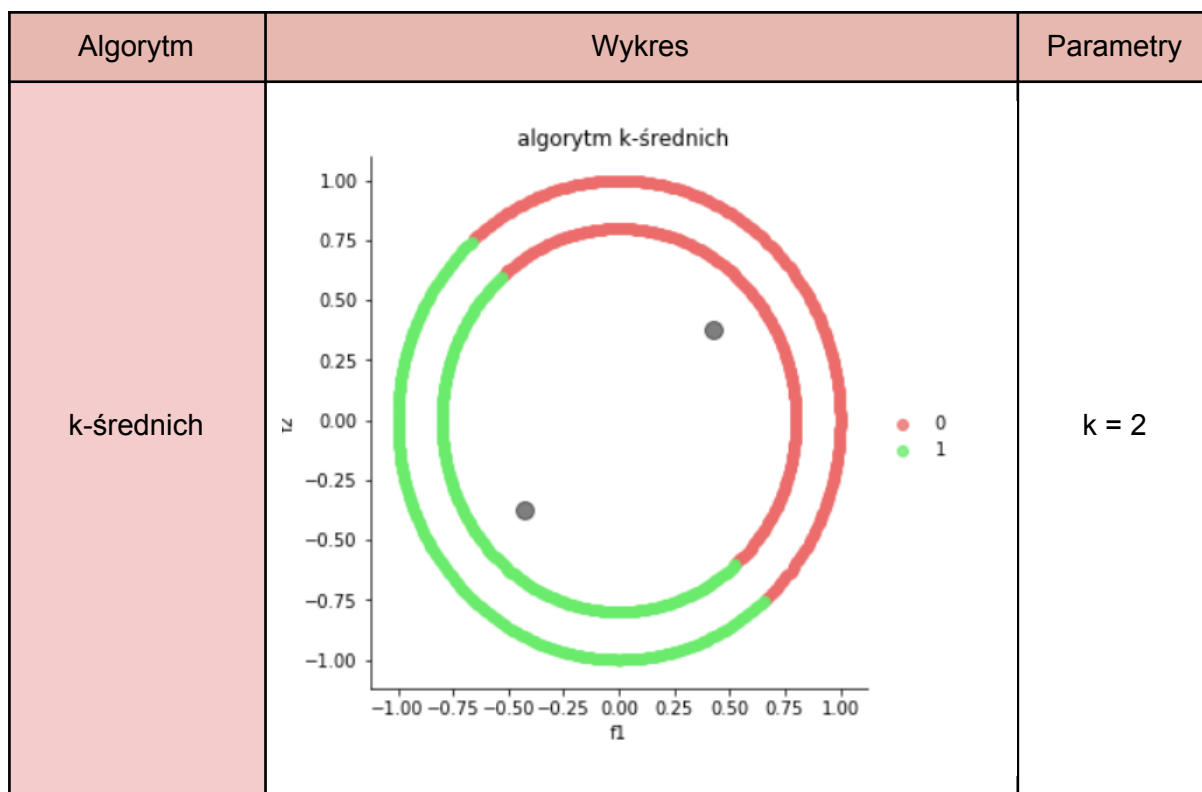
<p>hierarchiczny</p>	<p>algorytm hierarchiczny</p> 	<p>k = 3 linkType = average</p>
<p>DBSCAN</p>	<p>algorytm DBSCAN</p> 	<p>Epsilon = 0.0549 min Points = 7</p>

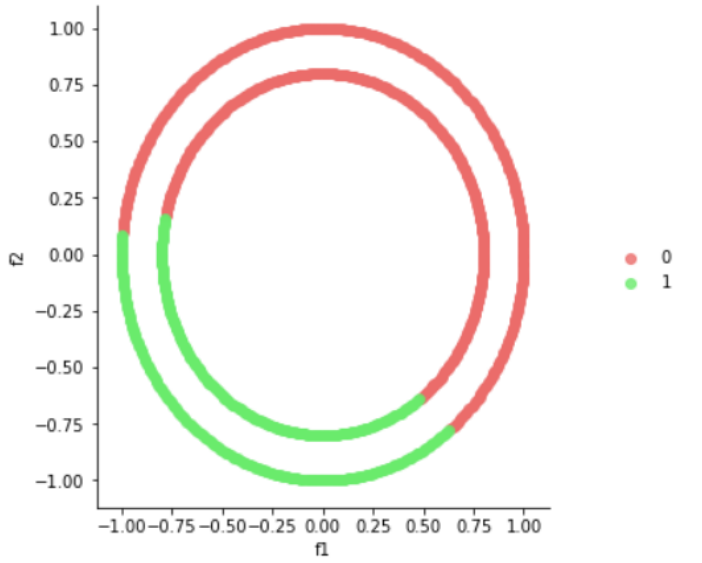
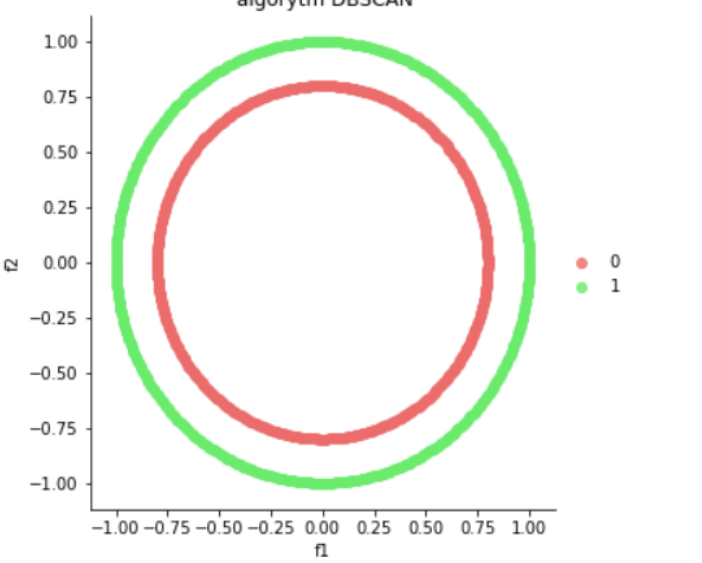
Dwa półksiężycy - moons

