Politechnika Gdańska
Wydział Fizyki Technicznej i Matematyki Stosowanej

Projekt 2
Klasteryzacja

Kamil Łangowski

prowadzący:
dr inż. Anna Szafrańska

22 czerwca 2022

Spis treści

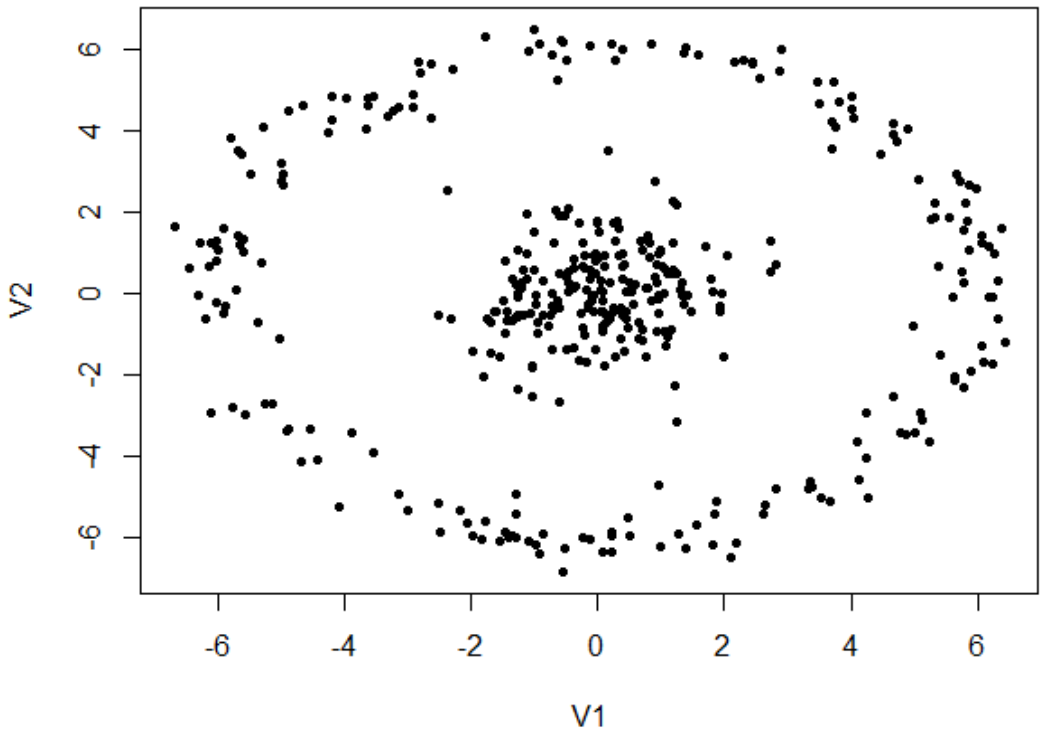
1 Zadanie 1

1.1 Wczytanie danych i wykres

W celu uniknięcia niedogodności w oryginalnym zbiorze danych zmieniono separator dziesiętny z przecinka na kropkę wykorzystując do tego celu operację *znajdź i zamień* w pakiecie *Excel*.

Dane wczytano do *Rstudio*. Ustalono, że zbiór danych składa się z 400 obserwacji opisywanych przez dwie numeryczne cechy - *V1* i *V2*.

Na rys. 1 przedstawiono wykres danych.

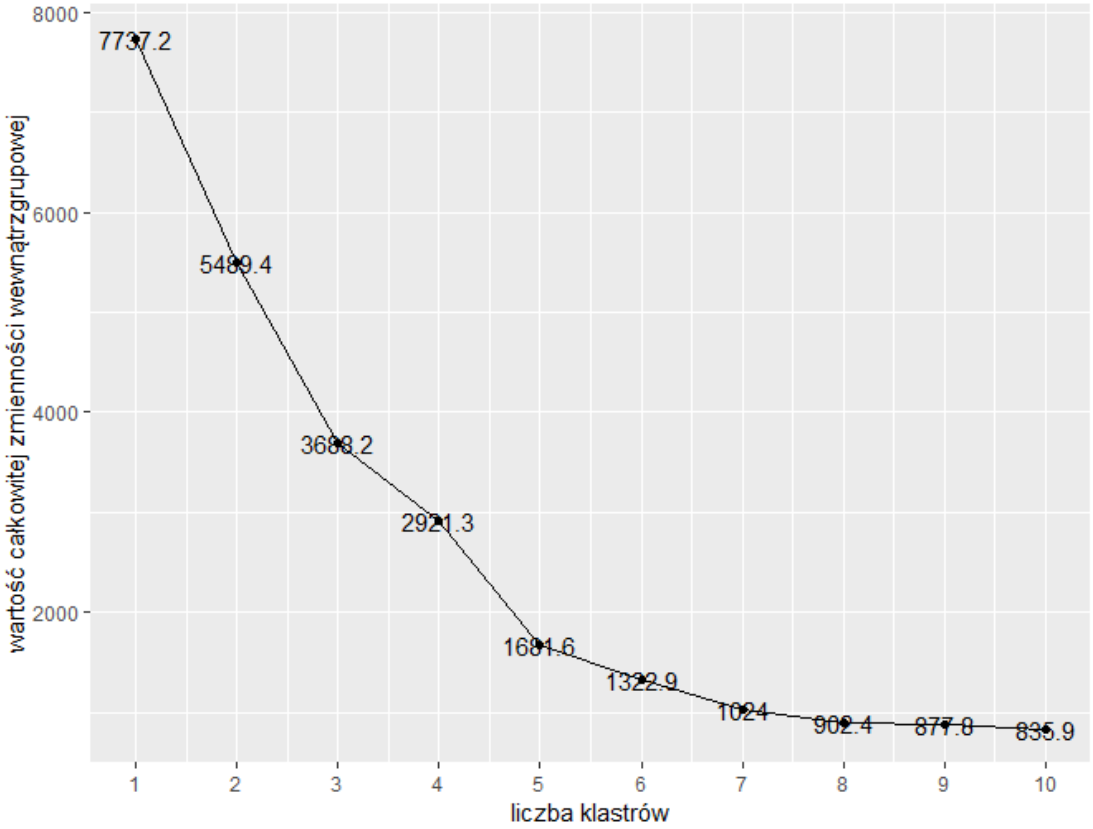


Rysunek 1: Wykres danych z zad. 1.

Jak łatwo zauważyć, dane są pogrupowane. Część danych znajduje się w grupie po środku, a część w „okręgu” otaczając środek.

