Analiza danych

prof. UAM dr hab. Tomasz Górecki

tomasz.gorecki@amu.edu.pl
http://drizzt.home.amu.edu.pl

Zakład Statystyki Matematycznej i Analizy Danych
Wydział Matematyki i Informatyki
Uniwersytet im. Adama Mickiewicza w Poznaniu

Idea

Analiza skupień (*ang. cluster analysis*) jest narzędziem analizy danych służącym do grupowania n obiektów, opisanych za pomocą wektora p-cech, w K niepustych, rozłącznych i możliwie jednorodnych grup – skupień. Obiekty należące do danego skupienia powinny być „podobne" do siebie, a obiekty należące do różnych skupień powinny być z kolei możliwie mocno „niepodobne" do siebie. Głównym celem tej analizy jest wykrycie z zbiorze danych, tzw. „naturalnych" skupień, czyli skupień, które dają się w sensowny sposób interpretować.

Algorytm zachłanny

Zwróćmy uwagę, że pod tym terminem kryje się szereg różnych algorytmów. Koncepcyjnie, najprostszym byłby następujący. Ustalamy liczbę skupień K oraz kryterium optymalnego podziału obiektów. Przeszukujemy wszystkie możliwe podziały n obiektów na K skupień, wybierając najlepszy podział ze względu na przyjęte kryterium optymalności. Bezpośrednie sprawdzenie wszystkich możliwych podziałów jest jednak, nawet przy niewielkim n, praktycznie niemożliwe. Ich liczba bowiem jest równa

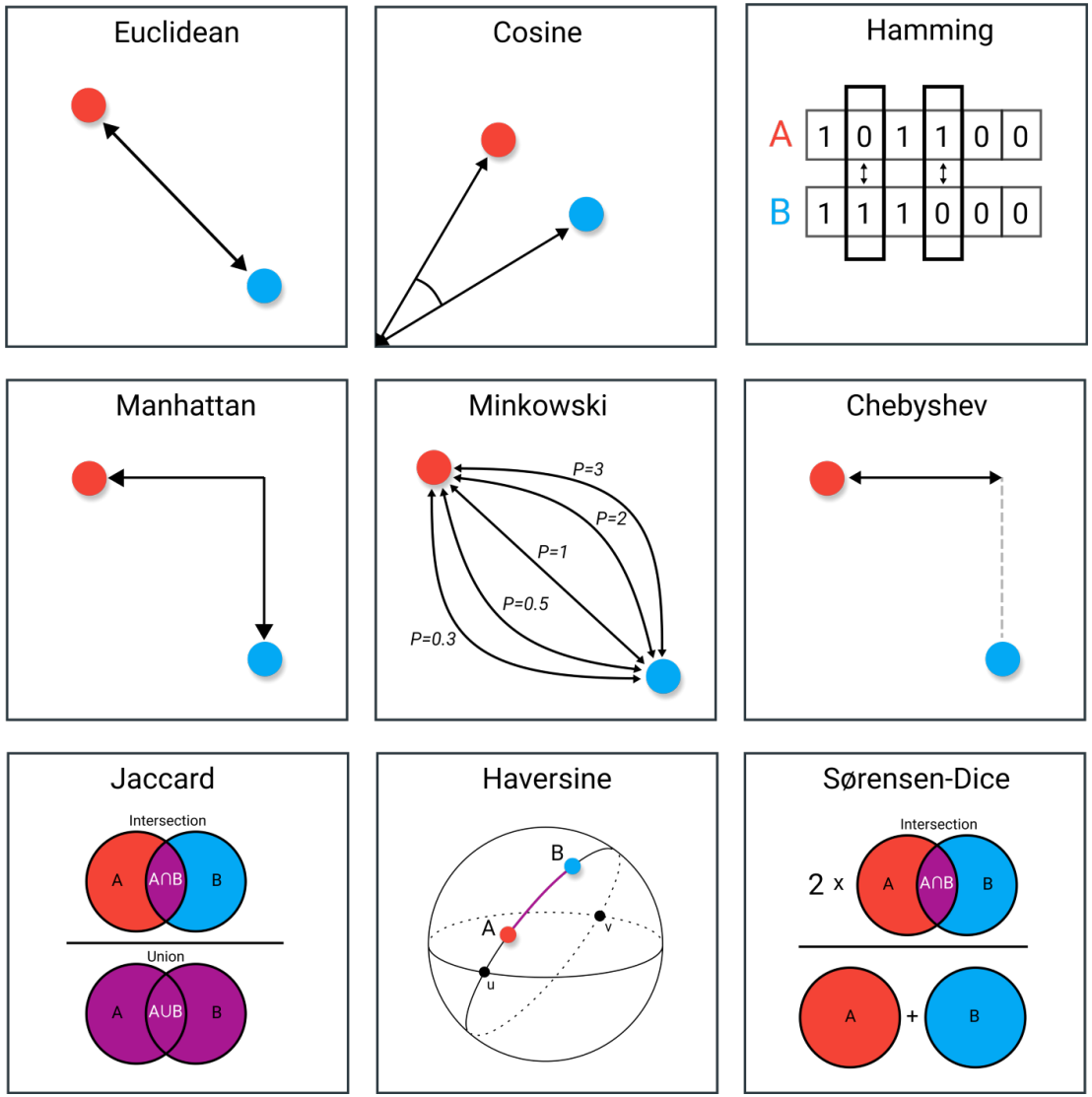
$$\frac{1}{K!}\sum_{k=1}^{K}(-1)^{K-k}\binom{K}{k}k^n$$

i np. dla $n = 100$ obiektów i $K = 4$ skupień jest rzędu 10^{58}.

