

Raport 2

Eksploracja danych

Mikołaj Langner, Marcin Kostrzewa
nr albumów: 255716, 255749

2021-04-19

Spis treści

1	Wstęp	1
2	Zadanie 1	2
2.1	Wczytanie danych i wstępna analiza	2
2.2	Metody dyskretyzacji	2
2.3	Metody dyskretyzacji z wartościami odstającymi	6
3	Zadanie 2	11
3.1	Wczytanie i przygotowanie danych	11
3.2	Składowe główne i ich analiza	11
3.3	Wizualizacja danych	13
3.4	Korelacja zmiennych	14
3.5	Wnioski do zadania 2	15
4	Zadanie 3	16
4.1	Wybrany zbiór danych	16
4.2	Redukcja wymiaru na bazie MDS i analiza jej jakości	17
4.3	Wizualizacja danych	20

1 Wstęp

Sprawozdanie zawiera rozwiązanie zadań z listy 2.

Zadanie pierwsze dotyczy pojęcia dyskretyzacji i badania jego jakości.

Zadanie drugie i trzecie dotykają pojęcia metod redukcji wymiaru:

- w zadaniu drugim skorzystamy z metody składowych głównych,
- w zadaniu trzecim z metody skalowania wielowymiarowego.

2 Zadanie 1

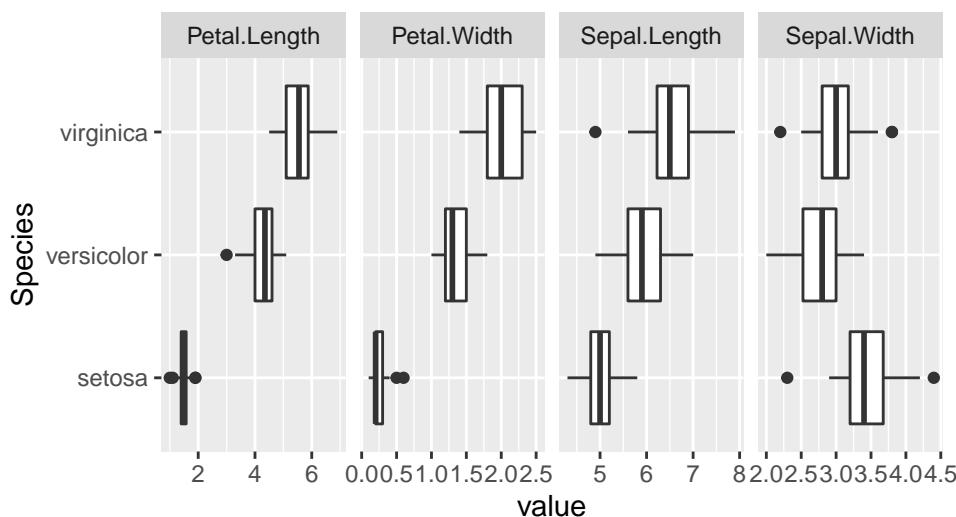
W pierwszym zadaniu mamy dokonać dyskretyzacji cech ciągłych ze zbioru `iris` i ocenić jej jakość.

2.1 Wczytanie danych i wstępna analiza

```
data(iris)
```

Wybierzmy zmienne o najlepszej i najgorszej zdolności dyskryminacyjnej. W tym celu narysujemy wykresy pudełkowe oraz wyliczymy współczynniki zmienności każdej ze zmiennych z podziałem na poszczególne gatunki irysów i porównamy ich rozkłady.

```
plot_boxplot(iris, by="Species")
```



Rysunek 1: Wykresy pudelkowe dla zmiennych ze zbioru iris

Tabela 1: Współczynniki zmienności dla poszczególnych zmiennych

Species	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
setosa	0.070	0.111	0.119	0.428
versicolor	0.087	0.113	0.110	0.149
virginica	0.097	0.108	0.099	0.136

Możemy zauważyć, że zmienna Petal.Length najefektywniej rozdziela poszczególne gatunki, natomiast zmienna Sepal.Width radzi sobie z tym najgorzej.