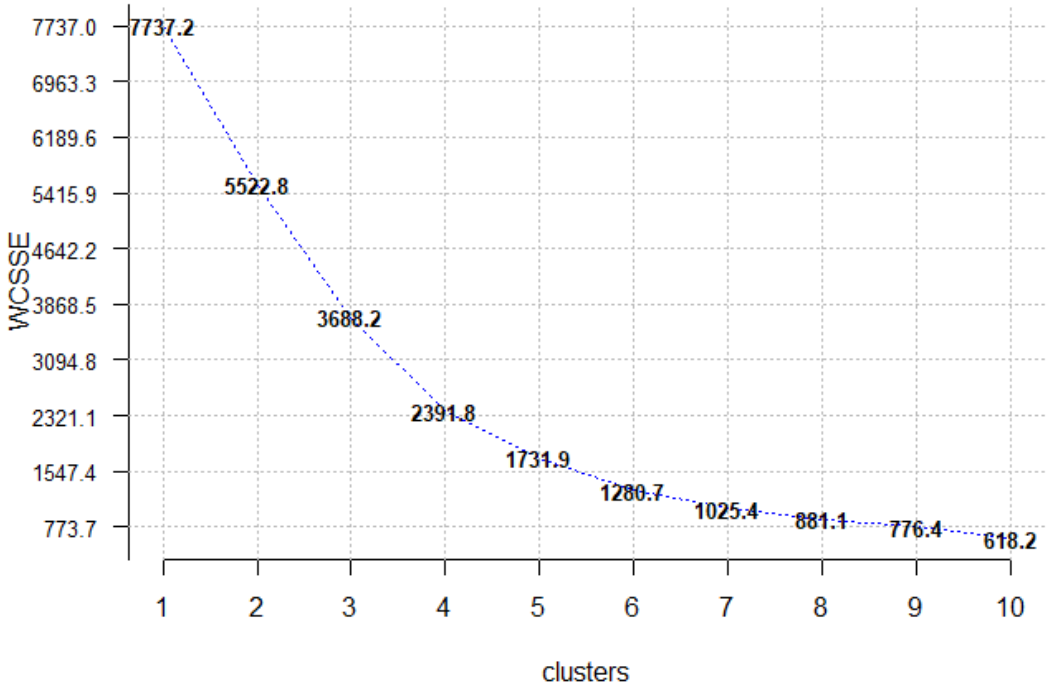
1.2 Klasteryzacja metodą k-średnich

W celu analizy optymalnej liczby klastrów wykonujemy klasteryzjację za pomocą funkcji *kmeans()*. Zaczynając od $k = 1$ do $k = 10$ (w pętli) zapisujemy wartości całkowitej zmienności wewnątrzgrupowej (WCSSE). Na podstawie danych WCSSE tworzymy wykres osypiska w zależności od liczby klastrów. Wykres umieszczono na rys. 2.



Rysunek 2: Wykres osypiska dla pętli.

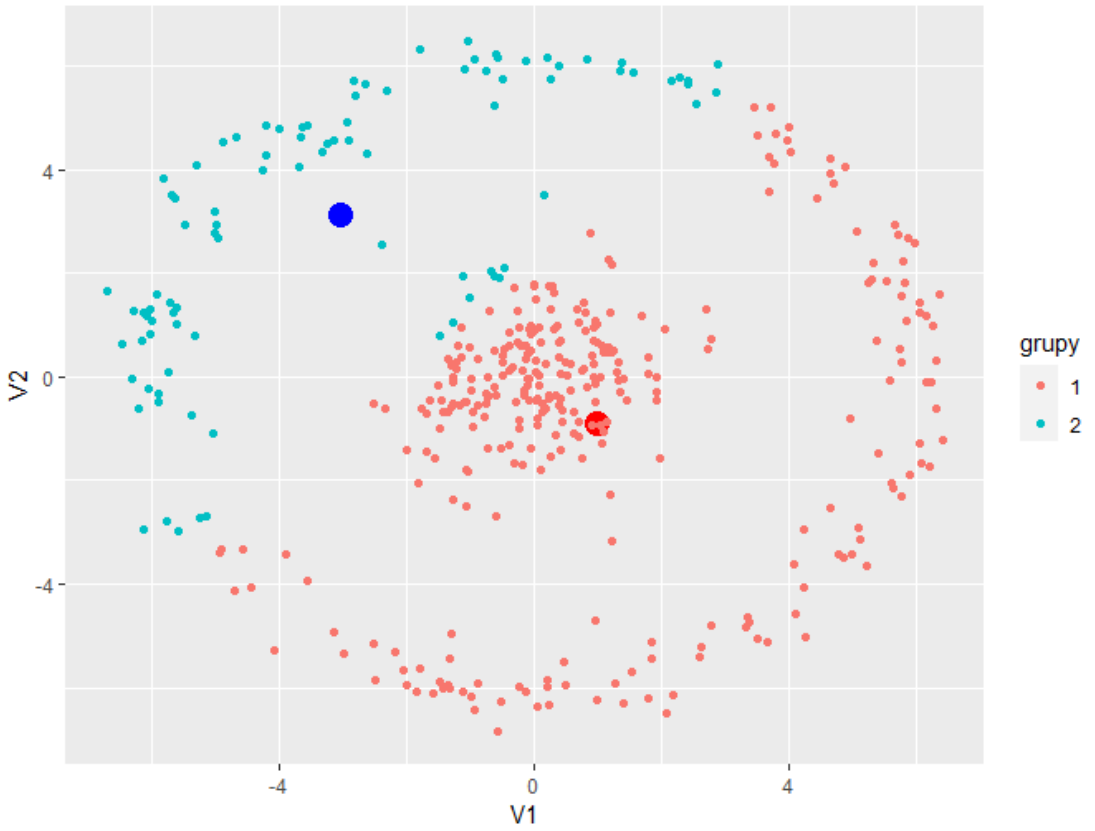
Dla porównania tworzymy wykres osypiska posługując się funkcją *Optimal_Clusters_KMeans()*. Wykres umieszczono na rys. 3.



Rysunek 3: Wykres osypiska dla funkcji.

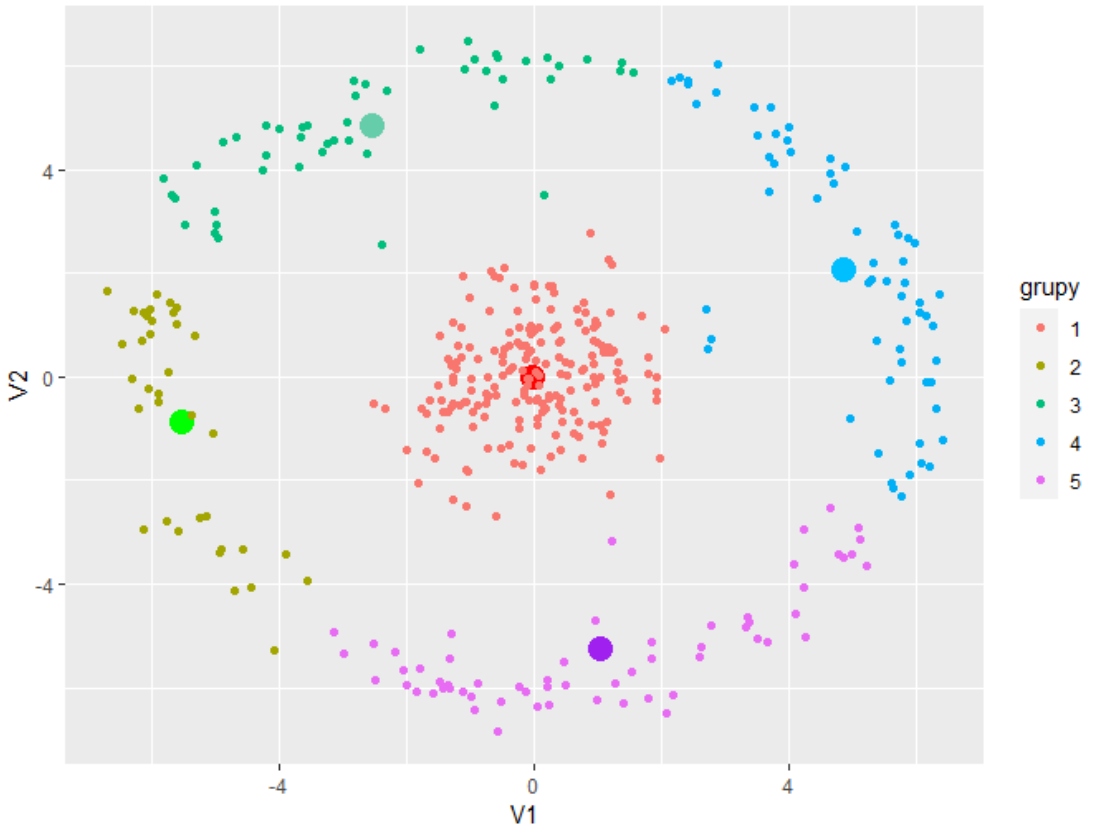
Wartości nieznacznie się różnią, jednak kształt krzywej jest w przybliżeniu ten sam. Wykres osypiska nie wskazuje jednoznacznie na najlepszy wybór liczby klastrów z uwagi na brak konkretnego punktu zgięcia wykresu, najbliższy to 4 lub 5.