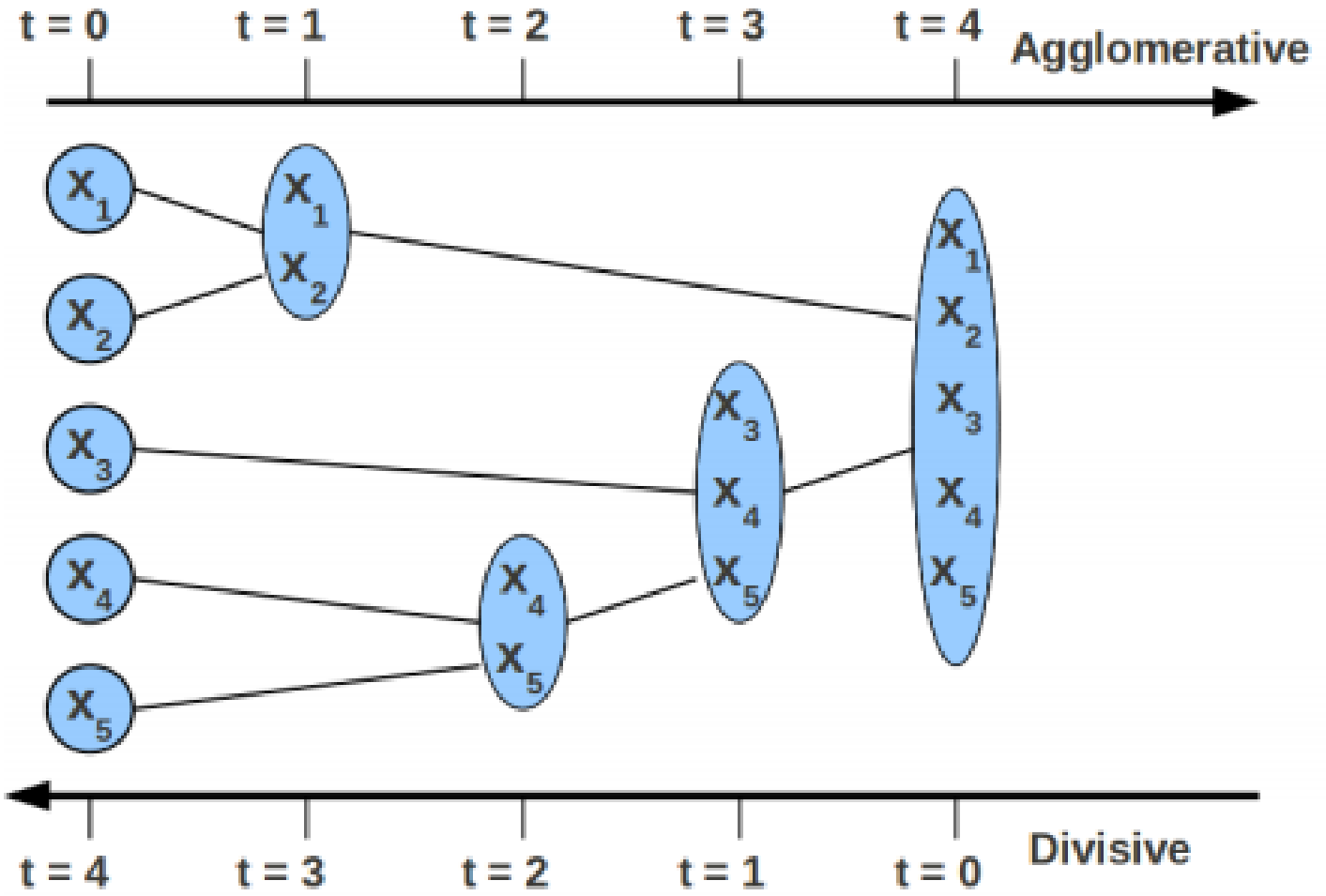
Algorytmy hierarchiczne – idea

Najprostszą i zarazem najczęściej używaną metodą analizy skupień jest metoda hierarchiczna. Wspólną cechą krokowych algorytmów tej metody jest wyznaczanie skupień poprzez łączenie (aglomerację) powstałych, w poprzednich krokach algorytmu, mniejszych skupień. Inne wersje tej metody zamiast idei łączenia skupień, bazują na pomyśle ich dzielenia. Podstawą wszystkich algorytmów tej metody jest odpowiednie określenie miary niepodobieństwa obiektów. Miary niepodobieństwa, to semi-metryki (a często również metryki) na przestrzeni próby $\mathcal{X}$.

Algorytmy hierarchiczne – idea



Algorytmy hierarchiczne – idea



Algorytmy hierarchiczne – algorytm aglomeracyjny

W pierwszym kroku każdy z obiektów tworzy oddzielne skupienie. Zatem skupień tych jest n. W kroku drugim w jedno skupienie połączone zostają dwa najbardziej podobne do siebie obiekty – w sensie wybranej miary niepodobieństwa obiektów. Otrzymujemy zatem $n-1$ skupień. Postępując analogicznie, tzn. łącząc (wiążąc) ze sobą skupienia złożone z najbardziej podobnych do siebie obiektów, w każdym następnym kroku, liczba skupień maleje o jeden. Obliczenia prowadzimy do momentu uzyskania zadeklarowanej, końcowej liczby skupień K lub do połączenia wszystkich obiektów w jedno skupienie.

Algorytmy hierarchiczne – dendrogram

Graficzną ilustracją algorytmu jest dendrogram (*ang. dendrogram*), czyli drzewo binarne, którego węzły reprezentują skupienia, a liście obiekty. Liście są na poziomie zerowym, a węzły na wysokości odpowiadającej mierze niepodobieństwa pomiędzy skupieniami reprezentowanymi przez węzły potomki.

Algorytmy hierarchiczne – metody wiązania skupień

Algorytm ten wykorzystuje nie tylko miary niepodobieństwa pomiędzy obiektami, potrzebne są nam również metody wiązania skupień. Niech R i S oznaczają skupienia, a $\rho(R, S)$ oznacza miarę niepodobieństwa pomiędzy nimi. Poniżej podano najczęściej wykorzystywane sposoby jej określenia.

Algorytmy hierarchiczne – metoda pojedynczego wiązania

Metoda pojedynczego wiązania (najbliższego sąsiedztwa) (*ang. single linkage*). Miara niepodobieństwa pomiędzy dwoma skupieniami jest określona jako najmniejsza miara niepodobieństwa między dwoma obiektami należącymi do różnych skupień:

$$\rho(R, S) = \min_{i \in R, j \in S} \rho(\boldsymbol{x}_i, \boldsymbol{x}_j).$$

Zastosowanie tego typu odległości prowadzi do tworzenia wydłużonych skupień, tzw. łańcuchów. Pozwala na wykrycie obserwacji odstających, nie należących do żadnej z grup, i warto przeprowadzić klasyfikację za jej pomocą na samym początku, aby wyeliminować takie obserwacje i przejść bez nich do właściwej części analizy.

Algorytmy hierarchiczne – metoda pojedynczego wiązania

Metoda pełnego wiązania (najdalszego sąsiedztwa) (*ang. complete linkage*). Miara niepodobieństwa pomiędzy dwoma skupieniami jest określona jako największa miara niepodobieństwa między dwoma obiektami należącymi do różnych skupień:

$$\rho(R,S) = \max_{i \in R, j \in S} \rho(\boldsymbol{x}_i, \boldsymbol{x}_j).$$

Metoda ta jest przeciwieństwem metody pojedynczego wiązania. Jej zastosowanie prowadzi do tworzenia zwartych skupień o małej średnicy. Ma tendencję do dzielenia dużych skupień.

Algorytmy hierarchiczne – metoda pojedynczego wiązania

Metoda średniego wiązania (*ang. average linkage*). Miara niepodobieństwa pomiędzy dwoma skupieniami jest określona jako średnia miara niepodobieństwa między wszystkimi parami obiektów należących do różnych skupień:

$$\rho(R,S) = \frac{1}{n_R n_S} \sum_{i \in R} \sum_{j \in S} \rho(\boldsymbol{x}_i, \boldsymbol{x}_j),$$

gdzie n_R i n_S są liczbami obiektów wchodzących w skład skupień R i S odpowiednio.