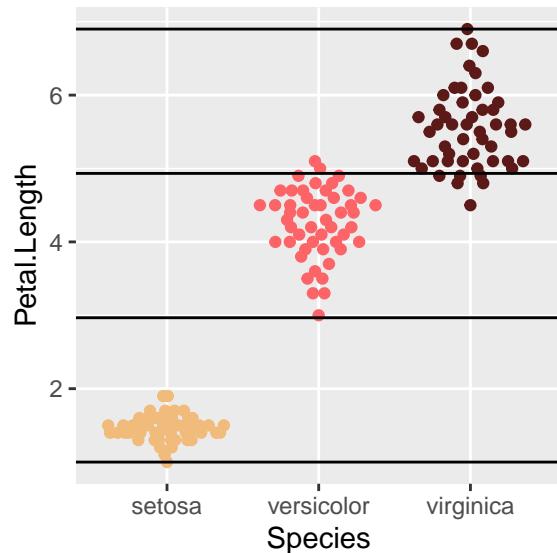
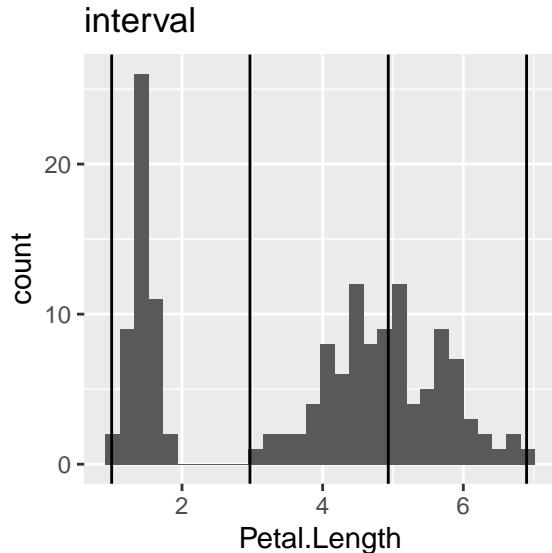
2.2 Metody dyskretyzacji

Porównamy ze sobą cztery metody dyskretyzacji nienadzorowanej:

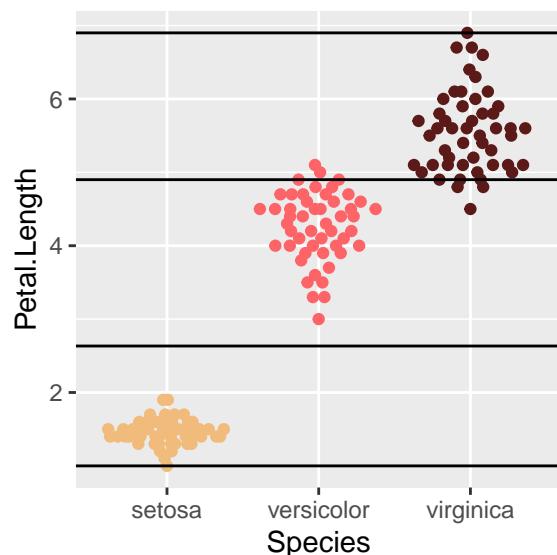
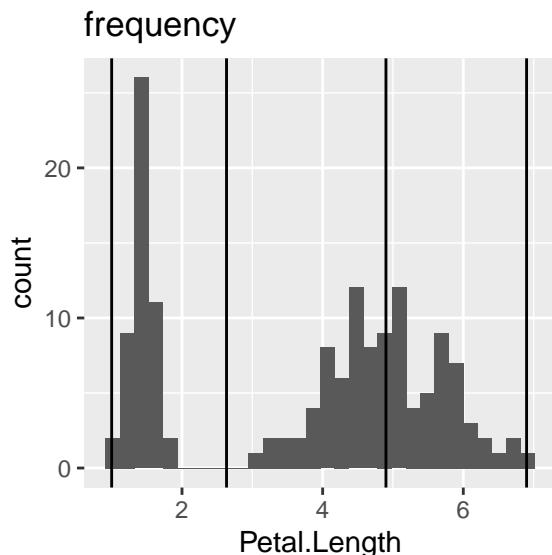
- equal width,
- equal frequency,
- k-means clustering,
- dyskretyzacje dla przedziałów zadanych przez użytkownika.

