Politechnika Wrocławska

Indukcyjne Metody Analizy Danych

Ćwiczenie 3 - Wybrane metody klasteryzacji w R

Marcel Cielinski
Index: 236747

prowadzący
dr inż. Paweł Myszkowski

30 kwietnia 2020

Wprowadzenie

1.1 Problem

Celem ćwiczenia było zapoznanie się z systemem *R*, na przykładzie zadania uczenia nienadzorowanego, zagadnienia klasteryzacji. Konkretniej, należało zaznajomić się z dwoma popularnymi tego typu algorytmami, jakimi są: *K-Means* oraz *PAM*. Do składowych zadania zalicza się oczywiście zbadanie wspomnianych algorytmów w języku *R*, przy wykorzystaniu bibliotek dostępnych na tej platformie. Należało także przeprowadzić analizy na trzech, ustalonych na potrzeby ćwiczenia, zbiorach danych. Elementem zadania jest również dobór odpowiednich dla problemu metryk, które pozwolą na ocenę jakości otrzymywanych wyników oraz identyfikacja parametrów obu technik grupowania i sprawdzenie ich wpływu owe miary.

1.2 Algorytm K-Means

Algorytm *K-średnich* (ang. *K-Means*), nazywany również algorytmem centroidów, służy do podziału danych wejściowych na z góry założoną liczbę klastrów. Jest to jedna z najprostszych technik rozwiązywania problemu klasteryzacji, należąca do grupy algorytmów zachłannych (bowiem nie ma gwarancji znalezienia optymalnego rozwiązania). Parametrem w *K-Means* jest liczba klastrów K.

Celem jest wyliczenie K centroidów (po jednym dla każdego klastra). Aby to osiągnąć algorytm dąży do minimalizacji tzw. błędu średniokwadratowego (dokładniej z ang. within-cluster sum of squares (WCSS)):

$$WCSS = \sum_{i=1}^{K} \sum_{x \in S_i} ||x - c_i||^2 \tag{1}$$

, gdzie

- K – to ilość klastrów,
- S_i – $\in S = S_1, S_2, \ldots, S_k$ to zbiór obiektów x przyporządkowanych do i-tego klastra,
- c_i – centroid i-tego klastra,
- $||x - c_i||^2$ – wybrana miara odległości między przyporządkowanym obiektem a centroidem klastra.

Zasada działania algorytmu jest następująca:

1. Wybór wartości K - liczby klastrów oraz wyznaczenie początkowych współrzędnych centroidów (np. w sposób losowy).
2. Obliczenie odległości każdej instancji danych do wszystkich centroidów - zwykle wykorzystuje się odległość euklidesową.
3. Klasteryzacja - przyporządkowanie każdego obiektu (instancji danych) do najbliższego centroida.
4. Ustalenie nowych środków klastrów - nadpisanie centroidów poprzez wyliczenie środka geometrycznego wszystkich przyporządkowanych instancji do danego klastra.

5. Powtarzanie kroków 2, 3 oraz 4 do momentu aż centroidy nie będą zmieniać położenia (lub osiągnięcia innego warunku stopu).

1.3 Algorytm PAM

Algorytm *PAM* (ang. Partitioning Around Medoids), nazywany zamiennie algorytmem *K-medoids*. Podobnie jak *K-Means*, jest to algorytm zachłanny i parametryzuje go również z góry założona liczba klastrów (K). Jednak w tym przypadku, nie są aktualizowane pozycje centroidów, a wyznaczane są pozycje medoidów - czyli najbardziej centralnych obiektów (instancji) ze zbioru danych, reprezentujących daną grupę. Odległość od wszystkich pozostałych elementów wewnątrz danej grupy jest minimalna. Algorytm ten dąży do minimalizacji sumy odległości wszystkich elementów niebędących medoidami, od najbliższych im medoidów.

Kolejną różnicą między algorytmem centroidów oraz metodą *PAM* jest sposób definiowania dystansu między instancjami danych. Algorytm ten bowiem zwykle używa odległości *Manhattan* zamiast euklidesowej. Do zalet *K-medoids* zaliczyć należy jego odporność na dane odstające (ang. outliers) oraz szumy występujące w danych (ang. robustness). Wadą natomiast jest brak możliwości zastosowania tej metody dla dużych zbiorów danych.