Metoda ta jest swoistym kompromisem pomiędzy metodami pojedynczego i pełnego wiązania. Ma ona jednak zasadniczą wadę. W odróżnieniu od dwóch poprzednich wykorzystywana w niej miara niepodobieństwa nie jest niezmiennicza ze względu na monotoniczne przekształcenia miar niepodobieństwa pomiędzy obiektami.

Algorytmy hierarchiczne – inne metody wiązania skupień

Omówione metody wiązania skupień, choć najczęściej stosowane, nie są jedyne. W przypadku gdy liczebności skupień są zdecydowanie różne, zamiast metodą średniego wiązania możemy posługiwać się jej ważonym odpowiednikiem. Wagami są wtedy liczebności poszczególnych skupień. Inna popularna metoda wiązania skupień pochodzi od Warda (1963). Do obliczania miary niepodobieństwa pomiędzy skupieniami wykorzystuje on podejście analizy wariancji. Metoda daje bardzo dobre wyniki (grupy bardzo homogeniczne), jednak ma skłonność do tworzenia skupień o podobnych rozmiarach. Często nie jest też w stanie zidentyfikować grup o szerokim zakresie zmienności poszczególnych cech oraz niewielkich grup.

Literatura

Ward, J.H. (1963). Hierarchical grouping to optimize an objective function. Journal of the American Statistical Association 58(301):236–244.

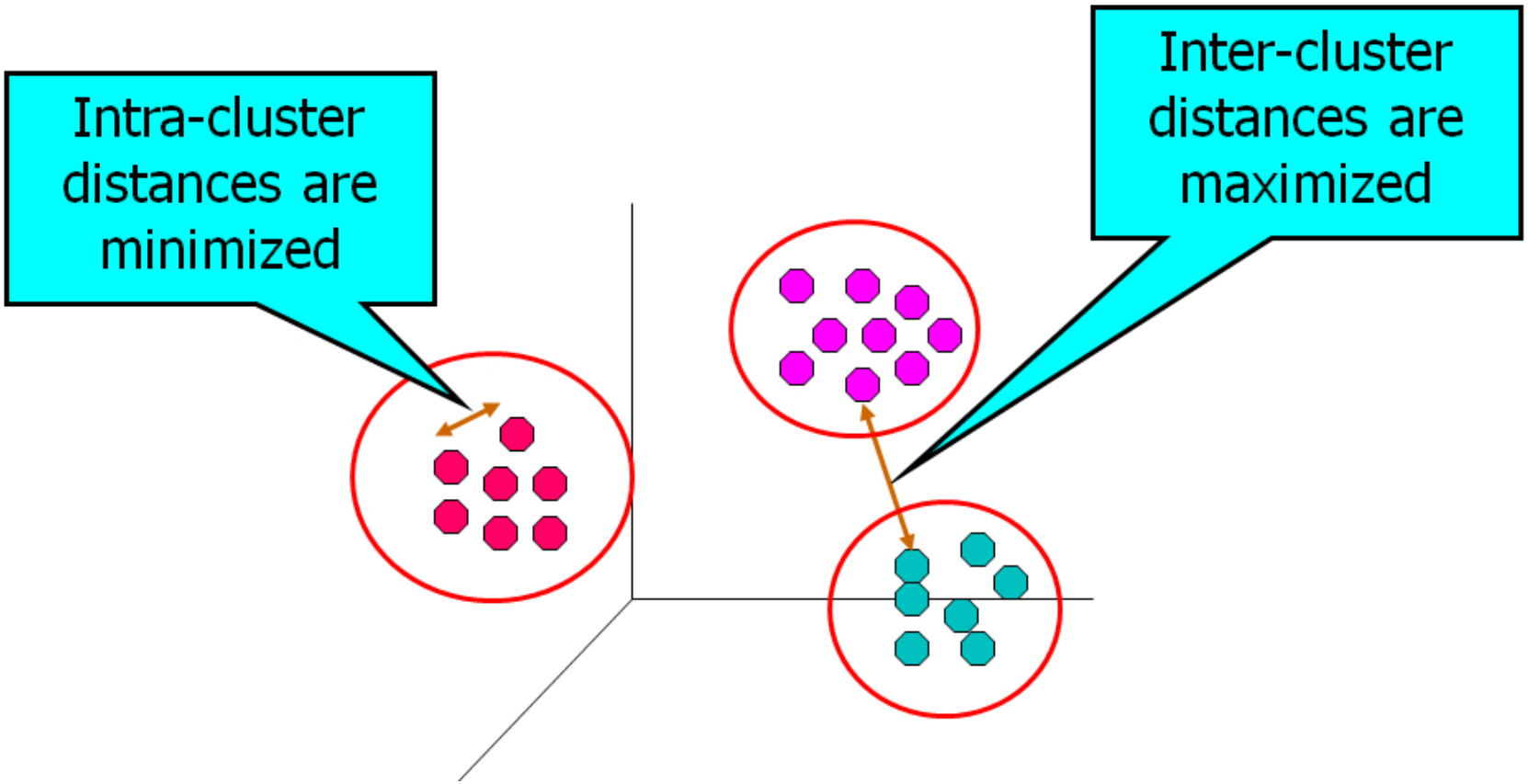
Algorytm aglomeracyjny – podsumowanie

Algorytm aglomeracyjny jest uniwersalny w tym sensie, że może być on stosowany zarówno do danych ilościowych jak i jakościowych. Wykorzystuje on jedynie miary niepodobieństwa pomiędzy obiektami oraz pomiędzy skupieniami. Należy podkreślić zasadniczy wpływ wybranej miary niepodobieństwa na uzyskane w końcowym efekcie skupienia. Do ustalenia końcowej liczby skupień wykorzystać możemy wykresy rozrzutu (przy wielu wymiarach w układzie dwóch pierwszych składowych głównych). Pomocny może być także dendrogram. Ustalamy wtedy progową wartość miary niepodobieństwa pomiędzy skupieniami, po przekroczeniu której zatrzymany zostaje proces ich dalszego łączenia.
Złożoność pamięciowa to $O(n^2)$, a naiwna implementacja ma złożoność obliczeniową $O(n^3)$, którą można zmniejszyć do $O(n^2)$.

Metoda K-średnich – idea

Najbardziej popularnym, niehierarchicznym algorytmem analizy skupień jest algorytm K-średnich (*ang. k-means*).
Przyporządkowanie n obiektów do zadanej liczby skupień K, odbywa się niezależnie dla każdej wartości K – nie bazując na wyznaczonych wcześniej mniejszych lub większych skupieniach.
Niech C_K oznacza funkcję, która każdemu obiektowi (dokładnie jego numerowi), przyporządkowuje numer skupienia do którego jest on przyporządkowany (przy podziale na K skupień). Zakładamy, że wszystkie cechy są ilościowe o wartościach rzeczywistych (przestrzeń próby to $\mathbb{R}^p$). Główną ideą metody K-średnich jest taka alokacja obiektów, która minimalizuje zmienność wewnątrz powstałych skupień, a co za tym idzie maksymalizuje zmienność pomiędzy skupieniami.