2.2.1 Najlepiej separująca zmienna

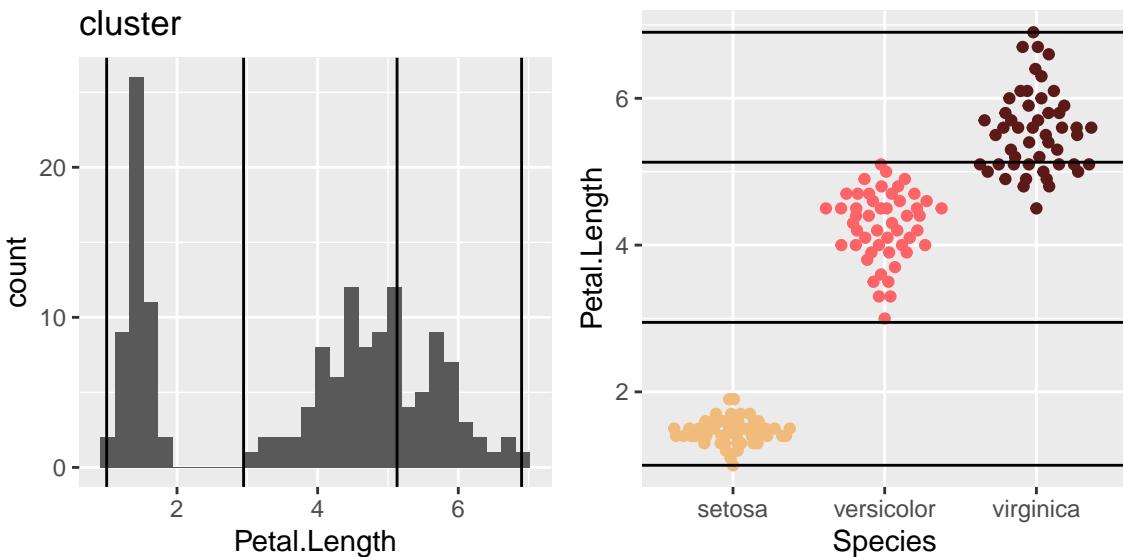
Zacznijmy od zmiennej Petal.Length, która najlepiej rozdziela poszczególne gatunki irysów.



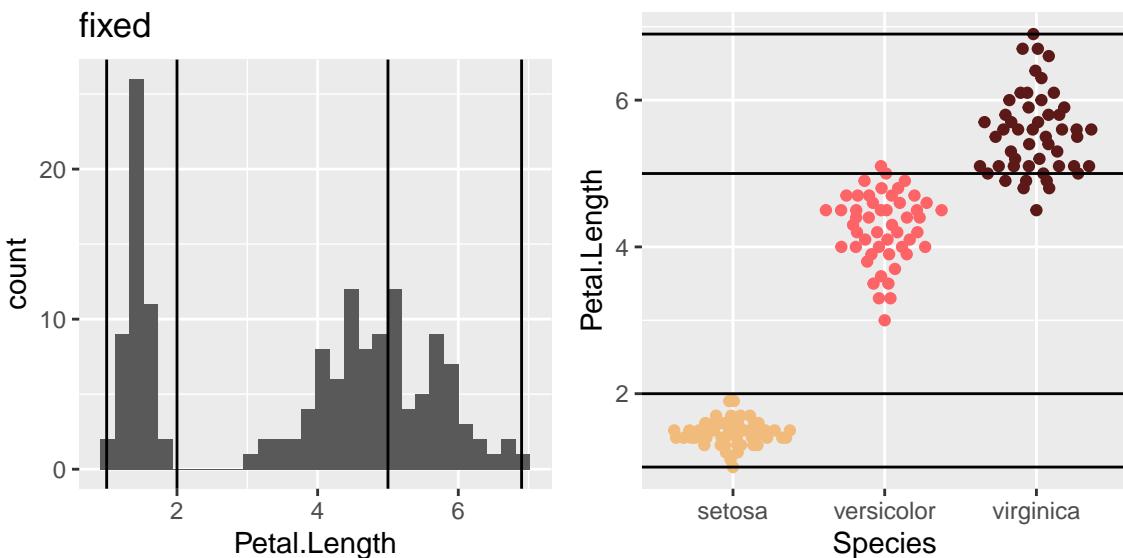
```
## Cases in matched pairs: 94.67 %
```



```
## Cases in matched pairs: 95.33 %
```