Przebieg algorytmu *PAM* można podzielić na dwie fazy: faza budowy (ang. build-phase) oraz faza zamiany (ang. swap-phase). Pierwsza z nich odpowiada za wybór początkowego zbioru medoidów, z kolei druga - za zamiany par m oraz o, takich aby jak najbardziej polepszyć klasteryzację.

Zasada działania algorytmu jest następująca:

1. Wybór wartości K oraz wyznaczenie początkowych medoidów (zwykle jako losowe instancje danych).
2. Obliczenie odległości każdej instancji danych do wszystkich medoidów - zwykle wykorzystuje się odległość *Manhattan*.
3. Klasteryzacja - przyporządkowanie każdego obiektu (instancji danych) do najbliższego medoida.
4. Faza zamiany - dopóki można ulepszyć obecne rozwiązanie, dla każdego medoida m i dla każdego niemedoida o:
   4.1. Zamiana m z o oraz przeliczenie kosztu (analogicznie jak Wzór 1).
   4.2. Jeżeli całkowity koszt jest większy, niż w poprzednim kroku - cofnij zamianę. Wróć do 4.

Implementacja

Do implementacji obu wybranych algorytmów klasteryzacji - *K-Means* oraz *PAM* użyto bibliotek uczenia maszynowego *cluster* i *stats* dla języka *R*. Do grupowania użyto funkcji *kmeans()* oraz *pam()*. Dla obu technik przetestowany zostanie wpływ ustawiania parametrów K (ilość klastrów) oraz metody wyznaczania dystansu między medoidem, a instancjami danych.

2.1 Ocena jakości

Do oceny jakości klasteryzacji stosuje się różne metryki. W zadaniu wykorzystano ich gotowe implementacje, dostarczone z pakietami: *clusterCrit* i *funtimes*. Konkretnie, użyto następujących miar:

- *DBI - Davies-Bouldin Index* - jest to wewnętrzny system oceny. Pod uwagę brany jest rozrzut instancji danych wewnątrz klastra oraz odległości między samymi klastrami. Wartość tej miary powinna być minimalizowana (klastry o małym rozrzucie i odległe od siebie będą traktowane jako lepsze). Wyraża się następującym wzorem:

$$DBI = \frac{1}{K}\sum_{k=1}^{K} M_k = \frac{1}{K}\sum_{k=1}^{K} \max_{k' \neq k}\left(\frac{\delta_k + \delta_{k'}}{\Delta_{kk'}}\right) \tag{2}$$

  , gdzie

  K – to ilość klastrów,
  δ_k – średnia odległość instancji w klastrze k od centroida,
  $\Delta_{kk'}$ – odległość między centroidami klastów k oraz k'.

- *Dunn Index* - jest to wewnętrzny system oceny, w którym wynik opiera się na samych danych klastrowych. Podobnie jak w przypadku wszystkich innych takich wskaźników, celem jest zidentyfikowanie zestawów klastrów, które są zwarte, z niewielką różnicą między instancjami należącymi do klastra oraz są dobrze oddzielone. Pod uwagę brana jest odległość między instancjami w różnych klastrach oraz w tym samym klastrze. Wyższy wskaźnik *Dunn* oznacza lepsze klastrowanie. Wyraża się następującym wzorem:

$$Dunn = \frac{d_{min}}{d_{max}} \tag{3}$$

  , gdzie

  d_{min} – to minimalna odległość między punktami należącymi do różnych klastrów (spośród wszystkich par klastrów),
  d_{max} – to maksymalna odległość między punktami w ramach jednego klastra (spośród wszystkich klastrów).

- *Silhouette coefficient* - jest to wewnętrzny system oceny. Wartość *silhouette* jest miarą podobieństwa obiektu do własnego klastra (kohezji) w porównaniu do innych klastrów (separacja). Wykres *silhouette* obrazuje miarę zbliżenia każdej instancji jednej grupy do obiektów w sąsiednich klastrach, a tym samym umożliwia wizualną ocenę parametrów takich jak liczba klastrów. Wyraża się następującym wzorem:

$$s(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}} \tag{4}$$

$$Silhouette = \frac{1}{K}\sum_{i=1}^{K} s_k \tag{5}$$

, gdzie

K – to ilość klastrów,
$a(i)$ – to średnia odległość między i-tym obiektem danego klastra, a wszystkimi innymi instancjami w tym samym klastrze,
$b(i)$ – to najmniejsza średnia odległość do wszystkich punktów w dowolnym innym klastrze, do którego i-ty obiekt nie należy,
$s(i)$ – to *silhouette* (wartość) dla i-tego obiektu,
s_k – to średnia *silhouette* (wartość) dla konkretnego klastra.