Metoda K-średnich – idea



Metoda K-średnich – idea

Dla ustalonej funkcji C_K, przez $W(C_K)$ i $B(C_K)$ oznaczmy macierze zmienności odpowiednio wewnątrz i pomiędzy skupieniami. Poniższa, znana z analizy wariancji, zależność opisuje związek pomiędzy tymi macierzami:

$$T = W(C_K) + B(C_K),$$

gdzie T jest niezależną od dokonanego podziału na skupienia macierzą zmienności całkowitej. Powszechnie stosowane algorytmy metody K-średnich minimalizują ślad macierzy $W(C_K)$.

Metoda K-średnich – algorytm

1. W losowy sposób rozmieszczamy n obiektów w K skupieniach. Niech funkcja $C_K^{(1)}$ opisuje to rozmieszczenie.
2. Dla każdego z K skupień obliczamy wektory średnich $\bar{\boldsymbol{x}}_k$, $(k = 1, 2, \ldots, K)$.
3. Rozmieszczamy ponownie obiekty w K skupieniach, w taki sposób że
$$C_K^{(l)}(i) = \arg \min_{1 \leq k \leq K} \rho_2(\boldsymbol{x}_i, \bar{\boldsymbol{x}}_k).$$
4. Powtarzamy kroki drugi i trzeci aż do momentu, gdy przyporządkowanie obiektów do skupień pozostanie niezmienione, tzn. aż do momentu, gdy $C_K^{(l)} = C_K^{(l-1)}$.

Metoda K-średnich – algorytm

Na metodę K-średnich można spojrzeć jak na metodę minimalizacji sumy kwadratów odległości punktów od środków skupień:

$$\sum_{i=1}^{K} \sum_{\boldsymbol{x} \in S_i} ||\boldsymbol{x} - \bar{\boldsymbol{x}}_i||^2,$$

gdzie $\boldsymbol{S} = \{S_1, S_2, \ldots, S_K\}$ są zbiorami rozłącznych punktów (skupieniami).

https://imgur.com/tBkCqXJ

Metoda K-średnich – modyfikacje

Istnieje wiele modyfikacji powyższego algorytmu. Przykładowo, losowe rozmieszczenie elementów w skupieniach – krok pierwszy algorytmu, zastąpione zostaje narzuconym podziałem, mającym na celu szybsze ustabilizowanie się algorytmu.

Wszystkie wersje algorytmu K-średnich są zbieżne. Nie gwarantują one jednak zbieżności do optymalnego rozwiązania C_K^*. Niestety, w zależności od początkowego podziału, algorytm zbiega do zazwyczaj różnych lokalnie optymalnych rozwiązań. W związku z tym, aby uzyskać najlepszy podział, zaleca się często wielokrotne stosowanie tego algorytmu z różnymi, wstępnymi rozmieszczeniami obiektów.

Metoda K-średnich – wybór K

Algorytm metody K-średnich bazuje na minimalizacji zmienności wewnątrz powstałych skupień, wyrażonej poprzez $W_K = \log(\text{tr}(W(C_K)))$. Zwróćmy uwagę, że zmienność ta maleje wraz ze wzrostem liczby skupień (dla $K = n$ jest wręcz zerowa). Wartości te nanosimy na wykres podobny do wykresu osypiska. Analizujemy różnice pomiędzy W_K i W_{K+1} poszukując różnic zdecydowanie większych od pozostałych. Sugeruje to, podział na skupienia. Trudno jest jednak precyzyjnie określić, którą z różnic uznać za istotnie małą.

Metoda K-średnich – wybór K – indeks CH

W literaturze znaleźć można wiele pomysłów na automatyczne wyznaczania końcowej liczby skupień. Dwa z nich zasługują na szczególną uwagę.

Caliński i Harabasz (1974) zaproponowali aby końcową liczbę skupień wybierać w oparciu o wartości indeksu postaci:

$$CH(K) = \frac{\text{tr}(B(C_K))/(K-1)}{\text{tr}(W(C_K))/(n-K)}.$$

Optymalną wartość K dobieramy tak, aby ją zmaksymalizować.

Literatura

Caliński, T., Harabasz, J. (1974). *A dendrite method for cluster analysis*. Communications in Statistics 3(1):1–27.

Metoda K-średnich – wybór K – zarys

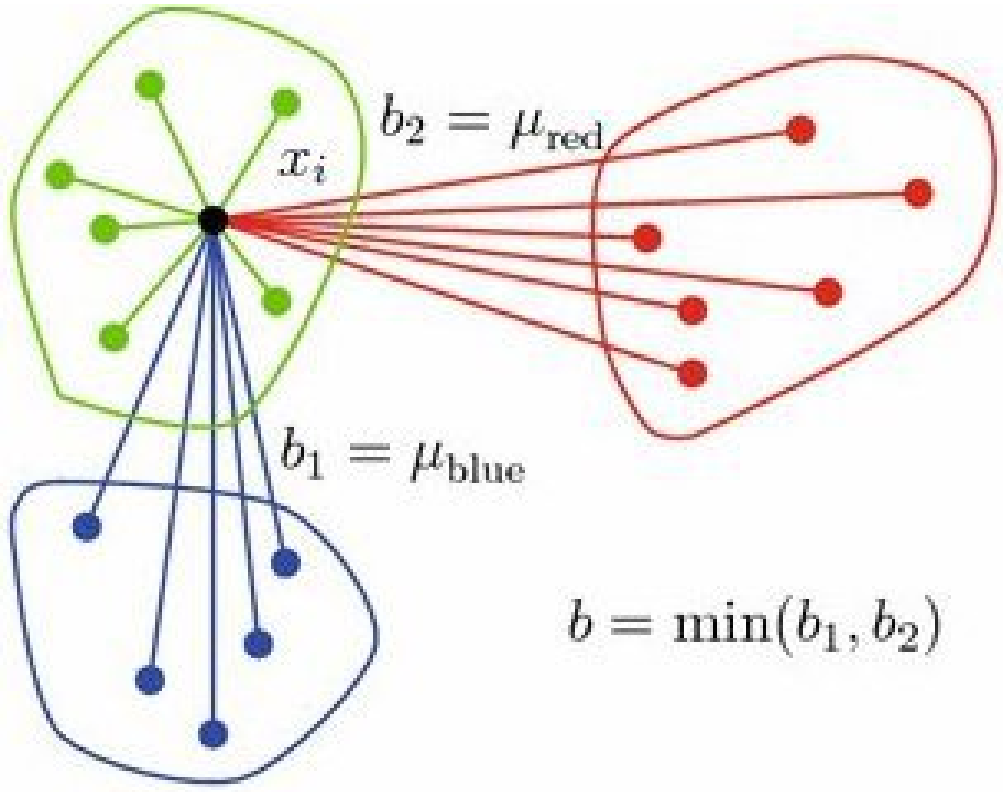
Z powodu łatwej możliwości wizualizacji coraz popularniejsza staje się miara zwana zarysem (*ang. silhouette*). Dla każdej obserwacji i niech $a(i)$ będzie średnią miarą niepodobieństwa pomiędzy nią, a wszystkimi obserwacjami z tej samej grupy (im mniejsza wartość tym lepsze dopasowanie obserwacji do grupy).

a(i): avg distance between i and all other datapoints within cluster

a(i)

Metoda K-średnich – wybór K – zarys

Następnie znajdujemy średnie odległości obserwacji i do pozostałych skupień. Przez $b(i)$ oznaczmy najmniejszą z tych średnich (średnia odległość do najbliższego skupienia, do którego i nie należy).