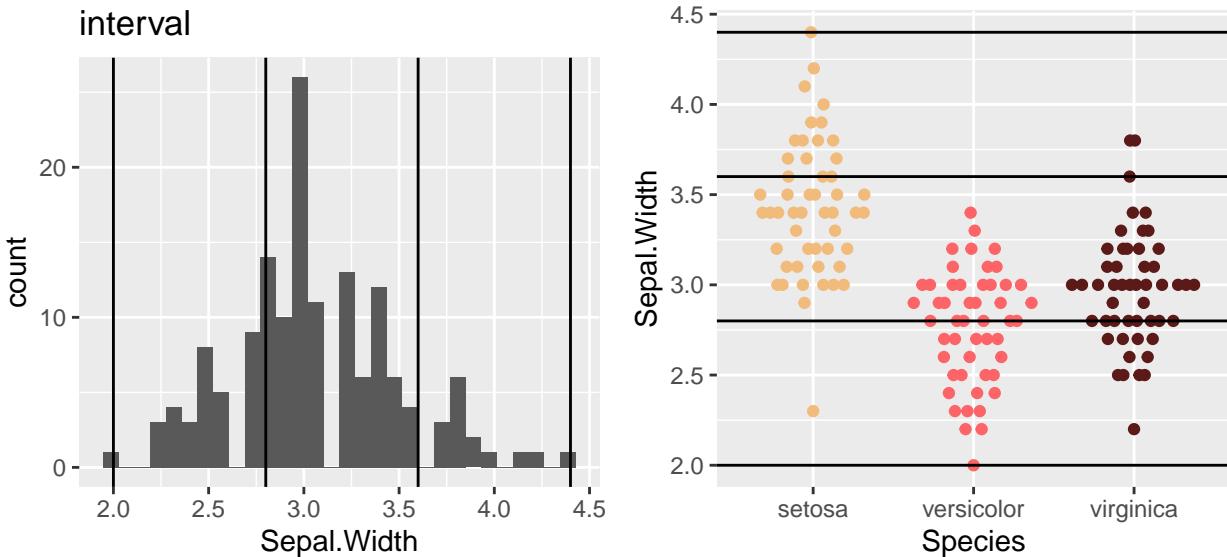
```
## Cases in matched pairs: 89.33 %
```



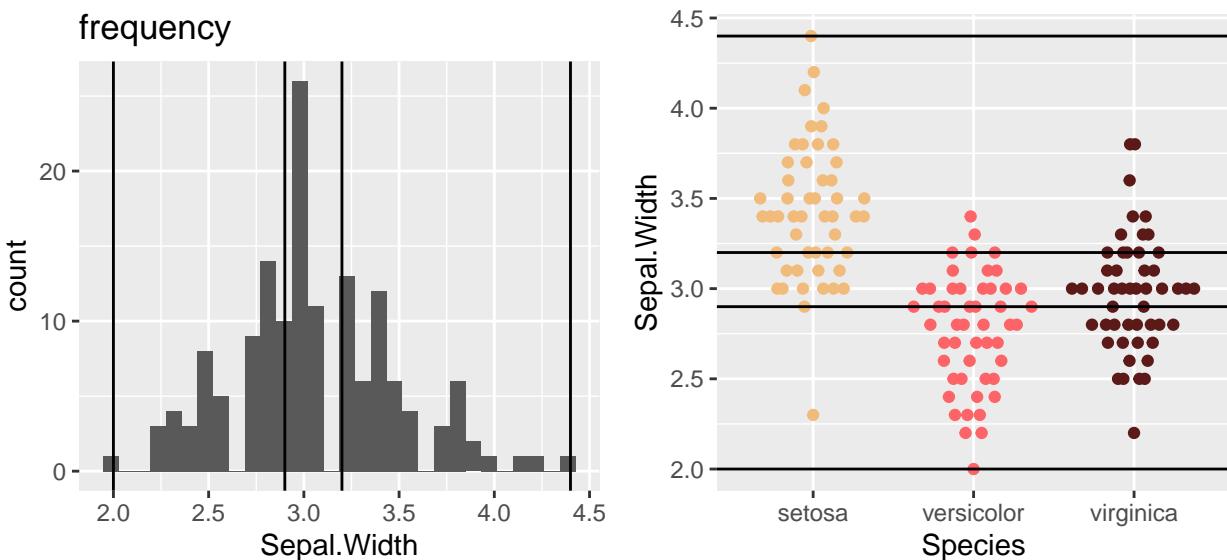
```
## Cases in matched pairs: 94.67 %
```

