Gęstościowe grupowanie danych i wyznaczanie najbliższego sąsiedztwa z użyciem nierówności trójkąta

inż. Bartłomiej Jańczak

B.Janczak@stud.elka.pw.edu.pl

Politechnika Warszawska

27-06-2013

Plan prezentacji

1. Cele pracy
2. Zarys teorii
3. Stworzone oprogramowanie
4. Wybrane wyniki eksperymentalne
5. Podsumowanie

Cele pracy

- Zbadanie możliwości wydajnego grupowania gęstościowego i wyznaczania k sąsiedztwa z zastosowaniem:
  - nierówności trójkąta,
  - rzutowania,
  - VP-Tree

  dla miary odległości euklidesowej i miary podobieństwa kosinusowego.

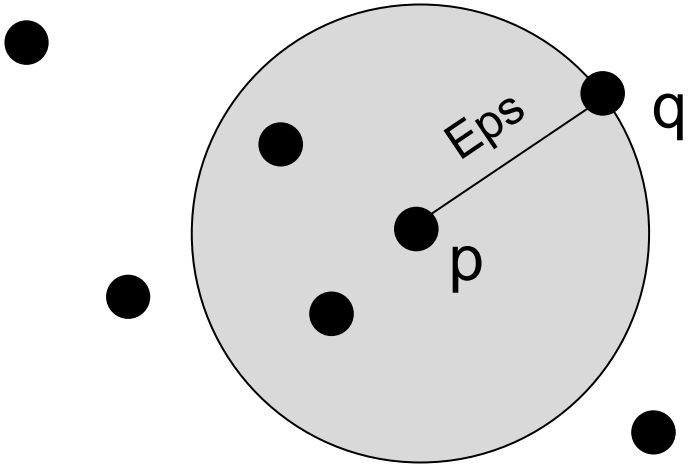
- Implementacja algorytmów uwzględniających optymalizacje.

- Eksperymentalna weryfikacja algorytmów na zbiorach danych o różnej charakterystyce.

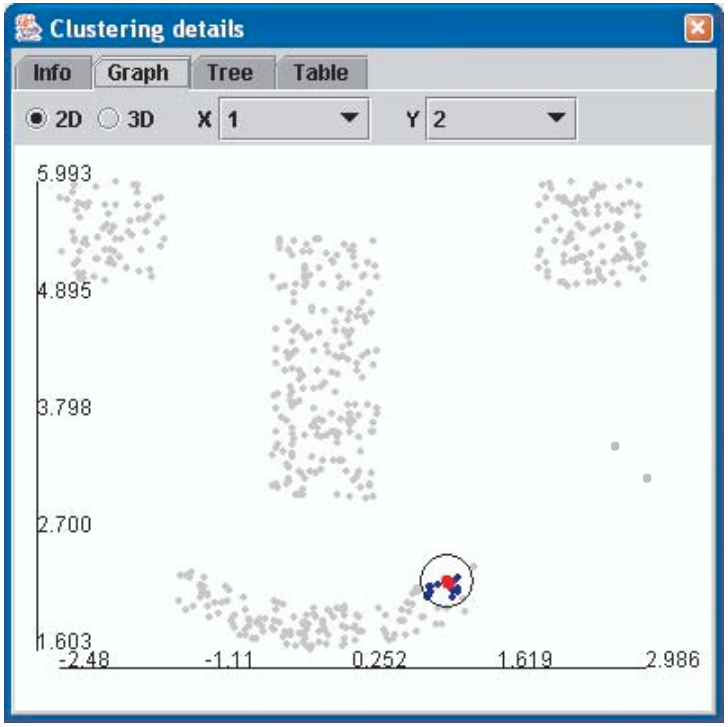
DBSCAN: Density-Based Clustering Algorithm with Noise

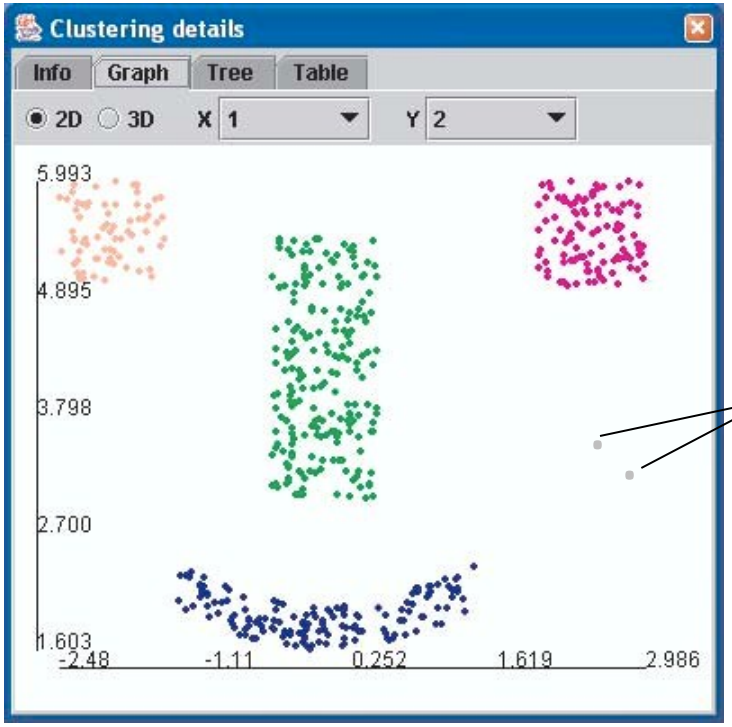
- otoczenie epsilonowe:
  $N_{Eps}(p) = \{q \in D | distance(p,q) \leq Eps\}$.
- punkt rdzeniowy:

$$\left|N_{Eps}(p)\right| \geq MinPts.$$



DBSCAN w akcji





szum

Wykorzystanie nierówności trójkąta

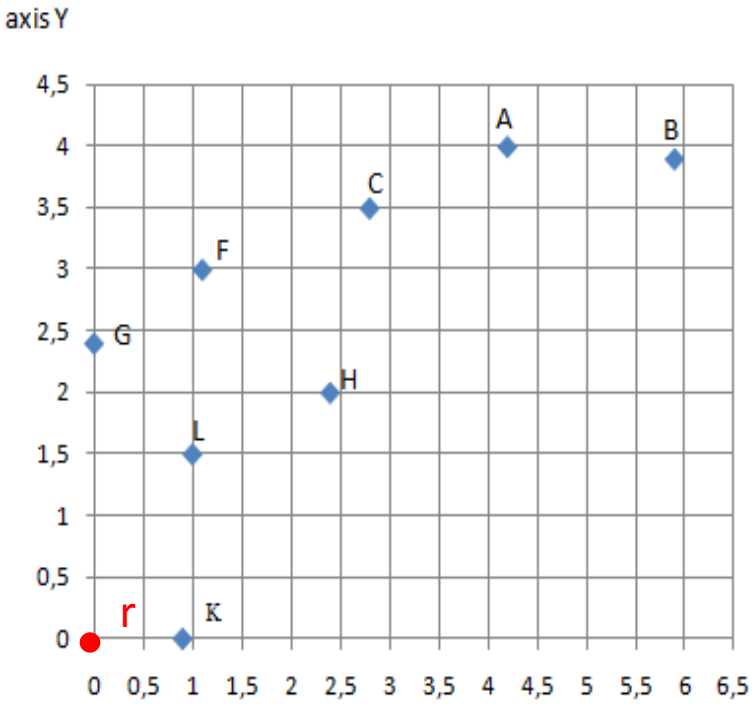
Dla dowolnych punktów p, q i r:

$$distance(p,q) + distance(q,r) \geq distance(p,r),$$

$$distance(p,q) \geq distance(p,r) - distance(q,r) = distance^r(p,q),$$

$$distance(p,q) \geq \underbrace{distance^r(p,q)}_{\text{pesymistyczne oszacowanie}} > Eps.$$

Wykorzystanie nierówności trójkąta - przykład



Uporządkowany zbiór D; Eps = 0,2; r=(0,0)

q∈D	X	Y	distance(q,r)
K	0,9	0,0	0,9
L	1,0	1,5	1,8
G	0,0	2,4	2,4
H	2,4	2,0	3,1
F	**1,1**	**3,0**	**3,2**
C	2,8	3,5	4,5
A	4,2	4,0	5,8
B	5,9	3,9	7,1

$\notin N_{Eps}(F)$

Wykorzystanie nierówności trójkąta - rzutowanie

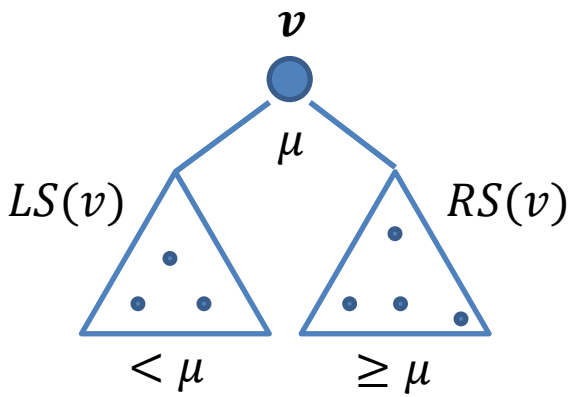
Dla każdego wymiaru $l, l \in [1, \dots, n]$ i punktów p i q:

$$|p_l - q_l| = \sqrt{(p_l - q_l)^2} \leq \sqrt{\sum_{i=1..n} (p_l - p_q)^2} = Euclidean(p, q)$$

$$|p_l - q_l| > Eps \Rightarrow Euclidean(p, q) > Eps \Rightarrow p \notin N_{Eps}(q) \wedge q \notin N_{Eps}(p)$$

Wykorzystanie indeksu metrycznego VP-Tree

- **Węzeł VP-Tree zawiera**:
  - $v \in D$,
  - $\mu = mediana(\{u \in S(v) | distance(u, v)\})$
  - $LS(v) = \{u \in S(v) \backslash \{v\} | distance(u, v) < \mu\}$
  - $RS(v) = \{u \in S(v) \backslash \{v\} | distance(u, v) \geq \mu\}$

