Zaawansowane uczenie maszynowe:
wykład 12

Paweł Cichosz

Zadanie grupowania

Dziedzina, przykład, atrybut, zbiór trenujący: jak dla klasyfikacji i regresji, ale bez wyróżnionego atrybutu docelowego.

Model: $h : X \to C_h$, gdzie C_h oznacza zbiór grup dla modelu grupowania h – nie tylko podział zbioru trenującego na grupy, lecz także możliwość przypisania do tych grup dowolnych przykładów z dziedziny (chociaż nie zawsze ta możliwość jest oferowana przez implementacje).

Grupy a podobieństwo: maksymalizacja podobieństwa wewnątrzgrupowego, minimalizacja podobieństwa międzygrupowego.

Zadanie grupowania

Scenariusze użycia:

- odkrycie wiedzy o wzorcach podobieństwa w dziedzinie,
- grupowanie na podstawie atrybutów obserwowalnych (dostępnych zawsze) w celu wnioskowania o atrybutach ukrytych (dostępnych rzadko),
- nienadzorowana detekcja anomalii jako przykładów niepodobnych do żadnej grupy,
- dekompozycja dziedziny lub wybór reprezentatywnych przykładów na potrzeby innych zadań modelowania.

Typy algorytmów grupowania

Na podstawie niepodobieństwa: wykorzystujące jawnie zadaną miarę niepodobieństwa do ustalenia przynależności przykładów do grup.

Na podstawie gęstości: wykorzystujące gęstość przykładów w przestrzeni atrybutów do wyznaczenia podziału na grupy.

Na podstawie rozkładu: dopasowujące mieszaninę rozkładów prawdopodobieństwa do danych (nie będą omawiane).

Miary niepodobieństwa

Odległość euklidesowa:

$$\delta_{\text{euc}}(x_1, x_2) = \sqrt{\sum_{i=1}^{n} (a_i(x_1) - a_i(x_2))^2}$$

Odległość Minkowskiego:

$$\delta_{\text{mink}}(x_1, x_2) = \left(\sum_{i=1}^{n} |a_i(x_1) - a_i(x_2)|^p \right)^{\frac{1}{p}}$$

Odległość miejska (Manhattan): odległość Minkowskiego dla $p = 1$:

$$\delta_{\text{man}}(x_1, x_2) = \sum_{i=1}^{n} |a_i(x_1) - a_i(x_2)|$$

Odległość Czebyszewa (maksimum): odległość Minkowskiego dla $p \to \infty$:

$$\delta_{\text{max}}(x_1, x_2) = \max_i |a_i(x_1) - a_i(x_2)|$$

Miary niepodobieństwa

Odległość Mahalanobisa: rozszerzony wariant odległości euklidesowej, kompensujący zróżnicowaną wariancję atrybutów oraz korelacje między atrybutami:

$$\delta_{\mathrm{mah}}(x_1, x_2) = \sqrt{\big(\mathbf{a}(x_1) - \mathbf{a}(x_2)\big)^{\mathsf{T}} \mathrm{Cov}_T^{-1}\big(\mathbf{a}\big)\big(\mathbf{a}(x_1) - \mathbf{a}(x_2)\big)}$$

gdzie:

- $\mathbf{a}(x)$ – (kolumnowy) wektor różnic wartości atrybutów $a_1, a_2, \ldots, a_n$ dla przykładu x,
- $\mathrm{Cov}_T^{-1}\big(\mathbf{a}\big)$ oznacza odwróconą macierz kowariancji atrybutów $a_1, a_2, \ldots, a_n$ na zbiorze trenującym T.