Wybieramy $k = 2$ (jest to wybór arbitralny) i tworzymy wykres. Jego rezultat przedstawiono na rys. 4.



Rysunek 4: Rezultat dla dwóch klastrów. Na wykresie kółkami zaznaczono centroidy: niebieski dla grupy 2, czerwony dla grupy 1.

Wykres osypiska sugeruje, że najtrafniejszym wyborem byłoby przyjęcie $k = 5$ (ew. 4), z uwagi na fakt, że krzywa po tej wartości spłaszcza się. Na rys. 5 przedstawiono rezultat dla $k = 5$.



Rysunek 5: Rezultat dla pięciu klastrów. Na wykresie kółkami zaznaczono centroidy.

Intuicja podpowiada nam, że wynik klasteryzacji dla obu przypadków nie jest poprawny. Spodziewamy się grupowania, w którym „środek okręgu" jest jedną grupą, natomiast „okrąg" jest grupą drugą.

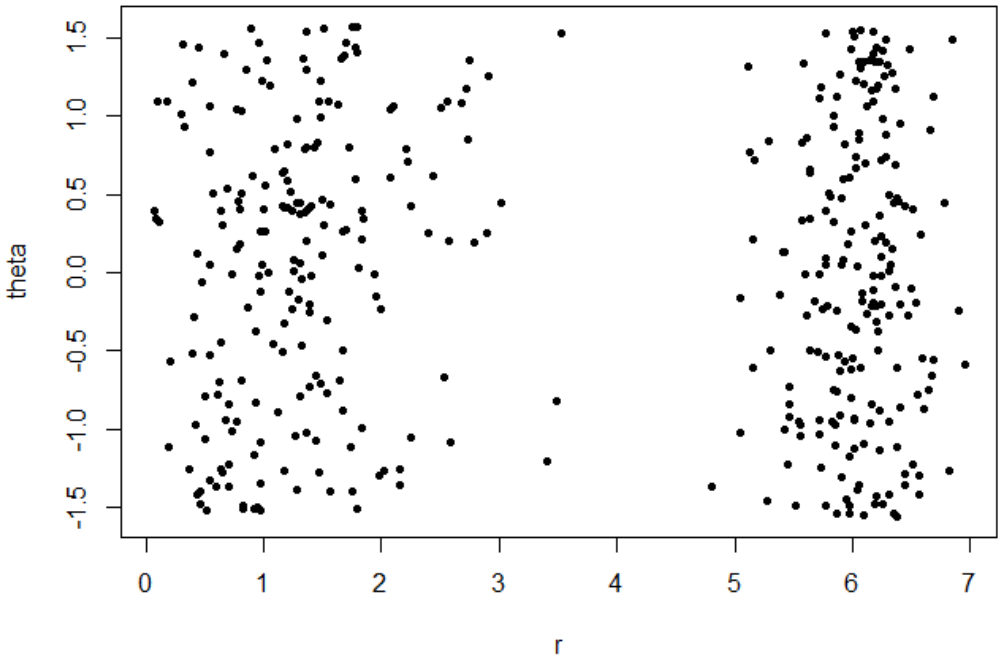
Rozbieżność intuicji z wynikami jest konsekwencją rozmieszczenia danych i sposobu działania algorytmu, w szczególności mierzenia odległości od centroidów. Kule odległości od centroidów nachodzą na siebie, gdyż „po drodze" trafiają na „środek okręgu".

1.3 Prawidłowa klasteryzjacja metodą k-średnich

Z uwagi na „okrągły" charakter naszego zbioru danych dokonamy transformacji układu kartezjańskiego w układ krzywoliniowy – zmiennych biegunowych. Zabieg ten pozwoli na rozdzielenie danych ze „środka okręgu" od danych z „okręgu".

Wyznaczamy nowe zmienne jako $r = \sqrt{x_{V1_i}^2 + x_{V2_i}^2}$ oraz $\theta = \arctan\left(\frac{x_{V2_i}}{x_{V1_i}}\right)$.

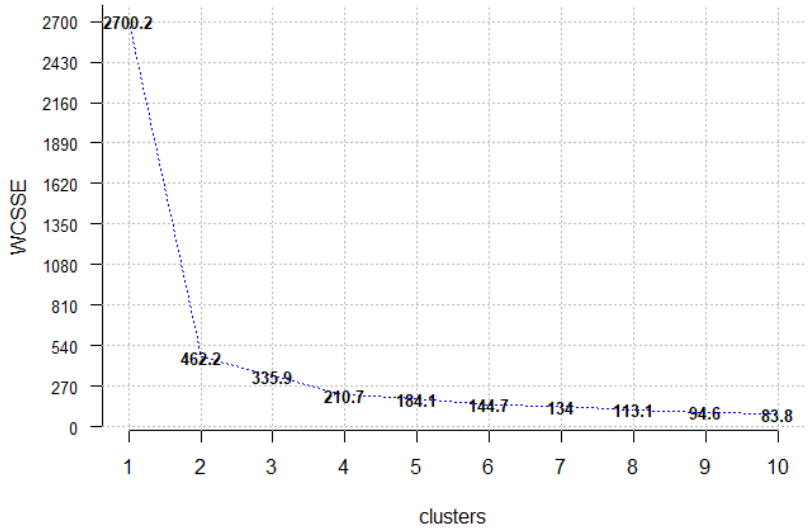
W rezultacie dostajemy dane rozdzielone w sposób przedstawiony na rys. 6.



Rysunek 6: Dane po zamianie na współrzędne biegunowe.