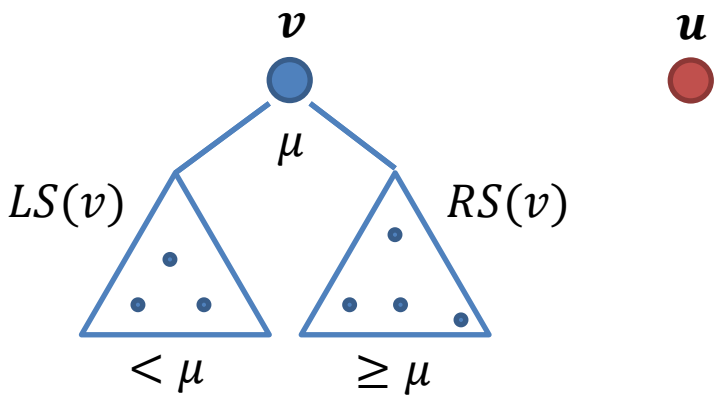


- **Wyszukiwanie sąsiedztwa punktu u o promieniu ε w węźle v VP-Tree**:
  - $distance(u, v)$,
  - W1: Jeśli $\mathrm{distance(u, v)} - \mu \geq \varepsilon$, to $LS(v)$ nie zawiera sąsiedztwa punktu u o promieniu ε
  - W2: Jeśli $\mathrm{distance(u, v)} - \mu < \varepsilon$, to R$S(v)$ nie zawiera sąsiedztwa punktu u o promieniu ε

- **Ulepszenie wyszukiwanie sąsiedztwa punktu u o promieniu ε w węźle v VP-Tree**:
  - W1': $\mathrm{distance(u, v)} - left\_boundary \geq \varepsilon$,
  - W2': Jeśli $\mathrm{distance(u, v)} - right\_boundary < \varepsilon$,

Wykorzystanie indeksu metrycznego VP-Tree

- **Węzeł VP-Tree zawiera**:
  - $v \in D$,
  - $\mu = mediana(\{u \in S(v) | distance(u, v)\})$
  - $LS(v) = \{u \in S(v) \backslash \{v\} | distance(u, v) < \mu\}$
  - $RS(v) = \{u \in S(v) \backslash \{v\} | distance(u, v) \geq \mu\}$

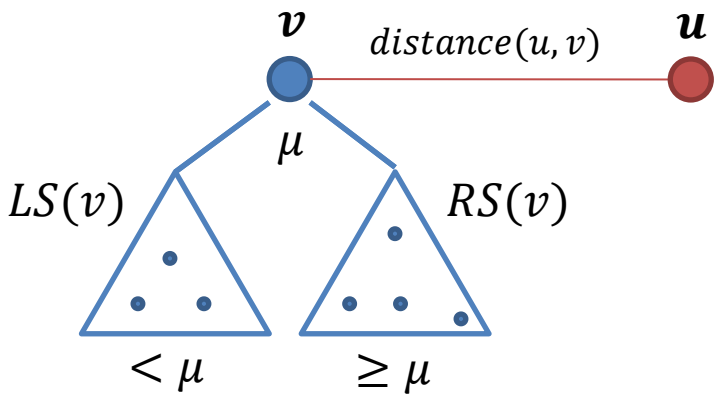


- **Wyszukiwanie sąsiedztwa punktu u o promieniu ε w węźle v VP-Tree**:
  - $distance(u, v)$,
  - W1: Jeśli $\mathrm{distance(u, v)} - \mu \geq \varepsilon$, to $LS(v)$ nie zawiera sąsiedztwa punktu u o promieniu ε
  - W2: Jeśli $\mathrm{distance(u, v)} - \mu < \varepsilon$, to R$S(v)$ nie zawiera sąsiedztwa punktu u o promieniu ε

- **Ulepszenie wyszukiwanie sąsiedztwa punktu u o promieniu ε w węźle v VP-Tree**:
  - W1': $\mathrm{distance(u, v)} - left\_boundary \geq \varepsilon$,
  - W2': Jeśli $\mathrm{distance(u, v)} - right\_boundary < \varepsilon$,

Wykorzystanie indeksu metrycznego VP-Tree

- **Węzeł VP-Tree zawiera**:
  - $v \in D$,
  - $\mu = mediana(\{u \in S(v) | distance(u, v)\})$
  - $LS(v) = \{u \in S(v) \backslash \{v\} | distance(u, v) < \mu\}$
  - $RS(v) = \{u \in S(v) \backslash \{v\} | distance(u, v) \geq \mu\}$



- **Wyszukiwanie sąsiedztwa punktu u o promieniu ε w węźle v VP-Tree**:
  - $distance(u, v)$,
  - W1: Jeśli $\mathrm{distance(u, v)} - \mu \geq \varepsilon$, to $LS(v)$ nie zawiera sąsiedztwa punktu u o promieniu ε
  - W2: Jeśli $\mathrm{distance(u, v)} - \mu < \varepsilon$, to $RS(v)$ nie zawiera sąsiedztwa punktu u o promieniu ε