Miary niepodobieństwa

Odległość Hamminga:

$$\delta_{\text{ham}}(x_1, x_2) = \sum_{i=1}^{n} \mathbb{I}_{a_i(x_1) \neq a_i(x_2)}$$

gdzie

$$\mathbb{I}_{a_i(x_1) \neq a_i(x_2)} = \begin{cases} 1 & \text{jeśli } a_i(x_1) \neq a_i(x_2) \\ 0 & \text{w przeciwnym przypadku} \end{cases}$$

Niepodobieństwo kosinusowe:

$$\delta_{\cos}(x_1, x_2) = 1 - \frac{\mathbf{a}(x_1) \circ \mathbf{a}(x_2)}{\|\mathbf{a}(x_1)\| \cdot \|\mathbf{a}(x_1)\|}$$

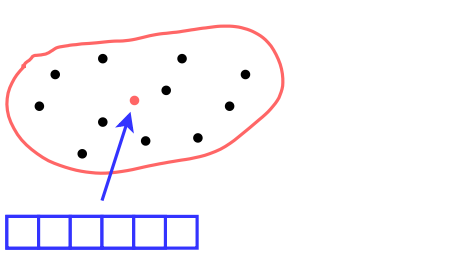
Miary niepodobieństwa

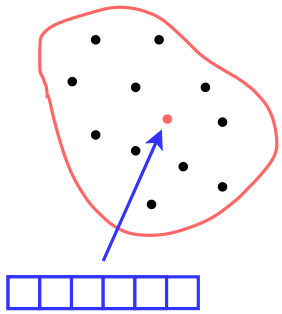
Transformacja atrybutów: w przypadku użycia miar opartych na różnicach wartości atrybutów zazwyczaj wskazane skalowanie (standardyzacja lub normalizacja).

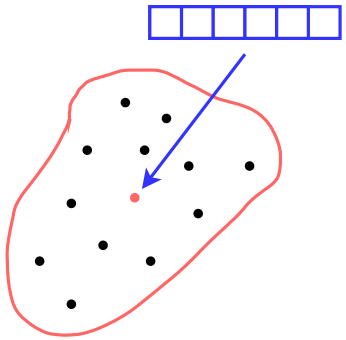
Ważenie atrybutów: uwzględnianie różnicy wartości $a_i(x_1) - a_i(x_2)$ z wagą $\eta_i \geq 0$.

Uwzględnianie atrybutów dyskretnych: np. zastąpienie różnicy wartości $a_i(x_1) - a_i(x_2)$ przez binarny indykator $\mathbb{I}_{a_i(x_1) \neq a_i(x_2)}$ (jak w odległości Hamminga) przy jednoczesnym skalowaniu różnic dla atrybutów ciągłych do przedziału $[0, 1]$.

Reprezentacja grupowania







Liczba grup: k – ustalona.

Reprezentacja grup: wektor środkowy ζ_d dla każdej grupy $d = 1, 2, \ldots, k$ – wektor wartości atrybutów używanych do grupowania.

Elementy grupy: zbiór przykładów trenujących T_d zaliczonych do grupy d dla $d = 1, 2, \ldots, k$ na podstawie niepodobieństwa od wektora środkowego ζ_d.

Schemat algorytmu

1. wybierz początkowe wektory środkowe $\zeta_1, \zeta_2, \ldots, \zeta_k$;
2. **powtarzaj**:
   1. **dla wszystkich** $x \in T$:

      przypisz x *do grupy* $d_x = \arg\min_d \delta(x, \zeta_d)$;
   2. **dla** $d = 1, 2, \ldots, k$:

      modyfikuj wektor środkowy ζ_d *na podstawie zbioru przykładów* T_d;

jak długo nie są spełnione kryteria stopu.

Schemat algorytmu

Inicjalizacja wektorów środkowych: np. losowy wybór k przykładów trenujących.

Modyfikacja wektorów środkowych: różne warianty.

Kryteria stopu:

- zbieżność (brak zmiany przynależności przykładów do grup),
- przybliżona zbieżność (mała liczba przykładów zmieniających przynależność do grup),
- zadana liczba iteracji.

Dobór k: na podstawie oceny jakości grupowania.

Warianty

k-średnich:

wektory środkowe: wektory *średnich* wartości atrybutów dla przykładów z poszczególnych grup,

właściwości: wrażliwość na wartości odstające, gwarancja zbieżności z odległością euklidesową, lokalna minimalizacja sumy kwadratów odległości euklidesowych przykładów od odpowiednich środków grup.

k-median:

wektory środkowe: wektory *median* wartości atrybutów dla przykładów z poszczególnych grup,

właściwości: większa odporność na wartości odstające, gwarancja zbieżności z odległością miejską, lokalna minimalizacja sumy odległości miejskich przykładów od odpowiednich środków grup.