- *Purity* - jest to zewnętrzne kryterium oceny jakości grupowania. Bazuje na na liczbie wystąpień najliczniejszej klasy instancji w każdym z klastrów. Jest to procent całkowitej liczby obiektów (instancji danych), które zostały poprawnie sklasyfikowane. Wyraża się następującym wzorem:

$$Purity = \frac{1}{N} \sum_{k \in K} \max_{d \in D} |k \cap d| \tag{6}$$

, gdzie

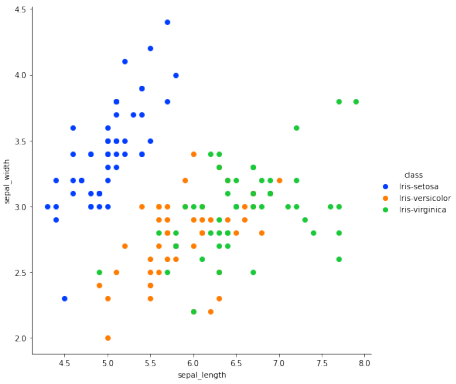
N – to liczba instancji danych,
K – to zbiór wszystkich klastrów,
D – to zbiór wszystkich klas.

2.2 Zbiory danych

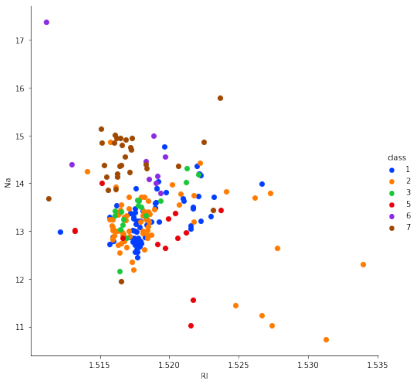
Do walidacji zaimplementowanych funkcjonalności wykorzystano łącznie cztery zbiory danych. Jeden w celu sprawdzenia działania i trzy pozostałe w celach testów i przeprowadzenia badań. Te zbiory, to:

- *iris* - gdzie liczba atrybutów to 4, a klas 3. Zbiór przedstawia odmiany kwiatów Iris, a atrybuty odpowiadają ich cechom.
- *glass* - gdzie liczba atrybutów to 9, a klas 7. Zbiór przedstawia typy szkła, gdzie atrybuty opisują skład chemiczny.
- *wine* - gdzie liczba atrybutów to 13, a klas 3. Zbiór ten zawiera dane o analizie chemicznej win z tego samego regionu, jednak z 3 różnych upraw.
- *seeds* - gdzie liczba atrybutów wynosi 7, a klas 3. Zbiór ten zawiera miary geometrycznych własności 3 różnych odmian ziaren pszenicy.

Na Rysunkach 1, 2, 3, 4 przedstawiono przykładowe rozkłady klas względem wybranych atrybutów. Z kolei w Tabelach 1, 2, 3, 4 zestawiono rozkład klas dla wszystkich badanych zbiorów.



Rysunek 1: Iris



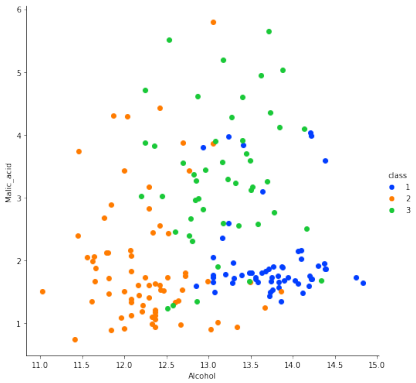
Rysunek 2: Glass

Klasa	Instancje	% rekordów
Setosa	50	33 (%)
Versicolor	50	33 (%)
Virginica	50	33 (%)