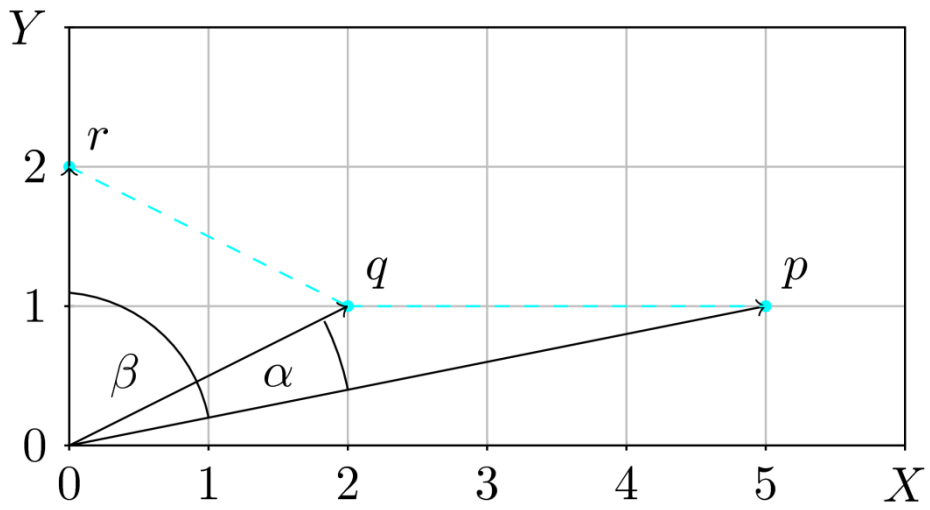
- **Ulepszenie wyszukiwanie sąsiedztwa punktu u o promieniu ε w węźle v VP-Tree**:
  - W1': $\mathrm{distance(u, v)} - left\_boundary \geq \varepsilon$,
  - W2': Jeśli $\mathrm{distance(u, v)} - right\_boundary < \varepsilon$,

Miary odległości i podobieństwa do określania sąsiedztwa

- Miara odległości euklidesowej:
  $Euclidean(p,q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$.

- Miara podobieństwa kosinusowego:
  $cosSim(p,q) = \frac{p \cdot q}{|p| \cdot |q|}$.

Miara podobieństwa kosinusowego nie spełnia nierówności trójkąta!

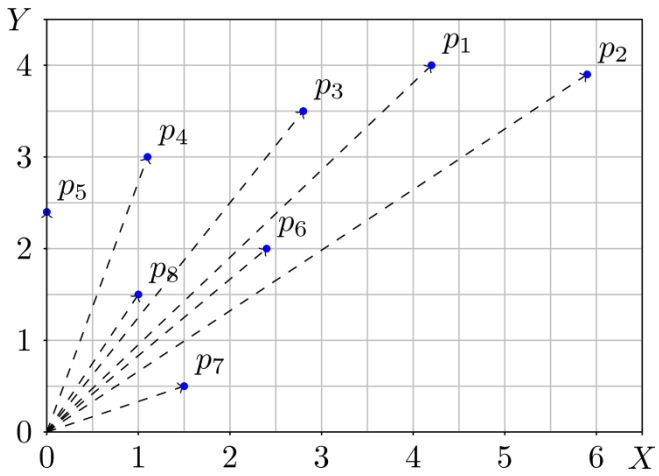


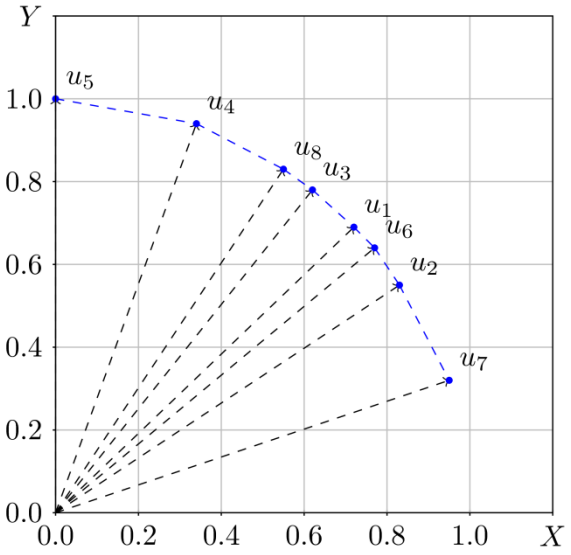
Miary odległości i podobieństwa do określania sąsiedztwa

$$cosSim(p,q) = cosSim\big(NF(p), NF(q)\big) = \frac{2 - Euclidean^2(NF(p), NF(q))}{2}$$

$$\downarrow$$

$$cosSim(p,q) \geq \varepsilon \Leftrightarrow Euclidean\big(NF(p), NF(q)\big) \leq \varepsilon' = \sqrt{2 - 2\varepsilon}$$





$$\varepsilon = 0{,}9659(15°) \quad \rightarrow \quad \varepsilon' = \sqrt{2 - 2\varepsilon} = 0{,}2611$$

Stworzone oprogramowanie

- Wsadowy tryb aplikacji
- Strojenie algorytmów poprzez pliki parametrów
- Generacja raportów wykonania algorytmów:
  - szczegółowych,
  - zbiorczych,