Warianty

k-medoid:

wektory środkowe: faktyczne przykłady trenujące minimalizujące sumę niepodobieństw od przykładów z poszczególnych grup,

popularna realizacja: PAM (*partitioning around medoids*),

właściwości: największa odporność na wartości odstające i zależności między atrybutami, gwarancja zbieżności z dowolną miarą niepodobieństwa, lokalna minimalizacja sumy niepodobieństw przykładów od odpowiednich środków grup.

DBSCAN

DBSCAN: *Density-Based Spatial Clustering of Applications with Noise*.

Liczba grup: dobierana automatycznie.

Reprezentacja grupowania: przypisanie przykładów trenujących do grup lub oznaczenie ich jako „szumu” (przykładów odstających).

Zasada działania: grupy powstają wokół przykładów o wystarczająco gęstym ϵ-sąsiedztwie:

$$T_{x,\epsilon} = \{x' \in T \mid \delta(x', x) \leq \epsilon\}$$

Schemat algorytmu

1. oznacz wszystkie przykłady jako *nieodwiedzone*;
2. **powtarzaj:**
   1. wybierz *nieodwiedzony* przykład x_s;
   2. oznacz x_s jako *odwiedzony*;
   3. **jeżeli** $|T_{x_s,\epsilon}| < m$:

      oznacz x_s jako szum;

      w przeciwnym przypadku:

      1. utwórz nową grupę d_{x_s};
      2. $T_{d_{x_s}} := \{x_s\}$;
      3. *rozszerzenie*$(d_{x_s}, T_{x_s,\epsilon})$;

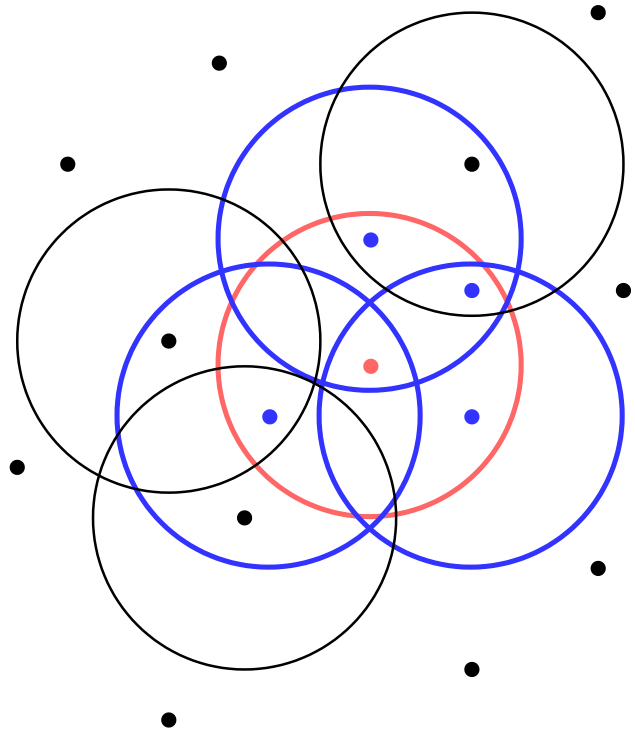
   dopóki wszystkie przykłady nie zostaną oznaczone jako *odwiedzone*.

Schemat algorytmu

rozszerzenie(d, S):

