Idea

Analiza skupień 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

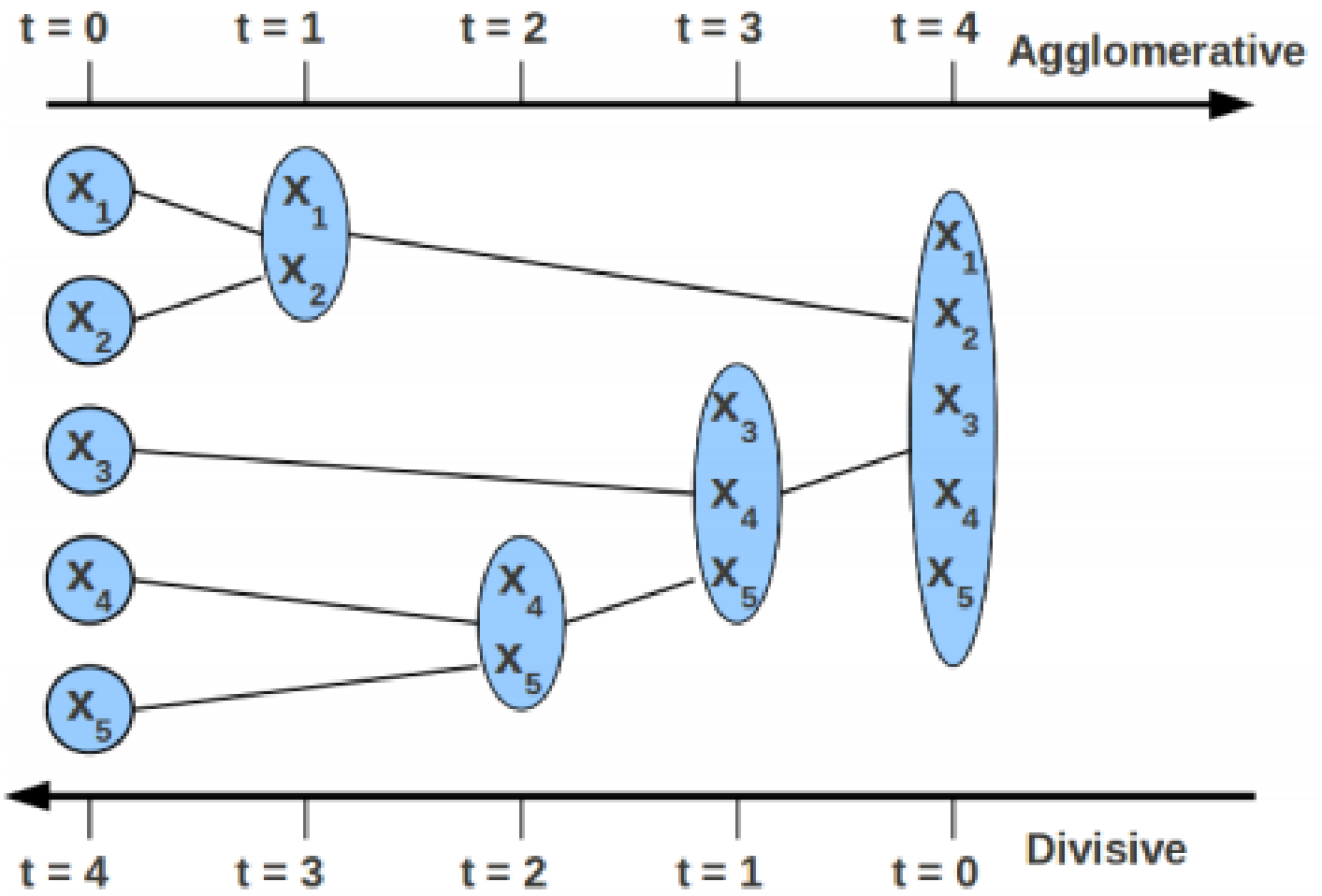
$$\frac{1}{K!}\sum_{k=1}^{K}(-1)^{K-k}\binom{K}{k}k^n$$

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 – 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, 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 trzy najczęściej wykorzystywane sposoby jej określenia.

Algorytmy hierarchiczne – metoda pojedynczego wiązania

Metoda pojedynczego wiązania (najbliższego sąsiedztwa). Miara niepodobieństwa pomiędzy dwoma skupieniami jest określona jako najmniejsza miara niepodobieństwa między dwoma obiektami należącymi do różnych skupień, tzn.

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

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). Miara niepodobieństwa pomiędzy dwoma skupieniami jest określona jako największa miara niepodobieństwa między dwoma obiektami należącymi do różnych skupień, tzn.

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

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

Algorytmy hierarchiczne – metoda pojedynczego wiązania

Metoda średniego wiązania. Miara niepodobieństwa pomiędzy dwoma skupieniami jest określona jako średnia miara niepodobieństwa między wszystkimi parami obiektów należących do różnych skupień, tzn.

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

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.

Algorytmy hierarchiczne – algorytm aglomeracyjny – podsumowanie

Algorytm aglomeracyjny jest bardzo szybki i 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.