Wybrane wyniki eksperymentalne

Dane testowe

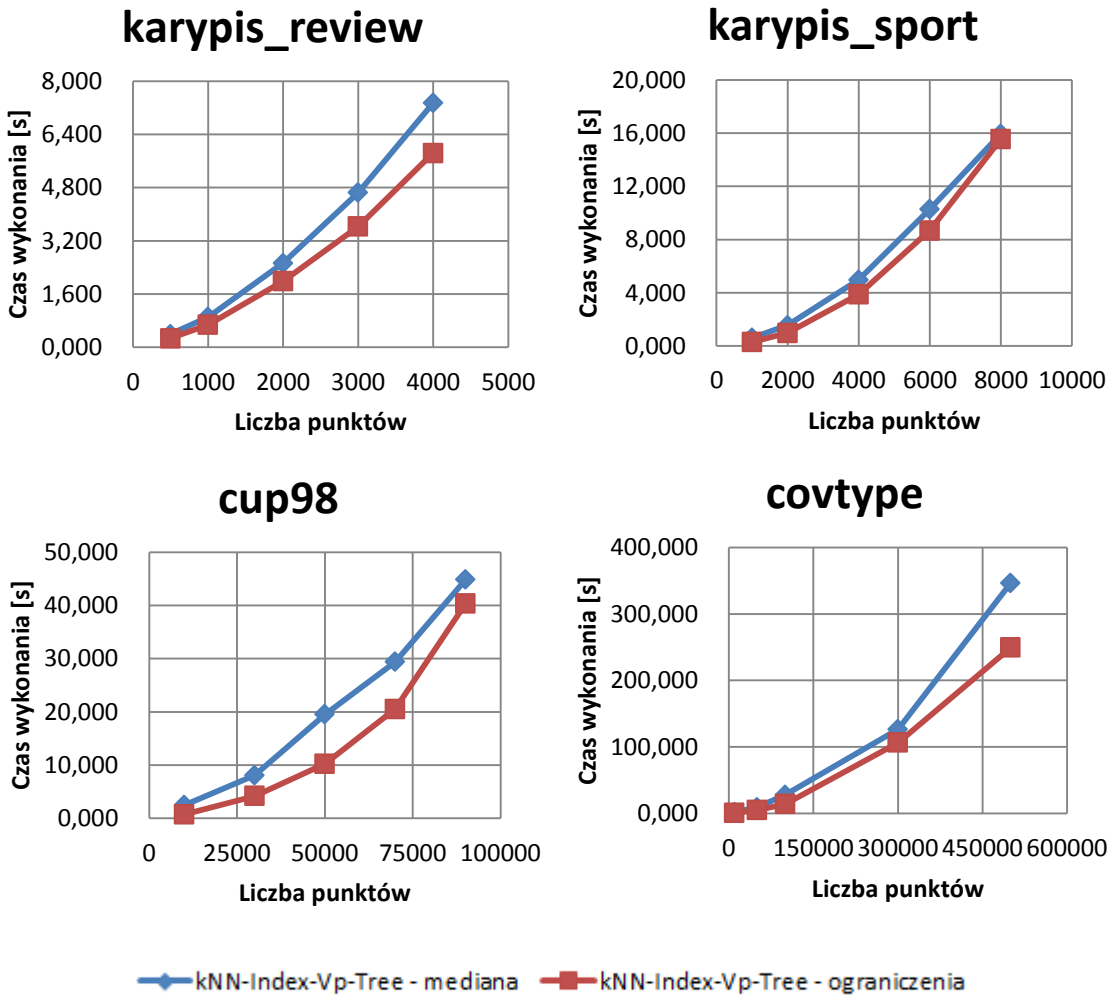
- **Zbiory danych powszechnie wykorzystywane w literaturze dziedzinowej**
- **Repozytorium danych tekstowych projektu CLUTO: http://glaros.dtc.umn.edu/gkhome/fetch/sw/cluto/datasets.tar.gz**

- **Zbiór *covtype***:
  - 581012 rekordów,
  - 55 atrybutów,
  - 44 atrybuty binarne,
  - gęsty.
- **Zbiór *cup98***:
  - 96367 rekordów,
  - 56 atrybutów,
  - gęsty.
- **Zbiór *karypis_sport***:
  - 8580 rekordów,
  - 126373 atrybutów,
  - średnio 129 atrybutów niezerowych,
  - rzadki.
- **Zbiór *karypis_review***:
  - 4069 rekordów,
  - 126373 atrybutów,
  - średnio 191 atrybutów niezerowych,
  - rzadki.