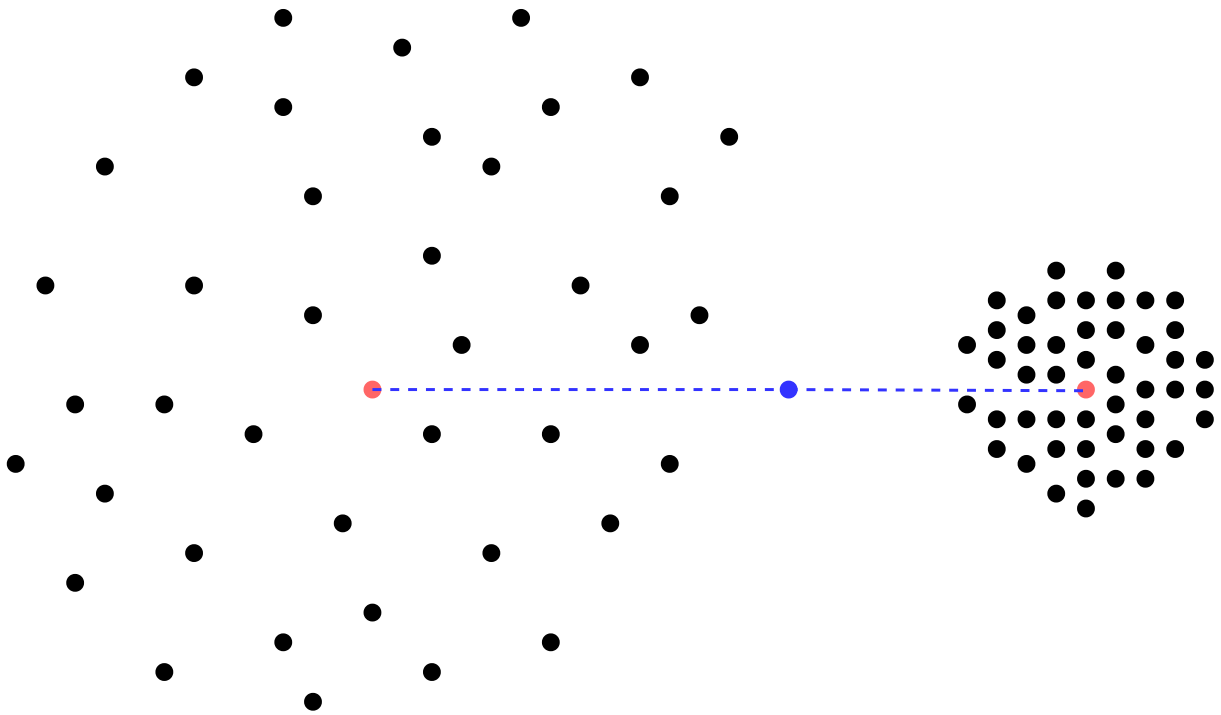
dla wszystkich $x \in S$*:*

1. *oznacz* x *jako odwiedzony;*
2. $T_d := T_d \cup \{x\}$*;*
3. *rozszerzenie*$(d, T_{x,\epsilon})$*.*



Właściwości

- Możliwość identyfikacji grup o dowolnym „kształcie”, niekoniecznie sferycznych jak w przypadku grupowania k-środków.
- Wykrywanie przykładów odstających.
- Brak optymalizacji jawnie określonej miary jakości grupowania.
- Dobór parametrów ϵ, m trudniejszy niż dobór k.



Ocena wewnętrzna

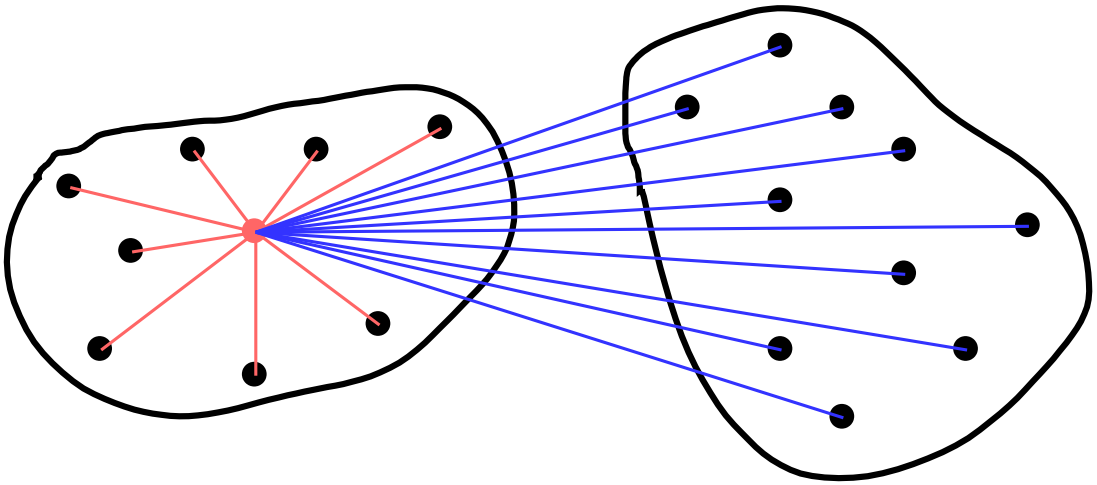
Wiele wskaźników jakości opartych na podobieństwie wewnątrzgrupowym i zróżnicowaniu międzygrupowym.