Algorytm	Wykres	Parametry
k-średnich	<p>algorytm k-średnich</p> 	k = 2
hierarchiczny	<p>algorytm hierarchiczny</p> 	k = 2



Pierścień - circles



<p>hierarchiczny</p>	<p>algorytm hierarchiczny</p> 	<p>k = 2</p>
<p>DBSCAN</p>	<p>algorytm DBSCAN</p> 	<p>Epsilon = 0.05 min Points = 6</p>

Powyższe wykresy zostały wygenerowane przy pomocy języka python.

Sprawdzimy jeszcze jak wyglądają i jakie cechy mają nasze dane.

Blobs

- Rozmiar
1500 2
- Jak wyglądają wartości początkowe
5.867498 8.1771519
5.613700 9.9329553
7.225084 10.4488619
6.762823 0.6051454
- Wartość najmniejsza: -7.923 -13.59028
Wartość największa: 10.529 12.27174

Circles

- Rozmiar
1500 2
- Jak wyglądają wartości początkowe
-0.6779994 -0.69875698
0.9314375 0.19139133
0.5482913 -0.00601715
0.8728369 0.37502332
- Wartość najmniejsza: -1.0949531 -1.0821617
Wartość największa: 1.0744448 1.1159302

Moons

- Rozmiar
1500 2
- Jak wyglądają wartości początkowe
0.4962713 -0.3427535
-0.1662996 0.9223421
0.7189560 0.6652904
-0.3378400 0.9120743
- Wartość najmniejsza: -1.12721 -0.6513
Wartość największa: 2.07686 1.1425

Właściwości algorytmów

Algorytm DBSCAN jest bardzo szybki, w większości przypadków okazał się szybszy od algorytmu hierarchicznego i k-średnich. Potrafi on również rozpoznawać gromady o dowolnym kształcie. Jest odporny na szum i wartości odstające. Problemy ma jednak przy danych o różnych gęstościach oraz dodatkowo gdy zbiór jest zbyt rzadki. Jest też bardzo wrażliwy na zmiany wartości minPoints oraz eps; próbkowanie, które wpływa znacząco na miary gęstości.

Algorytm k-średnich jest skalowalny, tworzy zwarte skupiska oraz jest szybszy dla danych o małych rozmiarach. Dokonując klasteryzacji tworzy zwarte skupiska. Lepiej więc działa dla dobrze ukształtowanych podziałów zbioru. Nie identyfikuje również wartości odstających i szumu. Innym minusem algorytmu jest sztywny podział danych według zadanej liczby klastrow. Nie jest brana pod uwagę rzeczywista struktura zbioru tylko z góry przypisana wcześniej wspomniana liczba klastrow.

Algorytm hierarchiczny, jako jedyny z omawianych, nie wymaga wcześniejszego określenia liczby grup. Jednakże przez to wymaga warunku zakończenia. Bardzo dobrze się sprawdza dla małych zestawów danych. Ma mniejszą wrażliwość na szum i wartości odstające. Trudność mu jednak sprawiają skupiska o różnej wielkości oraz wypukłe kształty. Metody te też nie są skalowalne przy dużym zbiorze.

Podsumowanie

W naszych analizach algorytmu miały różną skuteczność. Dla danych moons tylko algorytm K-means nie poradził sobie zadowalająco. Natomiast dla danych Blobs był on najbardziej skutecznym. Pozostałe algorytmy - hierarchiczny oraz DBSCAN - również poradziły sobie dobrze, choć to właśnie DBSCAN wypadł w tym przypadku najgorzej. Dla danych Circles poprawną klasyfikację otrzymujemy dla algorytmu hierarchicznego oraz DBSCAN - K-means ponownie nie poradził z nią sobie poprawnie. Wszystkie te wartości zostały uzyskane poprzez kombinację różnych parametrów dla każdego z wspomnianych algorytmów.

Tak więc podsumowując, na podstawie właściwości każdego z tych trzech algorytmów oraz otrzymanych wartości z procesu klasteryzacji możemy wywnioskować, iż kształt danych klastrow ma znaczący wpływ na działanie różnych algorytmów. Z niektórymi danymi poszczególne algorytmy nie potrafią sobie satysfakcjonująco poradzić, natomiast zmiana parametrów (czy to wyznaczonej liczby klastrow, epsilon lub minPoints) potrafi mocno wpłynąć na otrzymane wyniki.