Metoda K-średnich – wybór K – zarys

Zarys definiujemy jako:

$$s_i = \frac{b(i) - a(i)}{\max(a(i), b(i))}.$$

Miara ta przyjmuje wartości od -1 do 1. Obserwacje, dla których s_i jest bliskie 1 są poprawnie przydzielone, jeśli s_i jest bliskie 0 obserwacja leży na granicy skupień, jeśli natomiast s_i jest ujemne obserwacja znajduje się w złym skupieniu. Średnia wartość zarysu $\bar{s}_k$, dla każdego skupienia mówi o tym jak dobrze dane są przydzielone do tego skupienia. Z tego względu średni zarys dla całego zbioru danych może służyć jako miara jakości podziału.

Metoda K-średnich – wybór K – zarys

Współczynnik zarysu (*ang. silhouette coefficient*) ma postać $SC = \max_k \bar{s}_k$, i wykorzystujemy go do wyboru liczby skupień. Zarys podlega wizualizacji za pomocą wykresu zarysu (*ang. silhouette plot*).

Literatura


Kaufman, L., Rousseeuw, P.J. (1990). *Finding groups in data: An introduction to cluster analysis*. Wiley.


Rousseeuw, P.J. (1987). *Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis*. Computational and Applied Mathematics 20:53–65.

Inne metody oceny jakości grupowania

- Indeks Daviesa-Bouldina (*ang. Davies-Bouldin index*):

$$DB = \frac{1}{n} \sum_{i=1}^{K} \max_{j \neq i} \frac{\sigma_i + \sigma_j}{d(c_i, c_j)},$$

gdzie σ_i jest średnią odległością wszystkich punktów ze skupienia i do jego środka, a $d(c_i, c_j)$ jest odległością pomiędzy środkami skupień i oraz j.

- Indeks Dunna (*ang. Dunn index*):

$$D = \frac{\min\limits_{1 \leq i < j \leq K} d(i,j)}{\max\limits_{1 \leq k \leq K} d'(k)},$$

gdzie $d(i,j)$ jest odległością pomiędzy skupieniami i oraz j, a $d'(k)$ odległością wewnątrz skupienia k.

Inne metody oceny jakości grupowania

Indeks DB minimalizujemy, natomiast indeks Dunna maksymalizujemy.

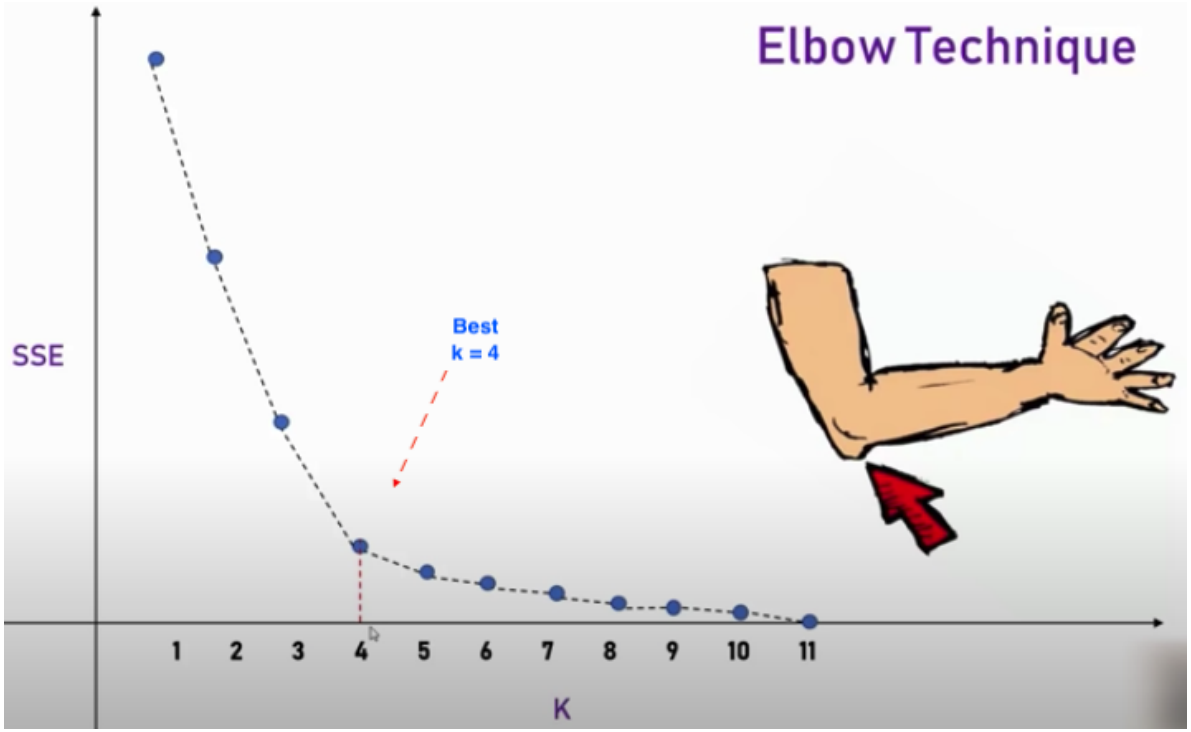
Literatura


Davies, D.L., Boudin, D.W. (1979). *A Cluster Separation Measure*. IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-1(2):224–227.


Dunn, J.C. (1973). *A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters*. Journal of Cybernetics 3(3):32–57.

Inne metody oceny jakości grupowania

- Metoda łokcia (*ang. elbow method*) – metoda ta polega na wykreśleniu wyjaśnionej zmienności jako funkcji liczby skupień i wybraniu zgięcia krzywej jako liczby skupień do użycia.