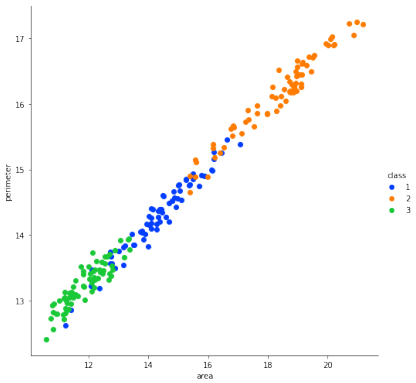
Tablica 1: Klasy zbioru *Iris*

Klasa	Instancje	% rekordów
1	70	33 (%)
2	76	36 (%)
3	17	8 (%)
4	0	0 (%)
5	13	6 (%)
6	9	4 (%)
7	29	13 (%)

Tablica 2: Klasy zbioru *Glass*



Rysunek 3: Wine



Rysunek 4: Seeds

Klasa	Instancje	% rekordów
1	59	33 (%)
2	71	40 (%)
3	48	27 (%)

Tablica 3: Klasy zbioru *Wine*

Klasa	Instancje	% rekordów
1	70	33 (%)
2	70	33 (%)
3	70	33 (%)

Tablica 4: Klasy zbioru *Seeds*

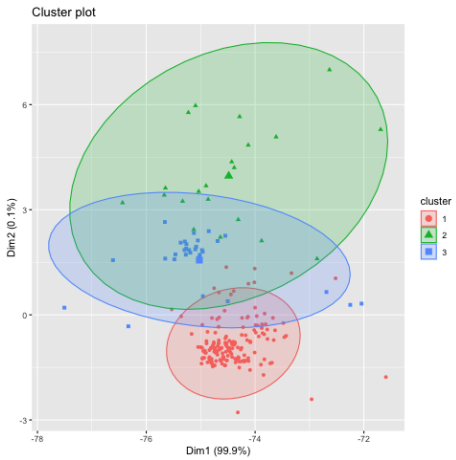
Wyniki eksperymentów

3.1 Zbiór Glass

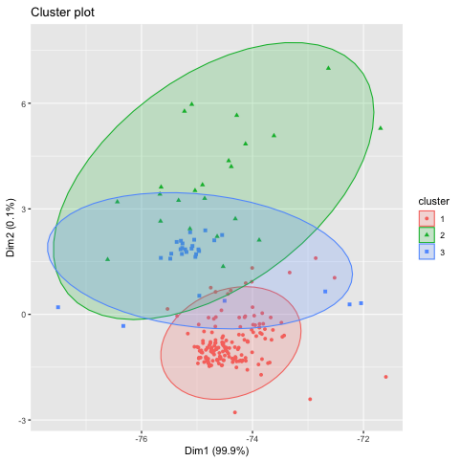
Poniższa tabela (Tabela 5) przedstawia wszystkie przeprowadzone testy na zbiorze *Glass*, z podziałem na wykorzystany algorytm klasteryzacji, wielkość parametru K (ilość klastrów) oraz metodę obliczania odległości instancji danych do medoidów dla *PAM*. W kolejnych kolumnach umieszczone zostały odczyty z metryk (wspomnianych w sekcji wyżej) dla każdego badania.

Dodatkowo na rysunkach: 7, 8 oraz 9 przedstawione są wykresy zależności owych metryk od wielkości K, nałożone na jedną płaszczyznę.

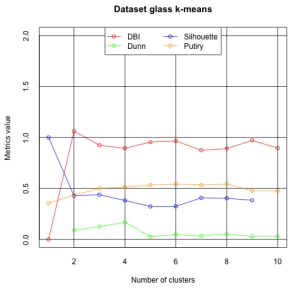
Dla tego zbioru, przy wykorzystaniu algorytmu *K-Means*, najlepsze wyniki osiągamy dla podziału danych na 3 lub 4 klastry. Natomiast stosując grupowanie *PAM* - dla 3 klastrów.