Wybrane wyniki eksperymentalne

Ulepszenie wyznaczania sąsiedztwa w VP-Tree – odległość euklidesowa

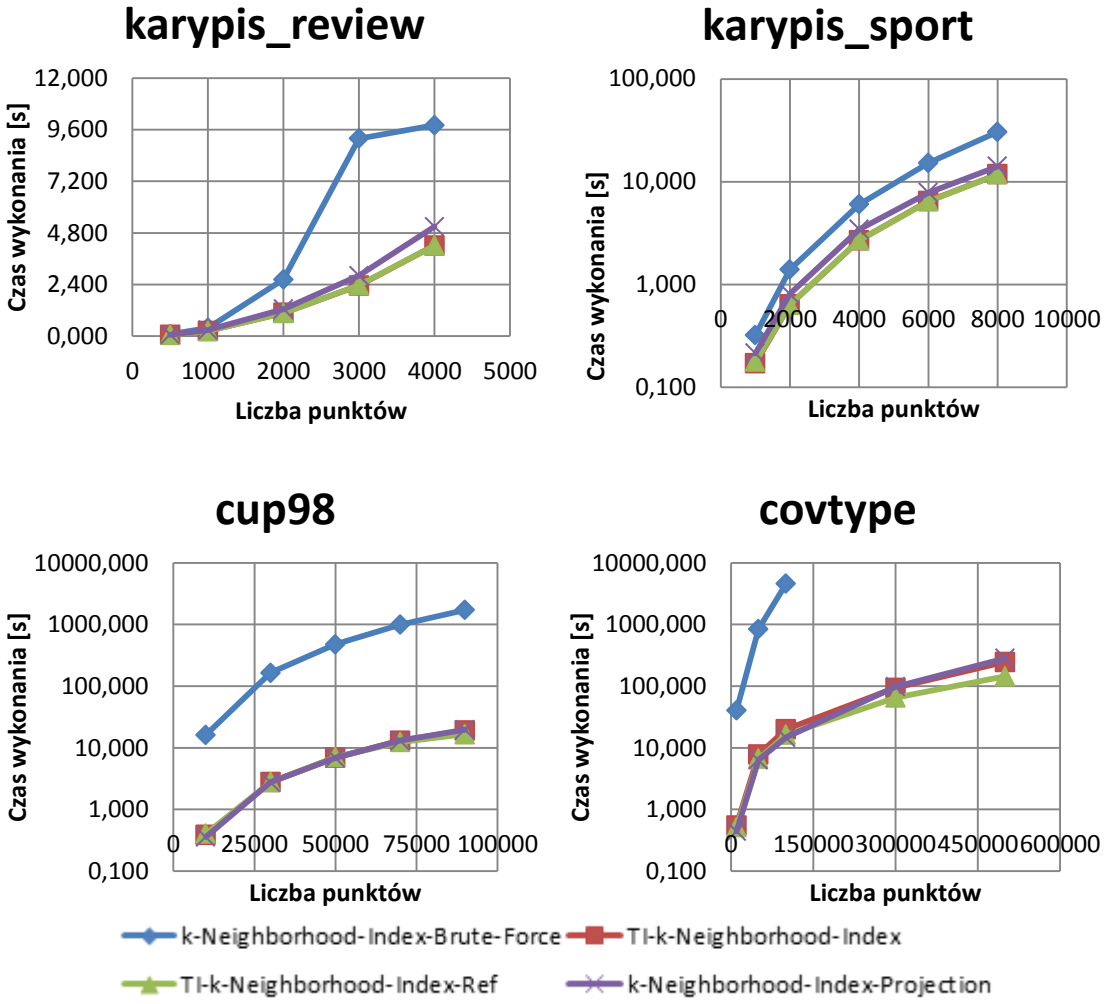
Porównanie wydajności algorytmu kNN-Index-Vp-Tree w zależności od implementacji metody przeszukiwania indeksu metrycznego przy zastosowaniu odległości euklidesowej jako miary podobieństwa. Wykresy zawierają czasy wykonania poszukiwań k=5 sąsiadów w przykładowych zbiorach dla 10% losowo wybranych punktów zbioru danych.



Wybrane wyniki eksperymentalne

Porównanie metod przyspieszania wyznaczania sąsiedztwa – odległość euklidesowa

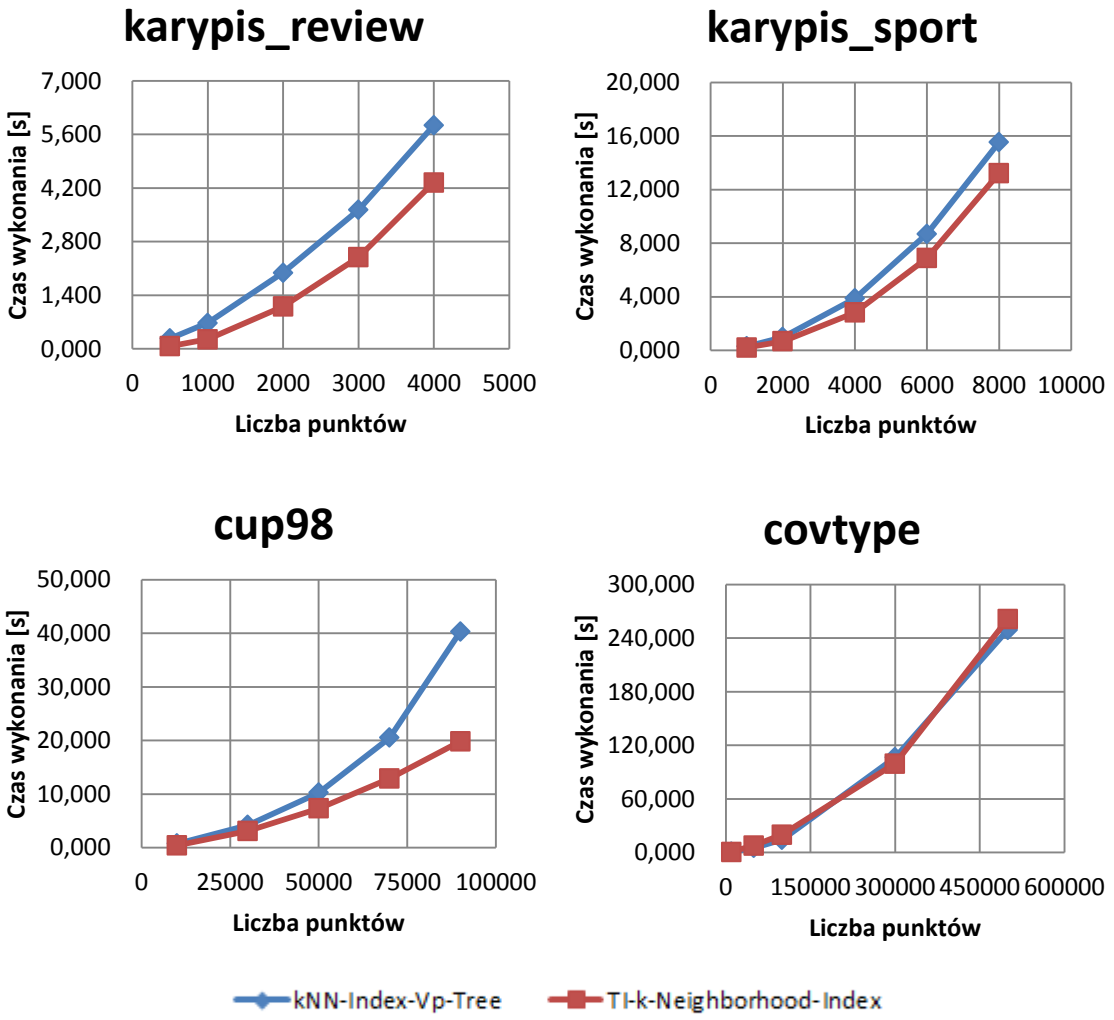
Porównanie wydajności odmian algorytmów k-Neighborhood-Index-Brute-Force, k-Neighborhood-Index-Projection, TI-k-Neighborhood-Index i TI-k-Neighborhood-Index-Ref przy zastosowaniu odległości euklidesowej jako miary podobieństwa. Wykresy zawierają czasy wykonania poszukiwań k=5 sąsiedztwa w przykładowych zbiorach danych dla 10% losowo wybranych punktów zbioru danych