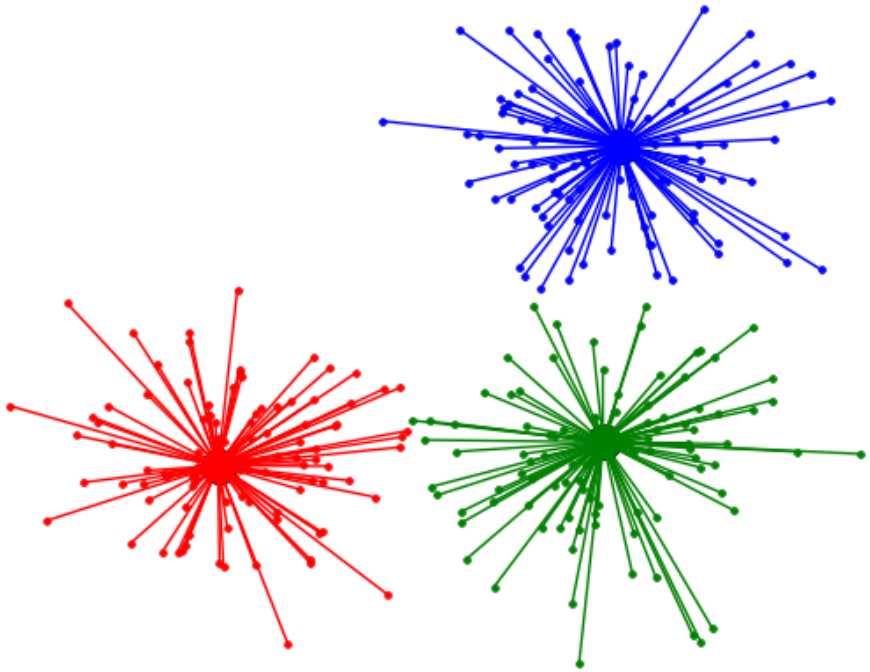


Metoda *K*-średnich – założenia (odległość euklidesowa)

1. Wszystkie zmienne ciągłe,
2. Rozkłady wszystkich zmiennych symetryczne,
3. Podobne średnie wszystkich zmiennych,
4. Podobne wariancje wszystkich zmiennych.

Affinity propagation

Algorytm affinity propagation (AP) polega na iteracyjnej wymianie komunikatów pomiędzy punktami w celu znalezienia rozwiązania, które maksymalizuje sumę podobieństw punktów do przypisanych im archetypów (*ang. exemplar*). Jest to zatem algorytm tego samego typu co metoda *k*-średnich.



Affinity propagation

Cechy AP:

- Nie nakłada żadnych ograniczeń na macierz podobieństwa, podobieństwa mogą być nawet asymetryczne.
- Wszystkie punkty jednocześnie są rozważane jako potencjalne archetypy.
- Liczba skupień nie jest zadawana wprost, wynika z wartości „preferencji" przypisanej każdemu punktowi (preferencja określa jak silnie dany punkt powinien być preferowany jako archetyp w ostatecznym wyniku).

Literatura

Frey, B.J., Dueck, D. (2007). *Clustering by passing messages between data points*. Science 315(5814):972–976

Affinity propagation – przykład

*Dane (**X**)*

Niech każdy obiekt będzie reprezentowany jako punkt w pięciowymiarowej przestrzeni rzeczywistej $\mathbb{R}^5$. Warto zauważyć, że w tym przykładzie zmienne są w tej samej skali. Na ogół jednak zmienne znajdują są w różnych skalach i należy je znormalizować.

Participant	Tax Rate	Fee	Interest Rate	Quantity Limit	Price Limit
Alice	3	4	3	2	1
Bob	4	3	5	1	1
Cary	3	5	3	3	3
Doug	2	1	3	3	2
Edna	1	1	3	2	3

Affinity propagation – przykład

W pierwszym kroku należy wyznaczyć macierz podobieństwa ($\boldsymbol{S}$) – *ang. similarity matrix*. Z wyjątkiem elementów na przekątnej, każda komórka w tej macierzy jest obliczana jako ujemna odległość euklidesowa podniesiona do kwadratu, czyli

$$s_{ik} = -||\boldsymbol{x}_i - \boldsymbol{x}_k||.$$

Np. dla podobieństwa między Alice i Bobem, suma kwadratów różnic wynosi:

$$(3-4)^2 + (4-3)^2 + (3-5)^2 + (2-1)^2 + (1-1)^2 = 7.$$

Zatem wartość podobieństwa wynosi $s_{12} = s_{21} = -7$.

Affinity propagation – przykład

Przekątna macierzy $\boldsymbol{S}$ (tj. s_{ii}) jest szczególnie ważna, ponieważ reprezentuje preferencje obiektu, czyli to, jak prawdopodobne jest, że dany obiekt stanie się archetypem. Zazwyczaj inicjalizowana jest jako mediana podobieństw wszystkich par danych wejściowych. Algorytm będzie zbiegał do małej liczby skupień, jeśli dla przekątnej zostanie wybrana mniejsza wartość i odwrotnie. Dlatego w naszym przykładzie elementy diagonalne macierzy podobieństwa wypełniamy liczbą -22, czyli najniższą liczbą spośród wyliczonych podobieństw.

Affinity propagation – przykład

Macierz podobieństwa (S)

Participant	Alice	Bob	Cary	Doug	Edna
Alice	-22	-7	-6	-12	-17
Bob	-7	-22	-17	-17	-22
Cary	-6	-17	-22	-18	-21
Doug	-12	-17	-18	-22	-3
Edna	-17	-22	-21	-3	-22

Affinity propagation – przykład

Następnie wykonywane są dwa kroki przekazywania wiadomości, które aktualizują dwie macierze:

- Macierz odpowiedzialności ($\boldsymbol{R}$) – *ang. responsibility matrix*, która określa jak dobrze $\boldsymbol{x}_k$ jest przystosowany do pełnienia roli wzorca dla $\boldsymbol{x}_i$, w stosunku do innych kandydatów na wzorce dla $\boldsymbol{x}_i$.
- Macierz dostępności ($\boldsymbol{A}$) – *ang. availability matrix* ,która określa jak „odpowiednie" byłoby dla $\boldsymbol{x}_i$ wybranie $\boldsymbol{x}_k$ jako swojego wzorca, biorąc pod uwagę preferencje innych punktów dla $\boldsymbol{x}_k$ jako wzorca.

Affinity propagation – przykład

Obie macierze są inicjalizowane na zera. Następnie algorytm wykonuje iteracyjnie następujące aktualizacje:

$$r_{ik} = s_{ik} - \max_{k' \neq k}(a_{ik'} + s_{ik'})$$

$$a_{ik} = \begin{cases} \min\left(0, r_{kk} + \sum_{i' \notin \{i,k\}} \max(0, r_{i'k})\right) & \text{dla } i \neq k \\ \sum_{i' \neq k} \max(0, r_{i'k}) & \text{dla } i = k. \end{cases}$$

Iteracje wykonywane są do momentu, gdy albo granice skupisk pozostaną niezmienione przez określoną liczbę iteracji, albo osiągnięta zostanie pewna z góry określona liczba iteracji. Z końcowych macierzy wyodrębniane są archetypy jako te, dla których suma $c_{ii} = r_{ii} + a_{ii}$ jest dodatnia.

Affinity propagation – przykład

Macierz odpowiedzialności (R)

Participant	Alice	Bob	Cary	Doug	Edna
Alice	-16	-1	1	-6	-11
Bob	10	-15	-10	-10	-15
Cary	11	-11	-16	-12	-15
Doug	-9	-14	-15	-19	9
Edna	-14	-19	-18	14	-19

Np. dla Boba (kolumna) i Alice (wiersz) wynosi -1, co jest różnicą ich podobieństwa (-7) i maksimum pozostałych podobieństw dla wiersza Alice (-6), czyli $r_{12} = -7 - (-6) = -1$.