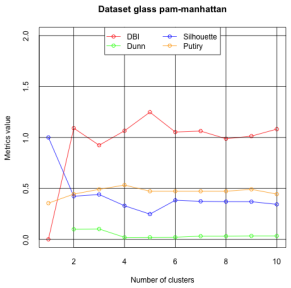
Rysunek 5: Glass - *K-Means* - $K = 3$



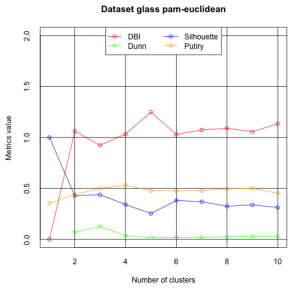
Rysunek 6: Glass - *PAM* - $K = 3$



Rysunek 7: *K-Means*



Rysunek 8: *PAM manh.*



Rysunek 9: *PAM eucl.*

K	*DBI*	*Dunn*	*Silhouette*	*Purity*
K-Means				
2	1.06	0.09	0.43	0.43
3	0.92	0.12	0.44	0.50
4	0.89	0.17	0.38	0.51
5	0.90	0.03	0.37	0.53
6	0.94	0.05	0.36	0.54
7	0.87	0.03	0.41	0.53
8	0.89	0.05	0.40	0.54
9	0.97	0.03	0.38	0.48
10	0.90	0.03		0.48
PAM (manhattan)				
2	1.09	0.10	0.42	0.44
3	0.92	0.10	0.44	0.49
4	1.07	0.02	0.33	0.53
5	1.25	0.02	0.25	0.47
6	1.05	0.02	0.38	0.47
7	1.06	0.03	0.37	0.47
8	0.99	0.03	0.37	0.47
9	1.01	0.03	0.37	0.49
10	1.08	0.03	0.34	0.44
PAM (euclidean)				
2	1.06	0.07	0.43	0.44
3	0.92	0.12	0.44	0.50
4	1.03	0.04	0.34	0.53
5	1.25	0.02	0.25	0.48
6	1.03	0.02	0.38	0.48
7	1.07	0.02	0.37	0.48
8	1.09	0.02	0.32	0.50
9	1.06	0.03	0.34	0.50
10	1.13	0.03	0.31	0.45

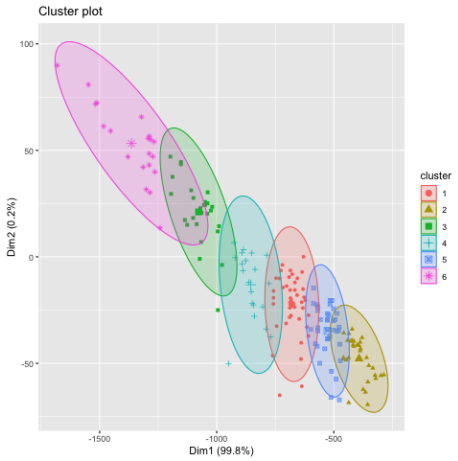
Tablica 5: Wpływ parametru K (ilość klastrów) oraz metody klasteryzacji na metryki dla zbioru *Glass*

3.2 Zbiór Wine

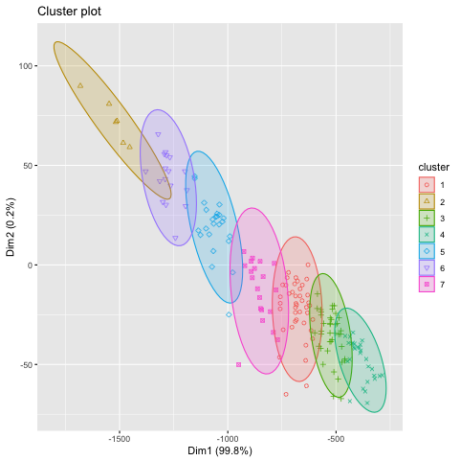
Poniższa tabela (Tabela 6) przedstawia wszystkie przeprowadzone testy na zbiorze *Wine*, z podziałem na wykorzystany algorytm klasteryzacji, wielkość parametru K (ilość klastrów) oraz metodę obliczania odległości instancji danych do medoidów dla *PAM*. W kolejnych kolumnach umieszczone zostały odczyty z metryk (wspomnianych w sekcji wyżej) dla każdego badania.

Dodatkowo na rysunkach: 12, 13 oraz 14 przedstawione są wykresy zależności owych metryk od wielkości K, nałożone na jedną płaszczyznę.

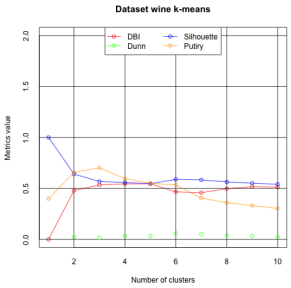
W przypadku zbioru *Wine*, przy wykorzystaniu algorytmu *K-Means*, najlepsze wyniki osiągamy dla podziału danych na 2, 3 lub 6 klastrów. Natomiast stosując grupowanie *PAM* - dla 2, 3 lub 7 klastrów.