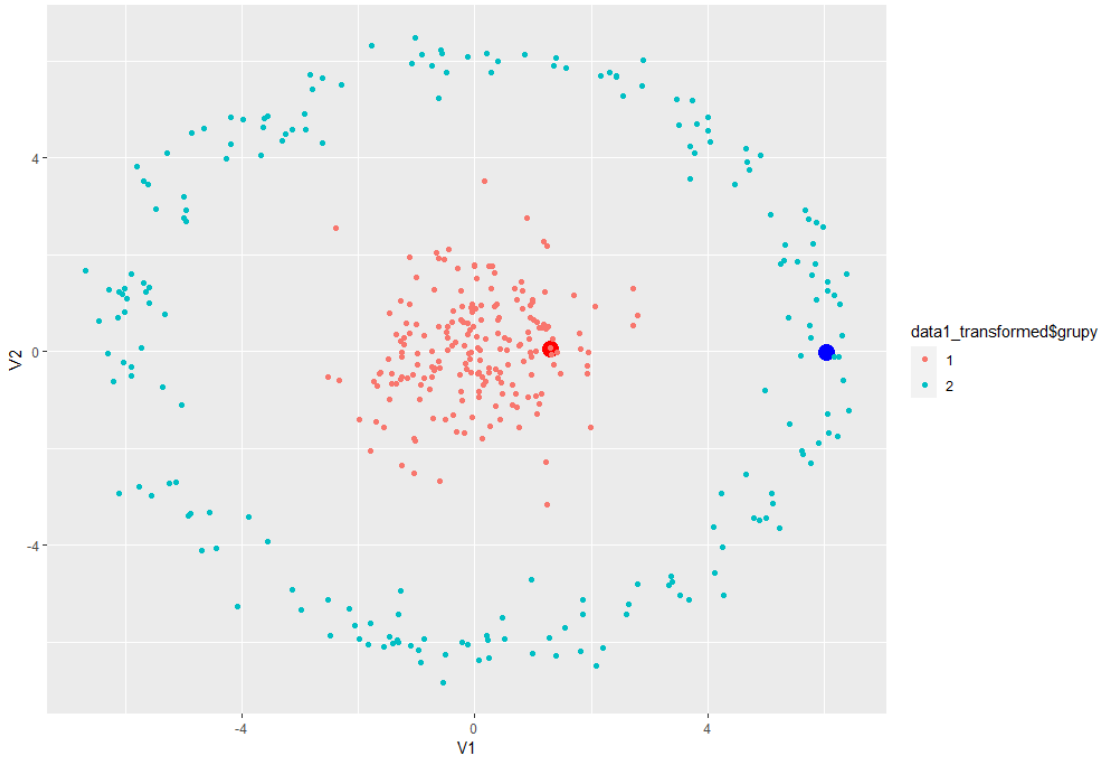
W powyższym układzie dużo łatwiej jest dostrzec podział danych.

Tworzymy wykres osypiska dla transformowanych danych.



Rysunek 7: Wykres osypiska we współrzędnych biegunowych.

Na podstawie wykresu przyjmujemy $k = 2$. Dokonujemy klasteryzacji. Wyniki przedstawiono na rys. 8.



Rysunek 8: Rezultat klasteryzacji we współrzędnych biegunowych. Kółkami zaznaczono centroidy.

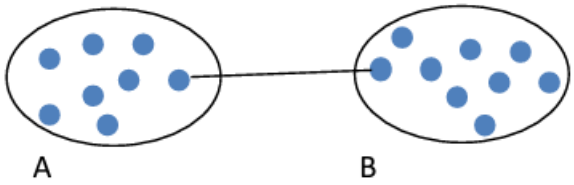
Powyższy wynik jest zgodny z naszymi przewidywaniami. Możemy stwierdzić, że klasteryzację wykonano poprawnie.

1.4 Algorytmy hierarchiczne

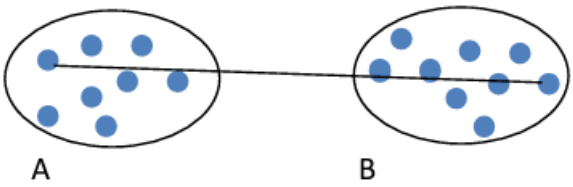
W tym podejściu zastosujemy dwie metody hierarchiczne.

- Metoda pojedynczego wiązania (SL) – w metodzie tej odległość między dwoma skupieniami jest określona przez odległość między dwoma najbliższymi obiektami (najbliższymi sąsiadami) należącymi do różnych skupień. Zgodnie z tą zasadą obiekty formują skupienia łącząc się w ciągi, a wynikowe skupienia tworzą długie "łańcuchy".

- Metoda pełnego wiązania (CL) – w tej metodzie odległość między skupieniami jest zdeterminowana przez największą z odległości między dwoma dowolnymi obiektami należącymi do różnych skupień (tzn. "najdalszymi sąsiadami"). Metoda ta zwykle zdaje egzamin w tych przypadkach, kiedy obiekty faktycznie formują naturalnie oddzielone "kępki". Metoda ta nie jest odpowiednia, jeśli skupienia są w jakiś sposób wydłużone lub mają naturę "łańcucha".

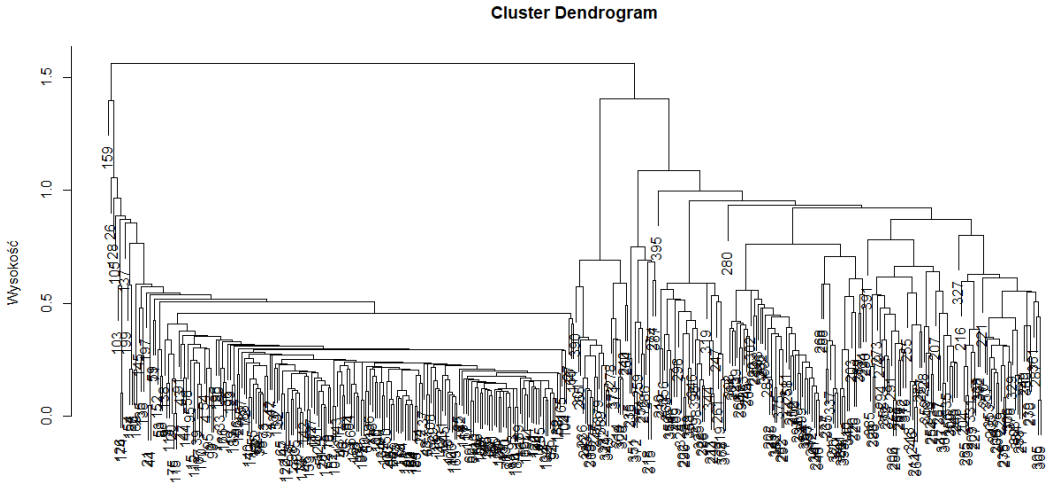


Rysunek 9: Metoda pojedynczego wiązania.

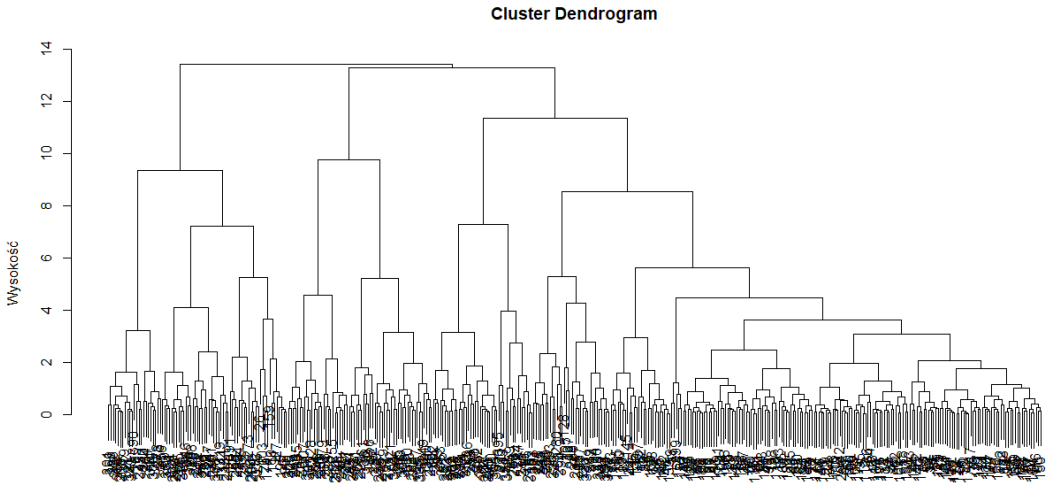


Rysunek 10: Metoda pełnego wiązania.