Affinity propagation – przykład

*Macierz dostępności (**A**)*

Participant	Alice	Bob	Cary	Doug	Edna
Alice	21	-15	-16	-5	-10
Bob	-5	0	-15	-5	-10
Cary	-6	-15	1	-5	-10
Doug	0	-15	-15	14	-19
Edna	0	-15	-15	-19	9

Np. samodostępność Alice jest sumą dodatnich odpowiedzialności kolumny Alice z wyłączeniem samoodpowiedzialności Alice czyli $a_{11} = 10 + 11 + 0 + 0 = 21$. Podobnie dostępność Boba (kolumna) dla Alice (wiersz) to odpowiedzialność własna Boba plus suma pozostałych pozytywnych odpowiedzialności Boba w kolumnie z wyłączeniem odpowiedzialności Boba wobec Alice, czyli $a_{12} = -15 + 0 + 0 + 0 = -15$.

Affinity propagation – przykład

Macierz kryterium (C)

Participant	Alice	Bob	Cary	Doug	Edna
Alice	**5**	-16	-15	-11	-21
Bob	**5**	-15	-25	-15	-25
Cary	**5**	-26	-15	-17	-25
Doug	-9	-29	-30	**-5**	-10
Edna	-14	-34	-33	**-5**	-10

Np. wartość kryterialna Boba (kolumna) dla Alice (wiersz) jest sumą odpowiedzialności i dostępności Boba dla Alicji, czyli $c_{12} = -1 + -15 = -16$. Najwyższa wartość kryterium w każdym wierszu jest oznaczana jako archetyp. Wiersze, które mają ten sam archetyp, znajdują się w tym samym skupieniu. Zatem w naszym przykładzie Alice, Bob i Cary tworzą jedno skupienie, podczas gdy Doug i Edna tworzą drugie.

Metoda hierarchiczna, a niehierarchiczna

Obie metody mają swoje wady i zalety. W przypadku metod hierarchicznych istnieje wiele algorytmów dających różne wyniki, z których nie jesteśmy w stanie określić, które rozwiązanie jest najlepsze. Poza tym nie ma możliwości korekty rozwiązania, obiekt raz przydzielony do klasy już w niej pozostaje. Ostatecznie metody hierarchiczne są mało wydajne w przypadku dużych zbiorów danych (duża czaso- i pamięciożerność). Główną wadą metod optymalizacyjnych jest konieczność zadania liczby klas z góry. Dodatkowo bardzo duże znaczenie ma wybór punktów początkowych. W praktyce często metoda hierarchiczna służy do wstępnej analizy i wyznaczenia punktów startowych dla metody *k*-średnich. Analiza skupień nie jest odporna na zmiany skali, oznacza to, że jeśli różne zmienne mają różne skale, to te największe mogą zdominować odległości.

DBSCAN – analiza skupień bazująca na gęstościach punktów

Poprzednio omówione metody są dostosowane do wykrywania skupień sferycznych lub wypukłych. Innymi słowy działają dobrze dla zwartych i dobrze rozdzielonych skupień. Dodatkowo duży wpływ na wyniki mają obserwacje odstające oraz szum w danych.

`https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/`

DBSCAN – analiza skupień bazująca na gęstościach punktów

Algorytm DBSCAN (*ang. Density-Based Spatial Clustering and Application with Noise*) został zaproponowany w 1996 roku przez Ester i innych. Zalety:

- Nie wymaga określenia przez użytkownika liczby skupień.
- Pozwala znaleźć dowolne kształty skupień.
- Pozwala zidentyfikować obserwacje odstające.

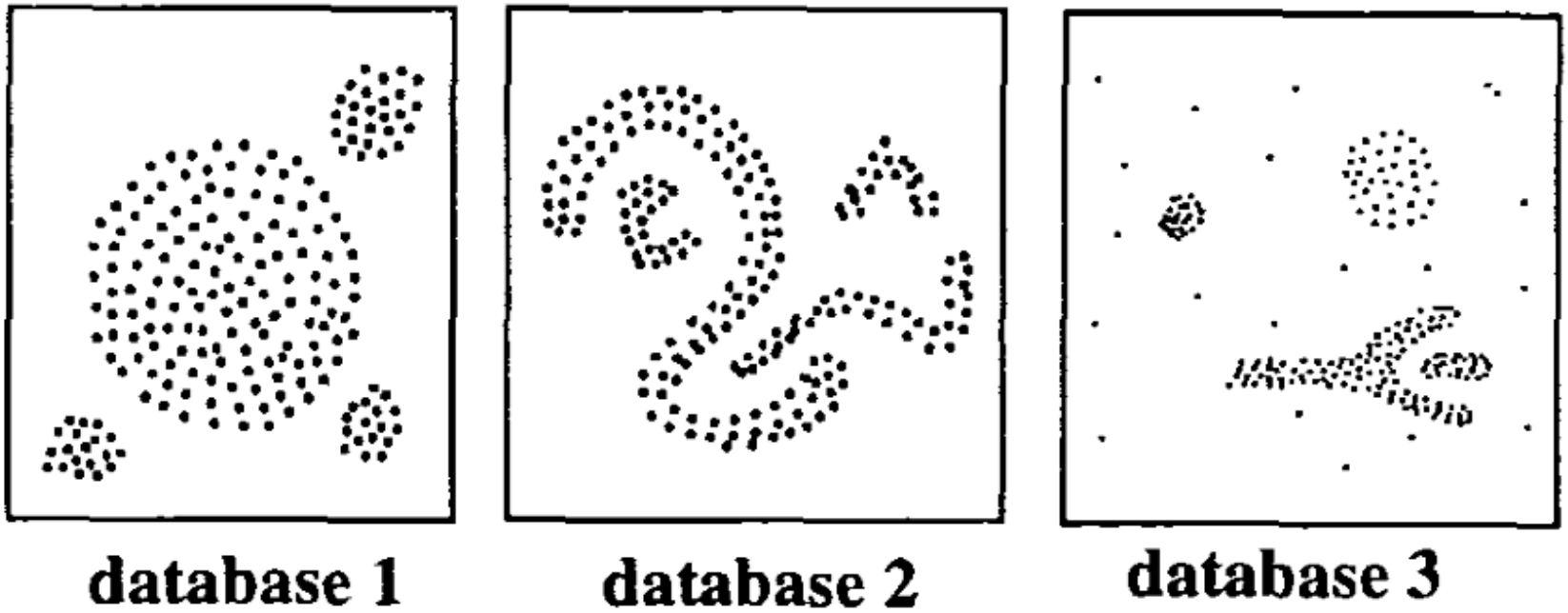
Literatura

M. Ester, H.-P. Kriegel, J. Sander, X. Xu (1996). *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Institute for Computer Science, University of Munich*. Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96).