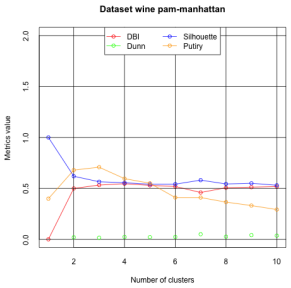
Rysunek 10: Wine - *K-Means* - $K = 6$



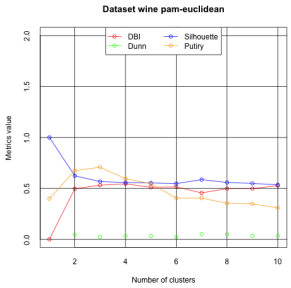
Rysunek 11: Wine - *PAM* - $K = 7$



Rysunek 12: *K-Means*



Rysunek 13: *PAM manh.*



Rysunek 14: *PAM eucl.*

K	*DBI*	*Dunn*	*Silhouette*	*Purity*
K-Means				
2	0.48	0.02	0.64	0.66
3	0.53	0.02	0.57	0.70
4	0.54	0.03	0.56	0.60
5	0.55	0.03	0.55	0.55
6	0.47	0.06	0.59	0.53
7	0.46	0.05	0.58	0.40
8	0.50	0.04	0.56	0.36
9	0.52	0.03	0.55	0.33
10	0.51	0.03	0.54	0.30
PAM (manhattan)				
2	0.50	0.02	0.62	0.68
3	0.53	0.02	0.57	0.71
4	0.55	0.02	0.56	0.60
5	0.53	0.02	0.54	0.55
6	0.52	0.02	0.54	0.41
7	0.46	0.05	0.58	0.41
8	0.51	0.03	0.54	0.37
9	0.51	0.04	0.55	0.33
10	0.52	0.04	0.53	0.29
PAM (euclidean)				
2	0.50	0.04	0.62	0.67
3	0.53	0.02	0.57	0.71
4	0.54	0.03	0.56	0.60
5	0.51	0.03	0.55	0.54
6	0.51	0.02	0.55	0.40
7	0.46	0.05	0.59	0.40
8	0.50	0.05	0.56	0.35
9	0.50	0.04	0.55	0.35
10	0.53	0.04	0.54	0.31

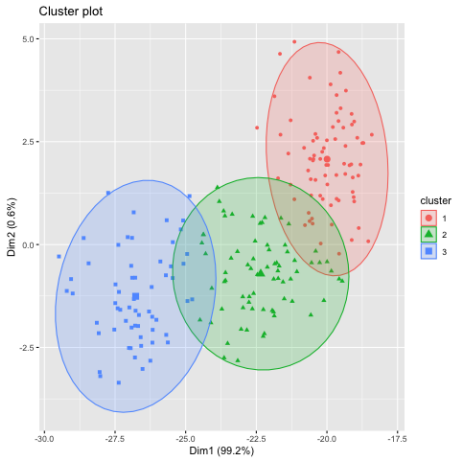
Tablica 6: Wpływ parametru K (ilość klastrów) oraz metody klasteryzacji na metryki dla zbioru *Wine*

3.3 Zbiór Seeds

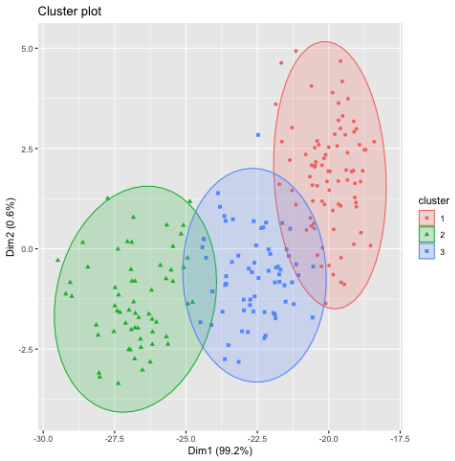
Poniższa tabela (Tabela 7) przedstawia wszystkie przeprowadzone testy na zbiorze *Seeds*, z podziałem na wykorzystany algorytm klasteryzacji, wielkość parametru K (ilość klastrów) oraz metodę obliczania odległości instancji danych do medoidów dla *PAM*. W kolejnych kolumnach umieszczone zostały odczyty z metryk (wspomnianych w sekcji wyżej) dla każdego badania.