Za pomocą funkcji *hclust()* tworzymy dendrogramy dla oryginalnych danych, zarówno dla SL jak CL.

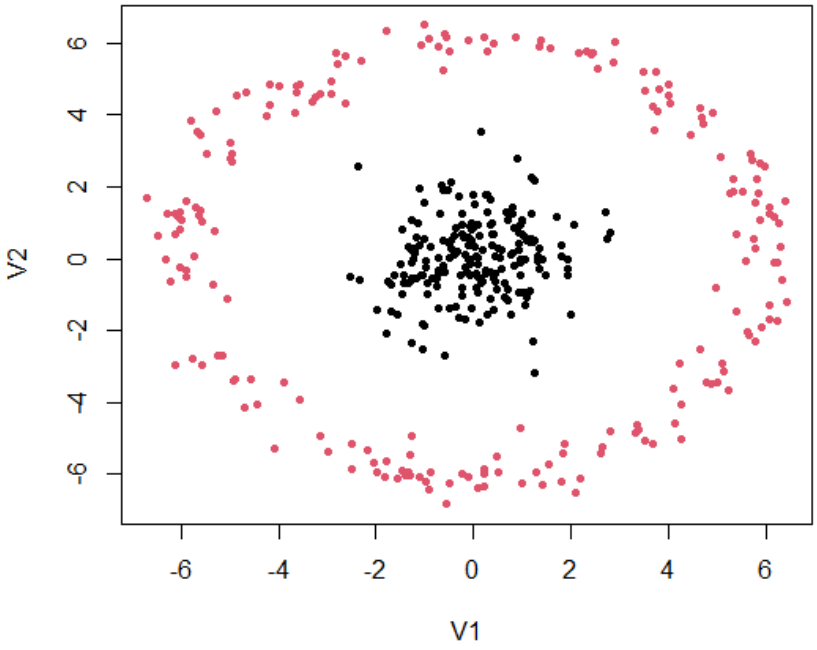


Rysunek 11: Metoda pojedynczego wiązania – dendrogram.

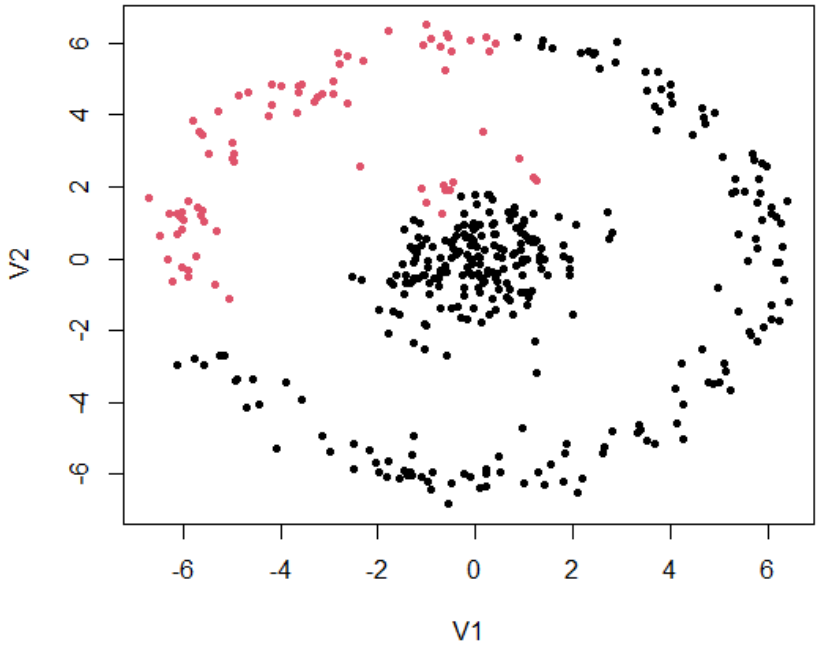


Rysunek 12: Metoda pełnego wiązania – dendrogram.

Generujemy wektor etykiet przynależności punktów (obserwacji) do poglądowej ilości klastrów za pomocą funkcji *cuttree()*, a następnie przedstawiamy wynik grupowania hierarchicznego na wykresie z kolorowaniem według wektora etykiet.



Rysunek 13: Metoda pojedynczego wiązania – wynik grupowania.



Rysunek 14: Metoda pełnego wiązania – wynik grupowania.

Łatwo zauważyć, że w przypadku naszych danych metoda pełnego wiązania nie sprawdza się tak dobrze jak metoda pojedynczego wiązania.

2 Zadanie 2

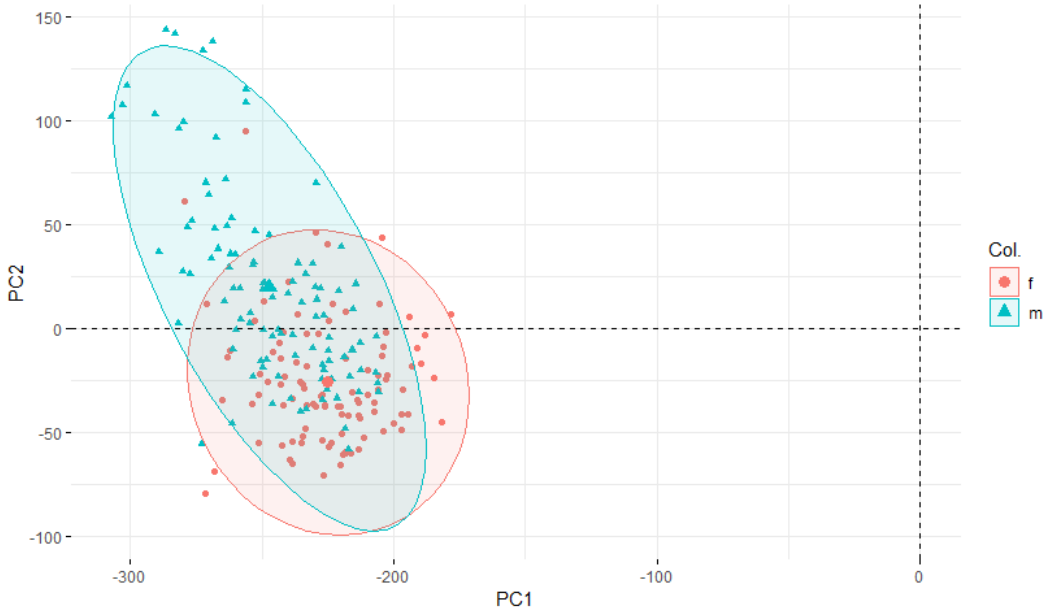
2.1 Wczytanie danych

z pakietu *DAAG* wczytujemy 11 pierwszych cech zbioru danych *ais*[1] . Zbiór zawiera 202 obserwacje, o zmiennych numerycznych. Zapisujemy także informację o płci osobnika.