R.F. Ling (1972). *On the theory and construction of k-clusters*. The Computer Journal 15(4):326–332.

DBSCAN – analiza skupień bazująca na gęstościach punktów

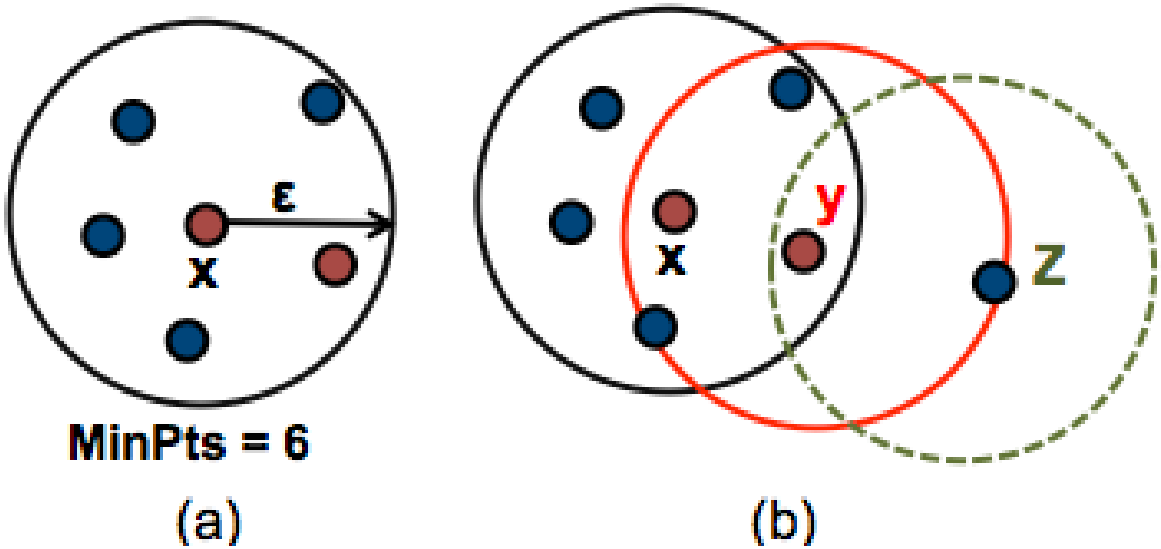
Główna idea wywodzi się z ludzkiej intuicji. Na przykład na poniższym obrazku widać, że mamy (bazując na gęstości punktów) cztery skupienia oraz kilka punktów odstających.



DBSCAN – analiza skupień bazująca na gęstościach punktów

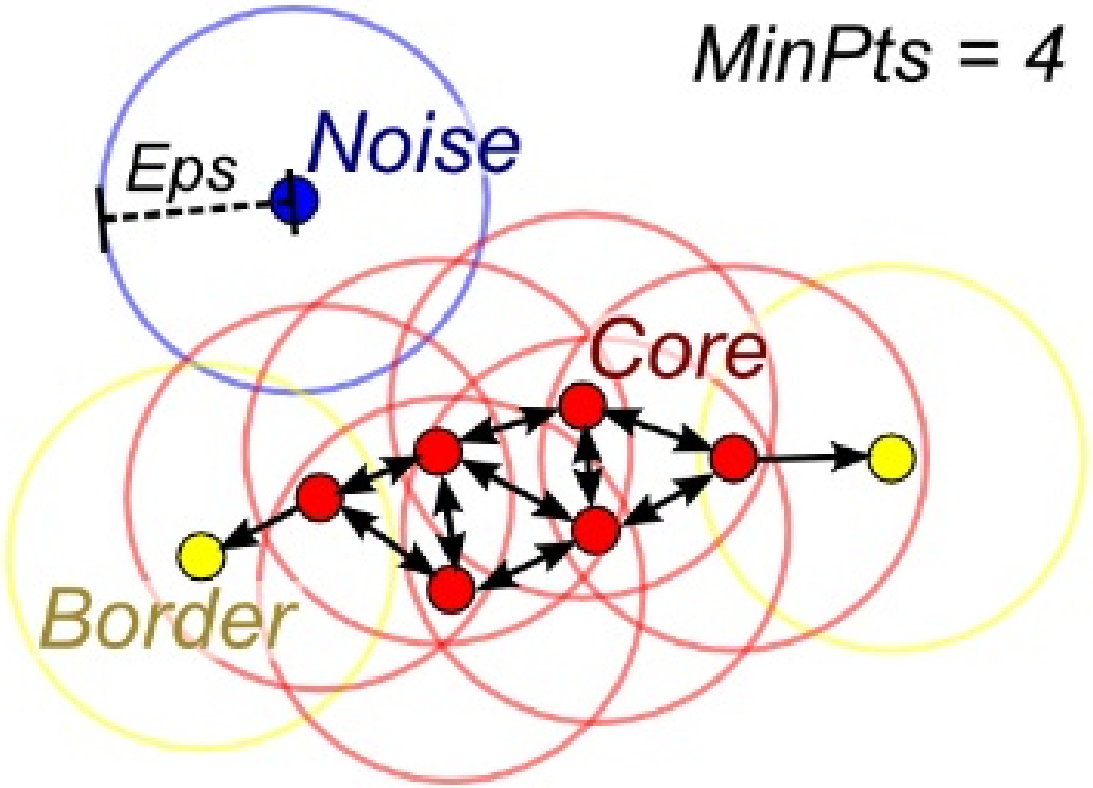
Głównym celem jest identyfikacja gęstych regionów, które mogą być widziane jako regiony z dużą liczbą punktów. Istotne są dwa parametry: „epsilon" (eps) oraz „minimum points" (MinPts). Pierwszy z nich określa promień sąsiedztwa punktu $\boldsymbol{x}$. Drugi definiuje minimalną liczbę punktów sąsiedztwa w promieniu epsilon. Każdy punkt zbioru danych z liczbą sąsiadów większą bądź równą MinPts jest nazywany punktem rdzeniowym (ang. core point). Punkt jest nazywany punktem granicznym (ang. border point) jeśli liczba jego sąsiadów jest mniejsza od MinPts, ale należy on do sąsiedztwa pewnego punktu rdzeniowego. Jeśli punkt nie jest rdzeniowym ani granicznym to nazwany jest punktem szumu lub odstającym (ang. noise point).

DBSCAN – analiza skupień bazująca na gęstościach punktów



Niech MinPts = 6. Punkt ***x*** jest tutaj punktem rdzeniowym ponieważ w jego otoczeniu o promieniu ϵ znajduje się 6 punktów. ***y*** jest punktem granicznym ponieważ w jego otoczeniu jest mniej niż 6 punktów (dokładnie 5), ale należy on do otoczenia punktu rdzeniowego ***x***. Natomiast punkt ***z*** jest punktem odstającym (ma dokładnie 2 sąsiadów, którzy nie są punktami rdzeniowymi).

DBSCAN – analiza skupień bazująca na gęstościach punktów



Porównanie metod analizy skupień