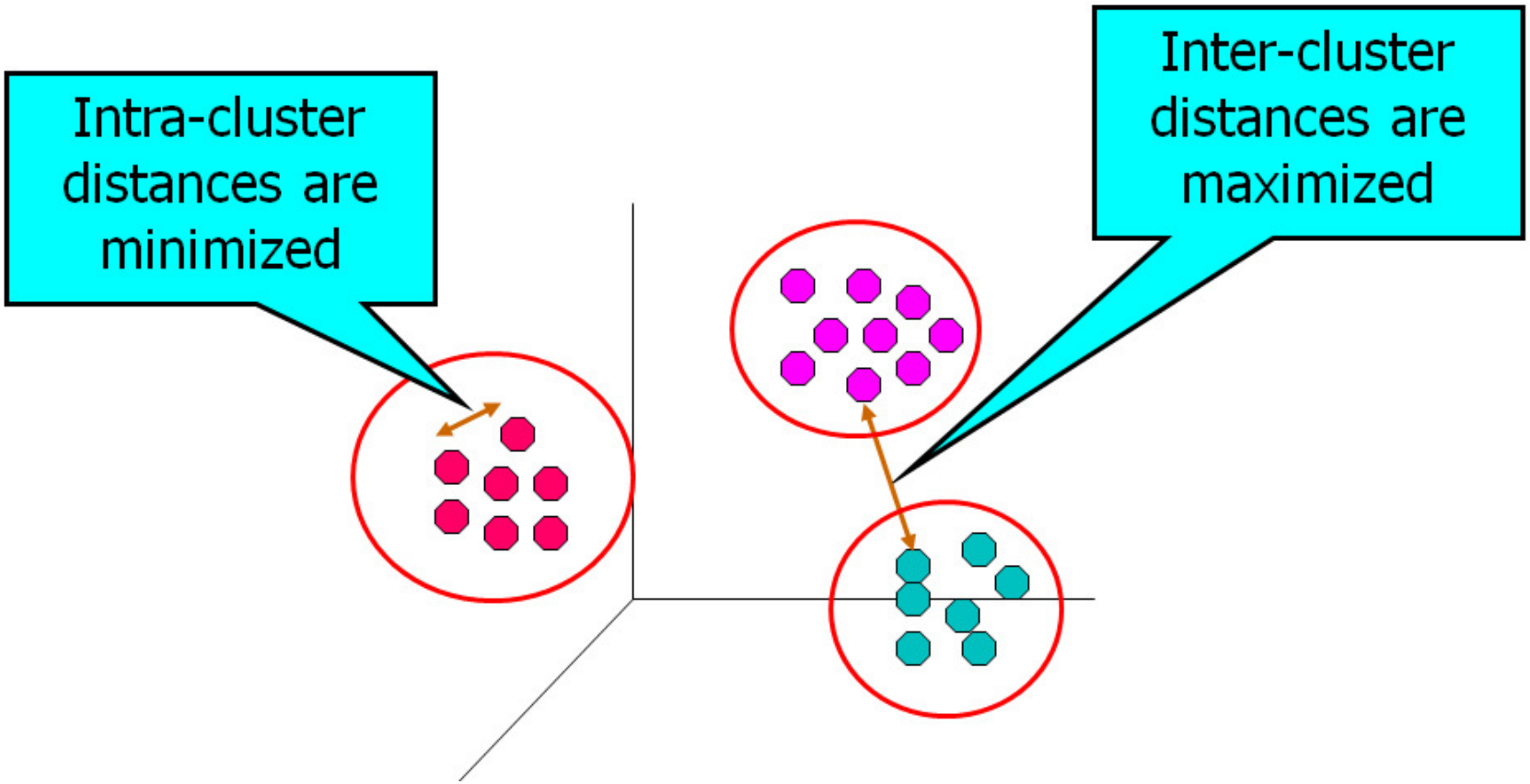
Metoda hierarchiczna w R

Służy do tego funkcja hclust, której pierwszym argumentem jest macierz odległości, a drugim metoda wiązania skupień. Funkcja rect.hclust pozwala automatycznie przyciąć dendryt, a funkcja cutree pozwala wydzielić zadaną z góry liczbę skupień.

Metoda K-średnich – idea

Najbardziej popularnym, niehierarchicznym algorytmem analizy skupień jest algorytm K-średnich. 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 – 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 $W_K = \log(\mathrm{tr}(W(C_K)))$. 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.

Metoda K-średnich – wybór K

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. Jeden z nich zasługują na szczególną uwagę.

Caliński i Harabasz (1974) zaproponowali aby końcową liczbę skupień wybierać w oparciu o wartości pseudo-statystyki F postaci:

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

Optymalną wartość K dobieramy tak, aby ją zmaksymalizować.

Metody niehierarchiczne w R

Metodę *k*-średnich realizuje funkcja kmeans, której pierwszym argumentem jest zbiór danych, a drugim liczba skupień. Do wizualizacji można użyć funkcji clusplot z pakiet `cluster`. W tym samym pakiecie znajdują się również funkcje pam oraz clara, które realizuję wydajniejsze metody poszukiwania skupień za pomocą metod niehierarchicznych. Pakiet `vegan` zawiera funkcję cascadeKM, która bazując na indeksie CH wyznacza optymalną liczbę skupień.

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 początkowych środków ciężkości. W praktyce często metoda hierarchiczna służy do wstępnej obróbki danych i wyznaczenia punktów startowych dla metody k-średnich (np. jako średnie w skupieniach). Analiza skupień nie jest odporna na zmiany skali, oznacza to, że jeśli różne zmienne mają różne skale, to te największe mogą zdominować odległości. Oczywiście może to być celowe, w takim przypadku pewne zmienne są „ważniejsze" od innych. W ogólności jednak warto wykonać wpierw skalowanie danych.

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.

DBSCAN – analiza skupień bazująca na gęstościach punktów

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

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

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).

DBSCAN – analiza skupień bazująca na gęstościach punktów

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