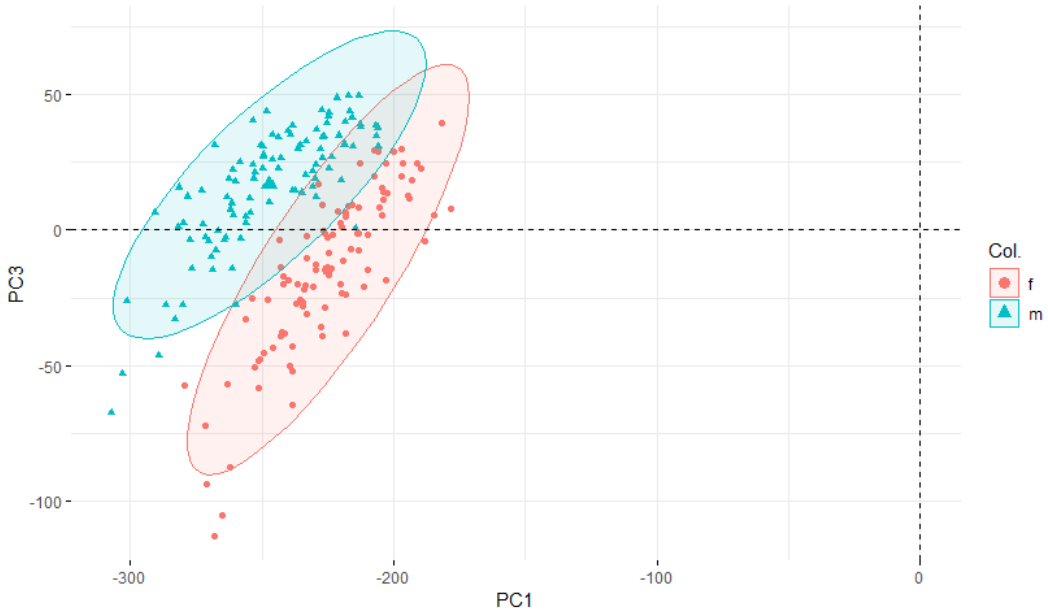
2.2 PCA na nieskalowanych danych

Za pomocą funkcji *prcomp* dokonujemy analizy PCA na danych nieskalowanych (odpowiednie argumenty funkcji). Wyniki dla pierwszych trzech składowych głównych obrazujemy na wykresach dwu- i trójwymiarowym, zaznaczamy kolorem podział danych względem płci.

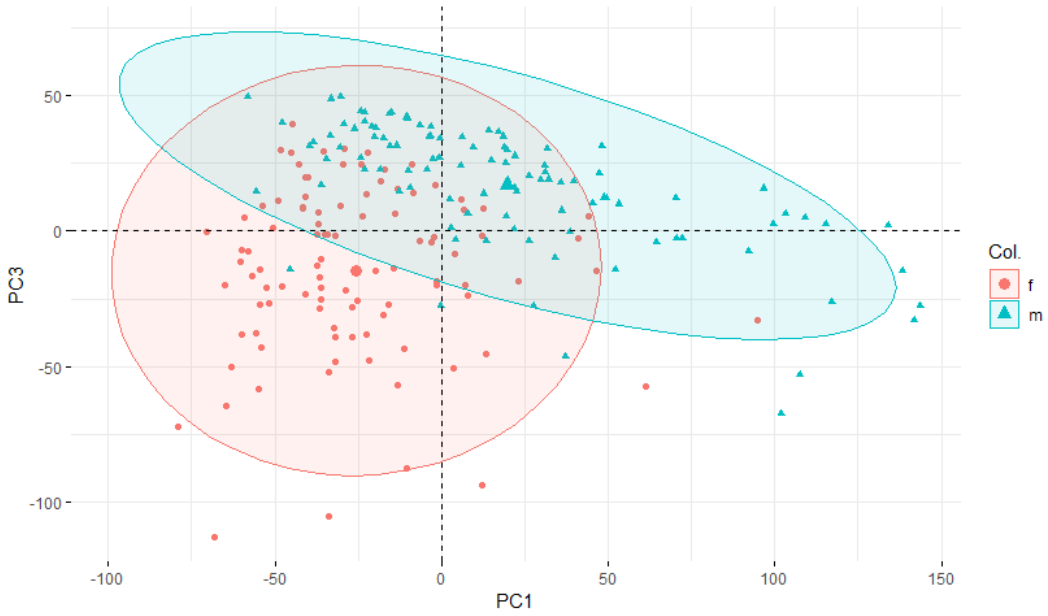
[1]Dane te zebrano w ramach badania, w jaki sposób dane dotyczące różnych cech krwi zmieniały się w zależności od sportu, wielkości ciała i płci sportowca.



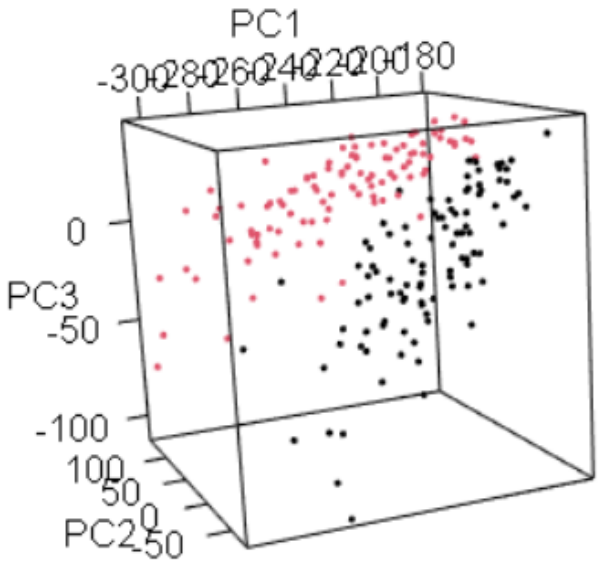
Rysunek 15: Wyniki dla PC1 i PC2.



Rysunek 16: Wyniki dla PC1 i PC3.



Rysunek 17: Wyniki dla PC2 i PC3.

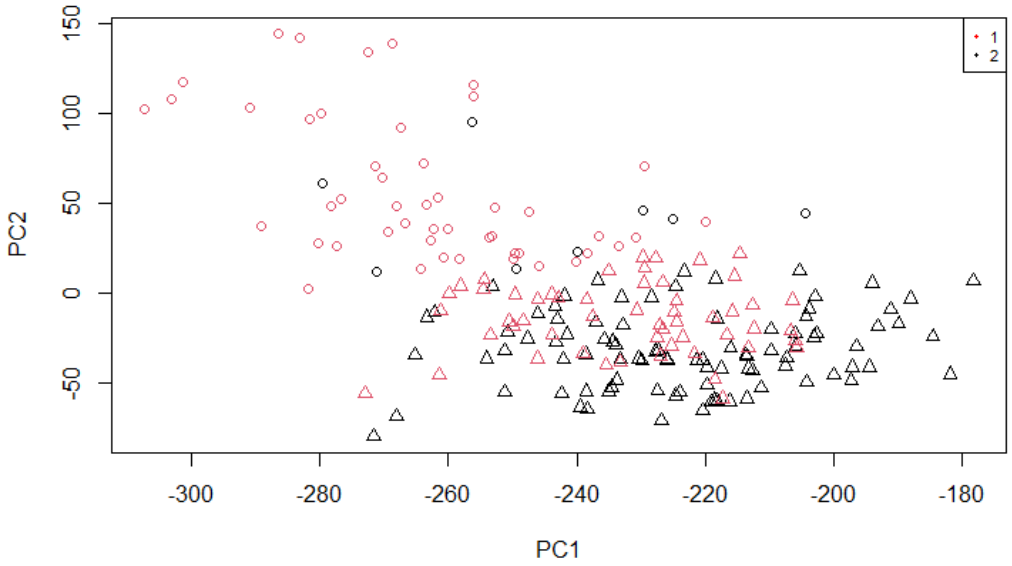


Rysunek 18: Wyniki 3D.