Wybrane wyniki eksperymentalne

Porównanie metod przyspieszania wyznaczania sąsiedztwa – odległość euklidesowa

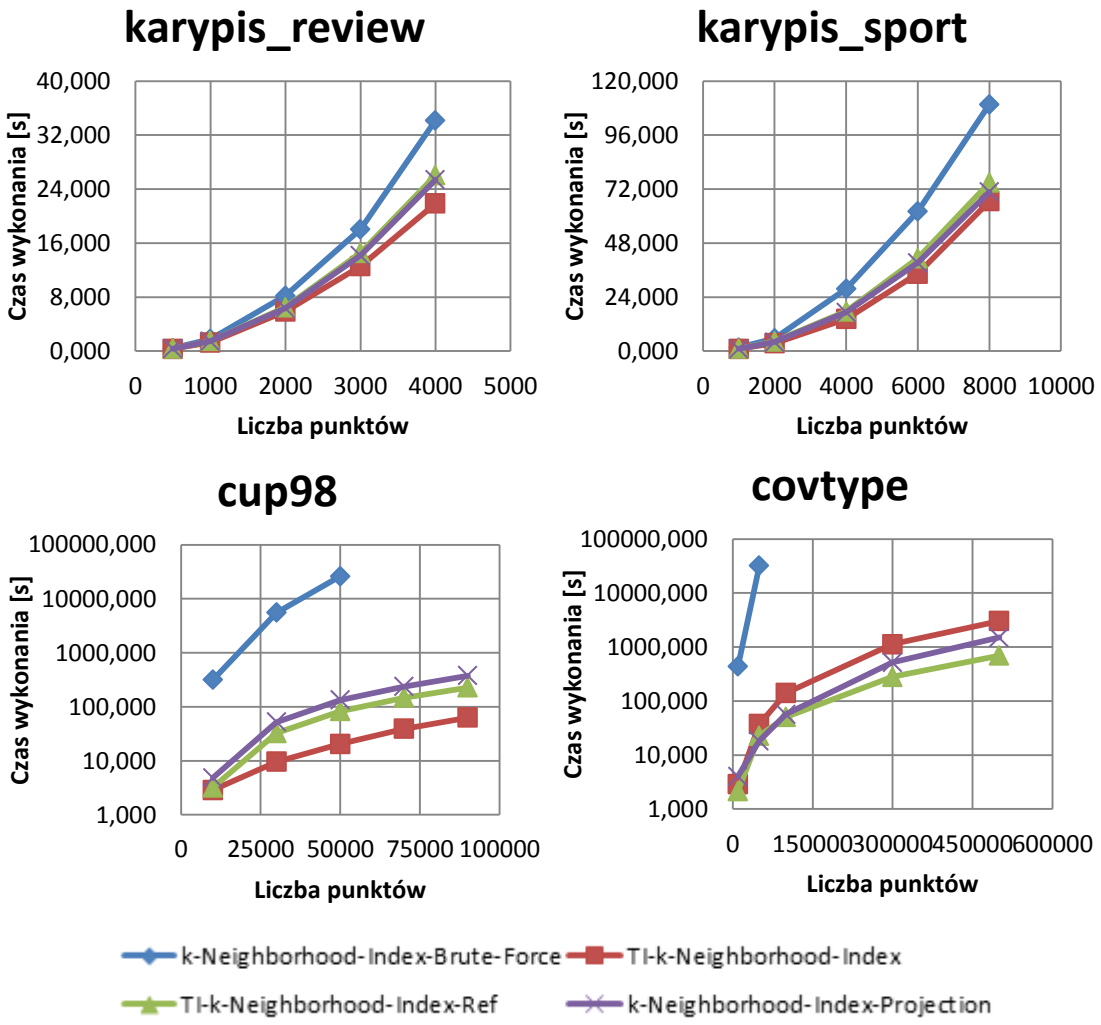
Porównanie wydajności algorytmów TI-k-Neighborhood-Index i kNN-Index-Vp-Tree przy zastosowaniu odległości euklidesowej jako miary podobieństwa. Wykresy zawierają czasy wykonania poszukiwań k=5 sąsiadów i k=5 sąsiedztwa w przykładowych zbiorach danych dla 10% losowo wybranych punktów zbioru danych