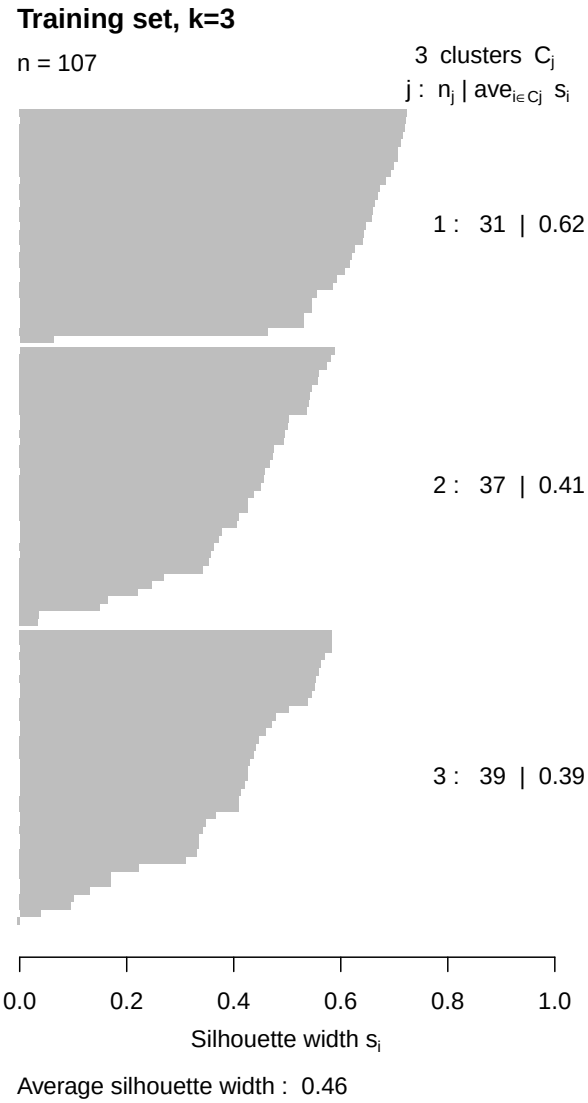
Szerokość sylwetki: miara dopasowania przykładu do jego grupy, której średnia wartość może być miarą jakości grupowania:

$$\mathrm{sw}(x) = \frac{\min_d \Delta(x,d) - \Delta(x)}{\max\{\min_d \Delta(x,d), \Delta(x)\}}$$

gdzie $\Delta(x,d)$ – średnie niepodobieństwo x do przykładów z innej grupy d, $\Delta(x)$ – średnie niepodobieństwo x do przykładów z jego własnej grupy.

Ocena wewnętrzna





Ocena zewnętrzna

Ocena zgodności z zewnętrznym etykietowaniem.

Indeks Randa:

$$\mathrm{rand}_c(h) = \frac{\mathrm{TP}_c(h) + \mathrm{TN}_c(h)}{\mathrm{TP}_c(h) + \mathrm{TN}_c(h) + \mathrm{FP}_c(h) + \mathrm{FN}_c(h)}$$

gdzie

$$\begin{aligned}
\mathrm{TP}_c(h) &= \left|\{\langle x_1, x_2\rangle \mid c(x_1) = c(x_2) \wedge h(x_1) = h(x_2)\}\right| \\
\mathrm{TN}_c(h) &= \left|\{\langle x_1, x_2\rangle \mid c(x_1) \neq c(x_2) \wedge h(x_1) \neq h(x_2)\}\right| \\
\mathrm{FP}_c(h) &= \left|\{\langle x_1, x_2\rangle \mid c(x_1) \neq c(x_2) \wedge h(x_1) = h(x_2)\}\right| \\
\mathrm{FN}_c(h) &= \left|\{\langle x_1, x_2\rangle \mid c(x_1) = c(x_2) \wedge h(x_1) \neq h(x_2)\}\right|
\end{aligned}$$