Możemy zauważyć, że dane są podzielone, w stosunku do osi PC3 jedne dane znajdują się „powyżej" drugich.

2.3 Klasteryzacja dla danych nieprzeskalowanych

Dokonujemy klasteryzacji metodą k-średnich. Wyniki przedstawiamy na wykresie.



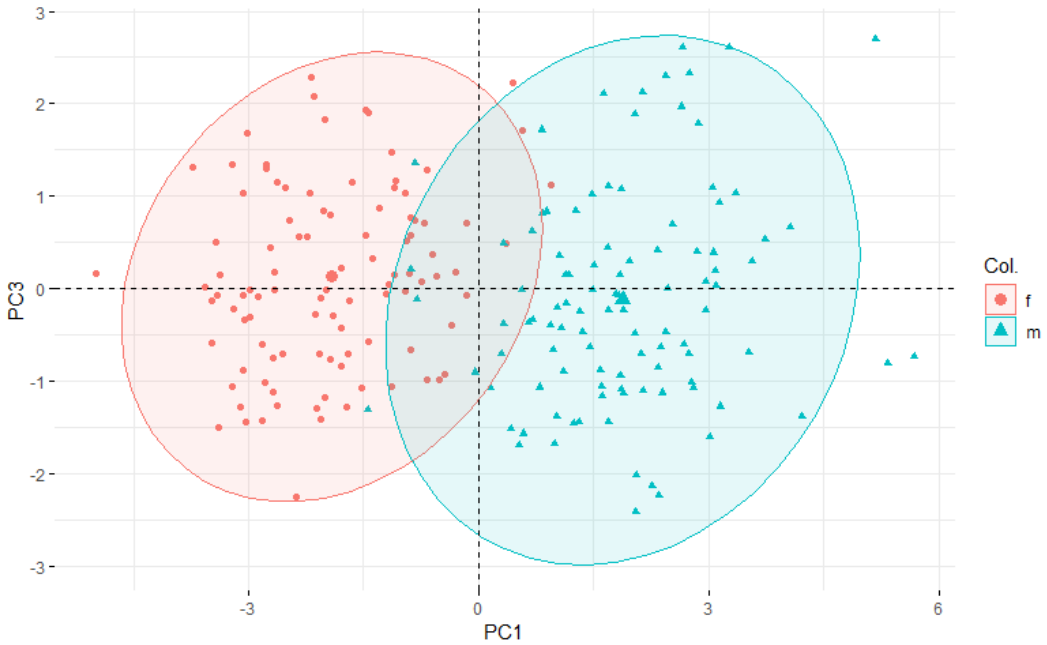
Rysunek 19: Wyniki klasteryzacji na danych nieskalowanych z rozróżnieniem na dwie grupy.

Otrzymane wyniki źle wyjaśniają zmienną *sex*, gdyż obserwacje mieszają się. Widzimy dużą liczbę niepoprawnie pogrupowanych obserwacji.

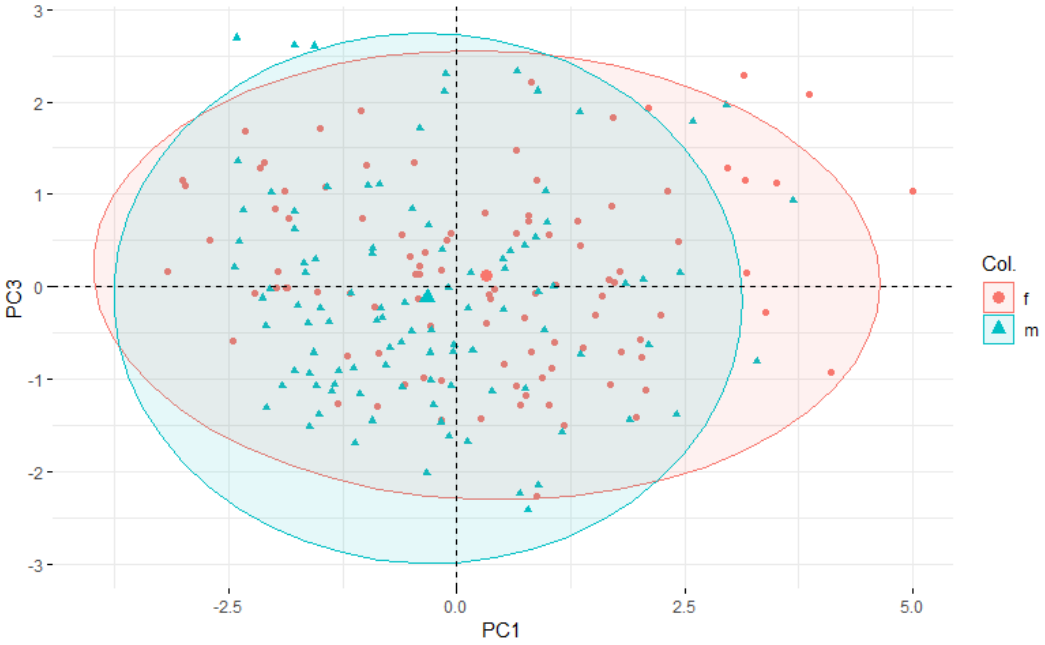
2.4 PCA na skalowanych danych

Wykonujemy skalowanie danych oraz analizę PCA. Wyniki dla pierwszych trzech składowych głównych obrazujemy na wykresach dwu- i trójwymiarowym, zaznaczamy kolorem podział danych względem płci.

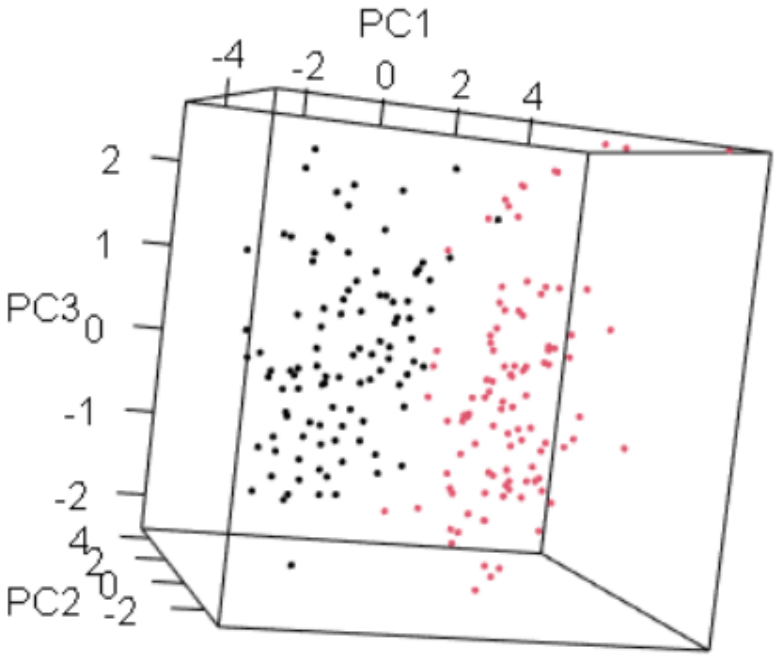
Rysunek 20: Wyniki dla PC1 i PC2.