Wybrane wyniki eksperymentalne

Porównanie metod przyspieszania wyznaczania sąsiedztwa – miara kosinusowa

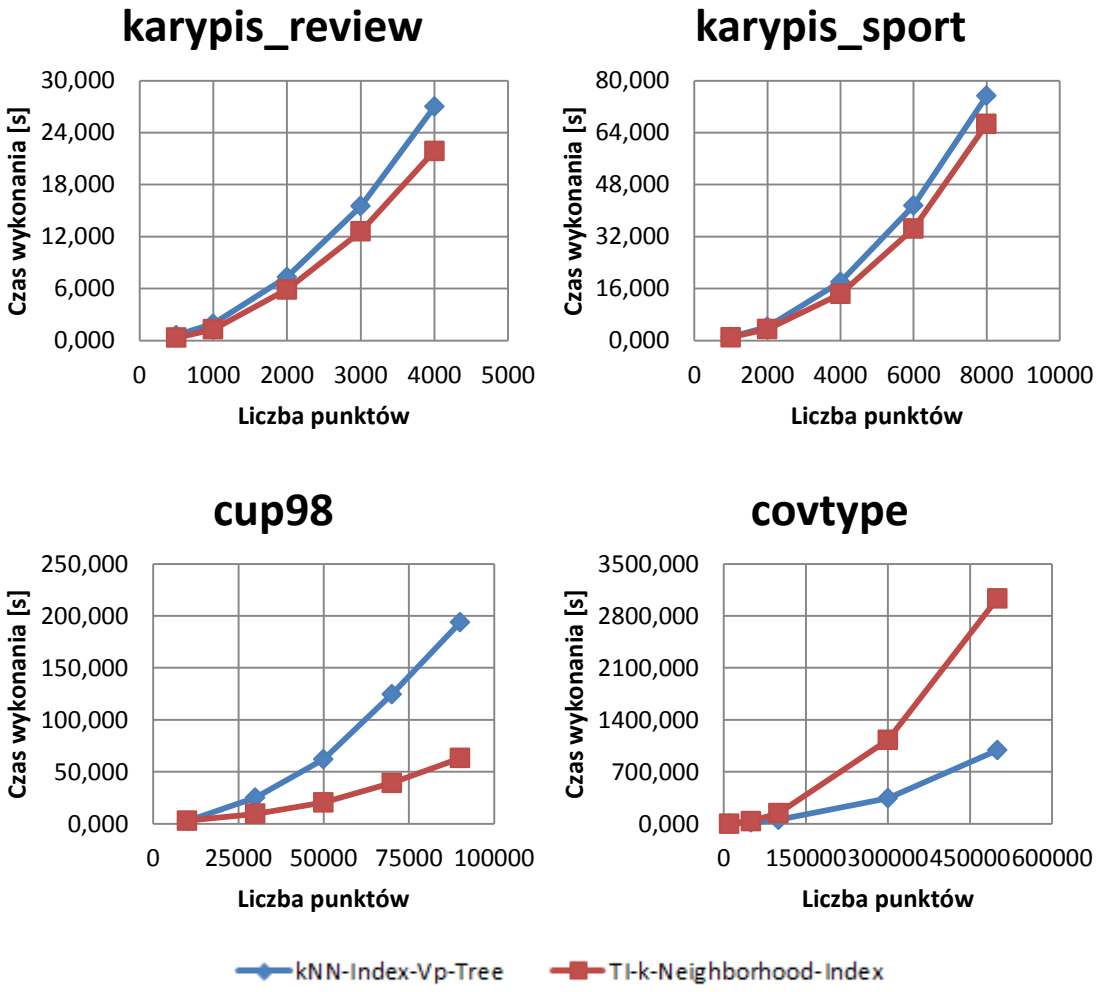
Porównanie wydajności odmian algorytmu k-Neighborhood- przy zastosowaniu miary kosinusowej jako miary podobieństwa. Wykresy zawierają czasy wykonania poszukiwań k=5 sąsiedztwa w przykładowych zbiorach danych dla 50% losowo wybranych punktów zbioru danych



Wybrane wyniki eksperymentalne

Porównanie metod przyspieszania wyznaczania sąsiedztwa – miara kosinusowa

Porównanie wydajności algorytmów kNN-Index-Vp-Tree i TI-k-Neighborhood-Index przy zastosowaniu miary kosinusowej jako miary podobieństwa. Wykresy zawierają czasy wykonania poszukiwań k=5 i k=5 sąsiedztwa sąsiadów w przykładowych zbiorach danych dla 50% losowo wybranych punktów zbioru danych



Podsumowanie

- Zaimplementowano wybrane algorytmy gęstościowego grupowania danych i wyszukiwania k sąsiedztwa z wykorzystaniem:
  - nierówności trójkąta,
  - rzutowania,
  - indeksu VP-Tree.
- W tym zaproponowano i zaimplementowano:
  - adaptację algorytmu DBSCAN i wyznaczania k sąsiedztwa z wykorzystaniem rzutowania,
  - wykorzystanie VP-Tree do wyznaczania k sąsiedztwa z użyciem mediany,
  - wykorzystanie VP-Tree do wyznaczania k sąsiedztwa z użyciem pary ograniczeń.
- Przeprowadzono serię eksperymentów, których rezultaty potwierdziły wzrost wydajności gęstościowego grupowania i wyznaczania k sąsiedztwa, gdy stosowane są badane usprawnienia, a w szczególności nierówność trójkąta.

Dziękuję za uwagę