Dodatkowo na rysunkach: 17, 18 oraz 19 przedstawione są wykresy zależności owych metryk od wielkości K, nałożone na jedną płaszczyznę.

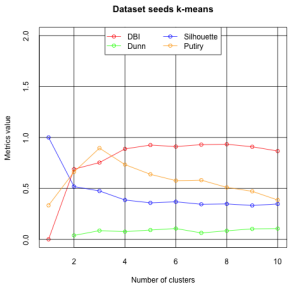
W przypadku zbioru *Seeds*, przy wykorzystaniu algorytmu *K-Means*, najlepsze wyniki osiągamy dla podziału danych na 2 lub 3 klastry. Stosując grupowanie *PAM*, najlepsze wskazania przypadają dla tych samych wartości K.



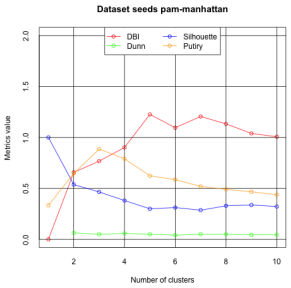
Rysunek 15: Seeds - *K-Means* - $K = 3$



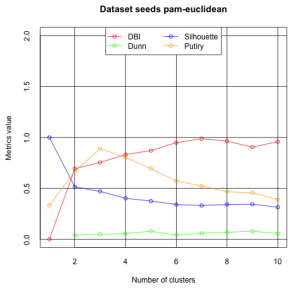
Rysunek 16: Seeds - *PAM* - $K = 3$



Rysunek 17: *K-Means*



Rysunek 18: *PAM manh.*



Rysunek 19: *PAM eucl.*

K	*DBI*	*Dunn*	*Silhouette*	*Purity*
K-Means				
2	0.69	0.04	0.52	0.66
3	0.75	0.09	0.47	0.90
4	0.89	0.08	0.39	0.73
5	0.93	0.09	0.36	0.64
6	0.91	0.11	0.37	0.58
7	0.93	0.06	0.34	0.58
8	0.93	0.08	0.35	0.51
9	0.91	0.10	0.33	0.47
10	0.87	0.11	0.35	0.39
PAM (manhattan)				
2	0.66	0.06	0.54	0.65
3	0.77	0.05	0.47	0.89
4	0.90	0.06	0.38	0.79
5	1.23	0.05	0.30	0.62
6	1.10	0.04	0.31	0.59
7	1.21	0.05	0.29	0.52
8	1.13	0.05	0.33	0.49
9	1.04	0.04	0.34	0.47
10	1.01	0.05	0.32	0.44
PAM (euclidean)				
2	0.69	0.04	0.51	0.67
3	0.75	0.05	0.47	0.89
4	0.83	0.06	0.40	0.80
5	0.87	0.08	0.38	0.70
6	0.95	0.04	0.34	0.57
7	0.99	0.06	0.33	0.52
8	0.96	0.07	0.34	0.47
9	0.90	0.08	0.35	0.46
10	0.96	0.06	0.32	0.39

Tablica 7: Wpływ parametru K (ilość klastrów) oraz metody klasteryzacji na metryki dla zbioru *Seeds*

Podsumowanie

Zadanie pozwoliło na zaznajomienie się nie tylko z algorytmami klasteryzacji, jakimi są *K-Means* oraz *PAM*, ale także z wieloma narzędziami służącymi pracy na danych oraz, ogólnie rzecz ujmując, platformą języka *R*, która jak wiadomo jet powszechnie wykorzystywana do obliczeń statystycznych. Dodatkową wartością edukacyjną jest zapoznanie się z metodami służącymi do oceny jakości grupowania.

W porównaniu z językiem *Python*, środowisko *R* zdaje się być bardzo dobrze przygotowane pod problemy bazujące na danych. Wiele elementów z tym związanych jest uproszczonych, jak również dostępne biblioteki pozwalają na implementację złożonych funkcjonalności niewielkim nakładem pracy (a przynajmniej kodu). Sprawnie rozwiązany jest także aspekt wizualizacji danych oraz wyświetlania różnego rodzaju wykresów, z koeli sposób wykonywania poleceń także służy temu, do czego środowisko to zostało stworzone.