Rysunek 21: Wyniki dla PC1 i PC3.



Rysunek 22: Wyniki dla PC2 i PC3.

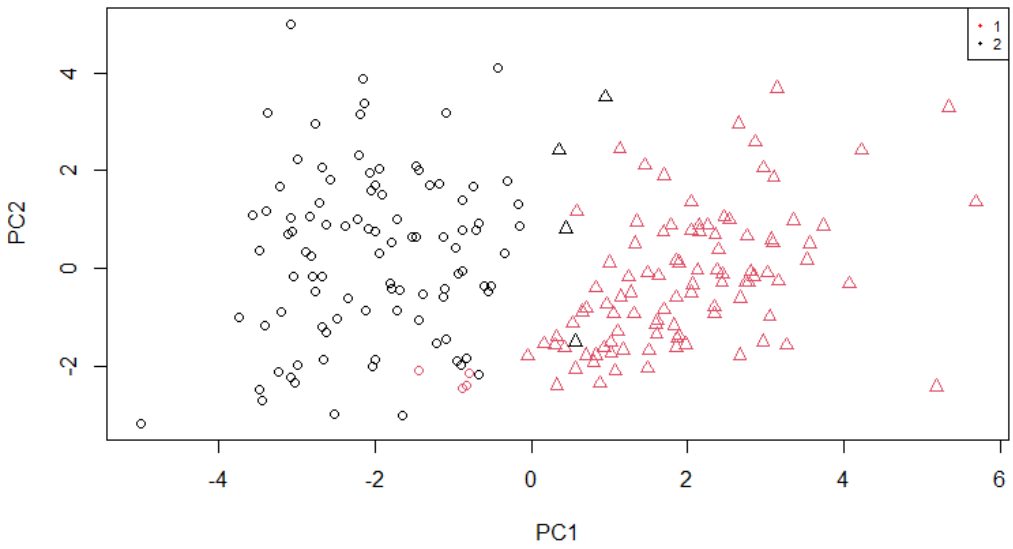


Rysunek 23: Wyniki 3D.

Możemy zauważyć grupowanie się danych na osiach PC1 i PC2, a także PC1 i PC3.

2.5 Klasteryzacja dla danych przeskalowanych

Dokonujemy klasteryzacji metodą k-średnich na danych przeskalowanych przed PCA. Wyniki przedstawiamy na wykresie.



Rysunek 24: Wyniki klasteryzacji na danych skalowanych. Kolor czerwony oznacza kobiety, kolor czarny oznacza mężczyzn. Trójkąt to jeden klaster, okrąg to klaster drugi.

Otrzymane wyniki dobrze wyjaśniają zmienną *sex*, gdyż obserwacje (w większości) nie mieszają się. Liczba niepoprawnych grupowań to 8.

3 Zadanie 3

3.1 Wczytanie ramki danych dla obrazu

Wczytujemy ramkę danych dla obrazka *papugi*. Wyświetlamy 6 pierwszych obserwacji.

```
> head(imgRGB)
  x   y         R         G         B
1 1 512 0.4549020 0.4549020 0.3450980
2 1 511 0.4784314 0.4666667 0.3607843
3 1 510 0.4941176 0.4823529 0.3686275
4 1 509 0.5137255 0.5058824 0.3882353
5 1 508 0.5294118 0.5215686 0.3960784
6 1 507 0.5529412 0.5333333 0.4196078
```

Rysunek 25: Zapisana ramka danych.

3.2 Klasteryzacja

Dokonujemy klasteryzacji za pomocą funkcji *kmeans()* dla $k = 2, 16, 64$.

3.3 Kodowanie kolorów obrazka

Tworzymy ramki danych z intensywnościami kolorów przypisanych do klastra (centroidami), która stanowi kod kolorowania.

```
> head(centra_2)
  nr_klastra         R         G         B
1          1 0.3884209 0.3504322 0.2324227
2          2 0.8003630 0.7190386 0.5334574
> head(centra_16)
  nr_klastra         R         G         B
1          1 0.3521552 0.5527653 0.5800362
2          2 0.1665110 0.1554144 0.1283518
3          3 0.5123482 0.1935051 0.1633770
4          4 0.8318242 0.7677439 0.7623583
5          5 0.2503699 0.2377185 0.1755174
6          6 0.6756207 0.6520569 0.6201072
> head(centra_64)
  nr_klastra         R         G         B
1          1 0.6734017 0.6615072 0.6822856
2          2 0.4230669 0.3955175 0.2938070
3          3 0.1776635 0.1703527 0.1537328
4          4 0.4440595 0.6631231 0.6858134
5          5 0.6244696 0.5739861 0.1773402
6          6 0.2778661 0.4244918 0.4294212
```

Rysunek 26: Pierwsze dane dla ramek z intensywnościami kolorów przypisanych do klastra.

Tworzymy ramkę danych z zakodowanymi w numerze klastra kolorami dla każdego piksela.

```
> head(img_kod_2)
  x   y nr_klastra
1 1 512          1
2 1 511          1
3 1 510          1
4 1 509          1
5 1 508          1
6 1 507          1
> head(img_kod_16)
  x   y nr_klastra
1 1 512         12
2 1 511          7
3 1 510          7
4 1 509         10
5 1 508         10
6 1 507         10
> head(img_kod_64)
  x   y nr_klastra
1 1 512         62
2 1 511         62
3 1 510         30
4 1 509         30
5 1 508         22
6 1 507         22
```

Rysunek 27: Pierwsze dane dla ramek z zakodowanymi w numerze klastra kolorami dla każdego piksela.

3.4 Dekodowanie obrazka

Łączymy stworzone tabele za pomocą funkcji *full_join()*. Następnie wykorzystując funkcję *array()* tworzymy z uzyskanych macierzy tablicę o wymiarze $N \times M \times 3$. Tak przygotowane tablice stanowi odkodowany skompresowany obraz. Wymiary każdego z obrazka to $512 \times 768 \times 3$. Na poniższych rys. przedstawiono uzyskane obrazki wraz z obrazkiem oryginalnym.

Rysunek 28: Obrazek dla $k = 2$.

Rysunek 29: Obrazek dla $k = 16$.

Rysunek 30: Obrazek dla $k = 64$.

Rysunek 31: Obrazek oryginalny.

W wyniku przeprowadzonych procedur udało się zkompresować obrazek *papugi*. Wyniki kompresji przedstawiono w tabeli 1.

Tabela 1: Wartości kompresji obrazka *papugi*.

obrazek	rozmiar
papugi oryginalny	550 KB
papugi $k = 2$	14 KB
papugi $k = 16$	79 KB
papugi $k = 64$	170 KB