2.2.2 Najgorzej separująca zmienna

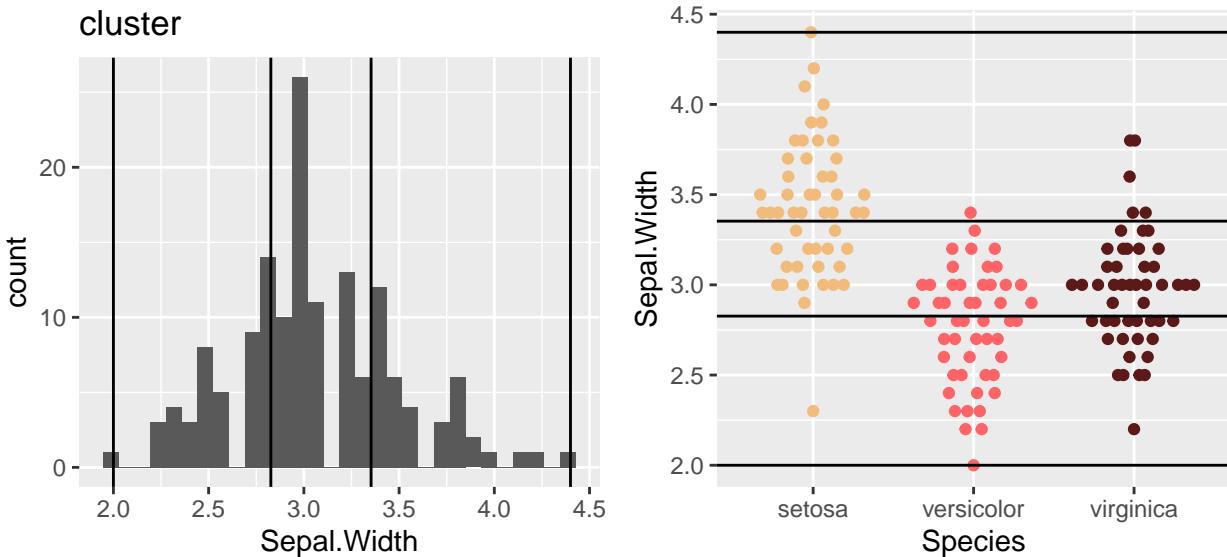
Możemy zobaczyć teraz jak poszczególne metody działają dla zmiennej Sepal.Width, która najgorzej radzi sobie z rozdzieleniem gatunków.



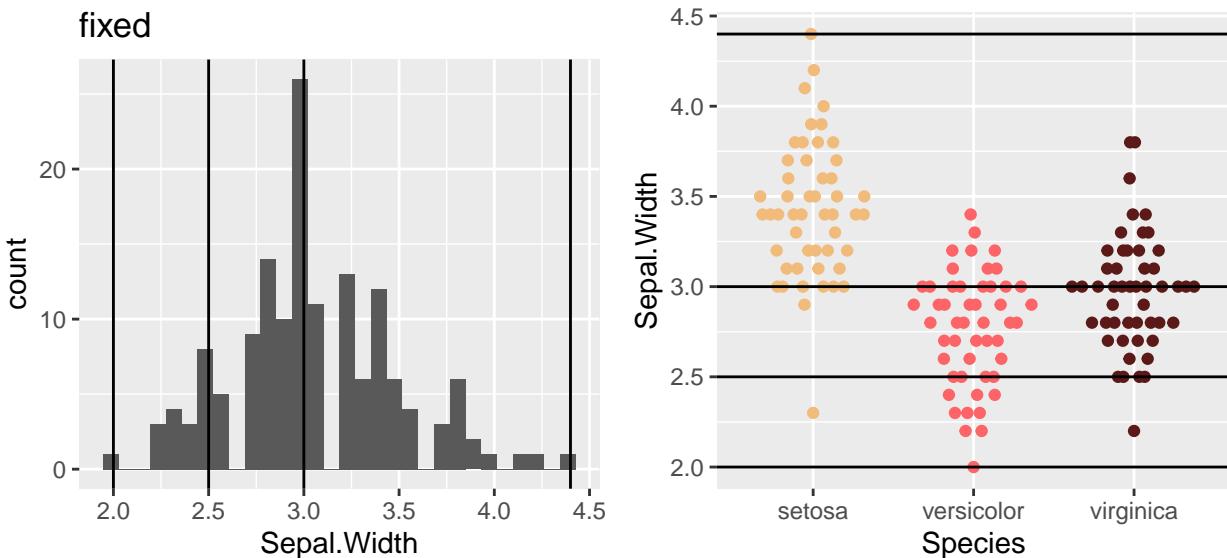
Cases in matched pairs: 50.67 %



Cases in matched pairs: 55.33 %



```
## Cases in matched pairs: 56 %
```



```
## Cases in matched pairs: 54.67 %
```

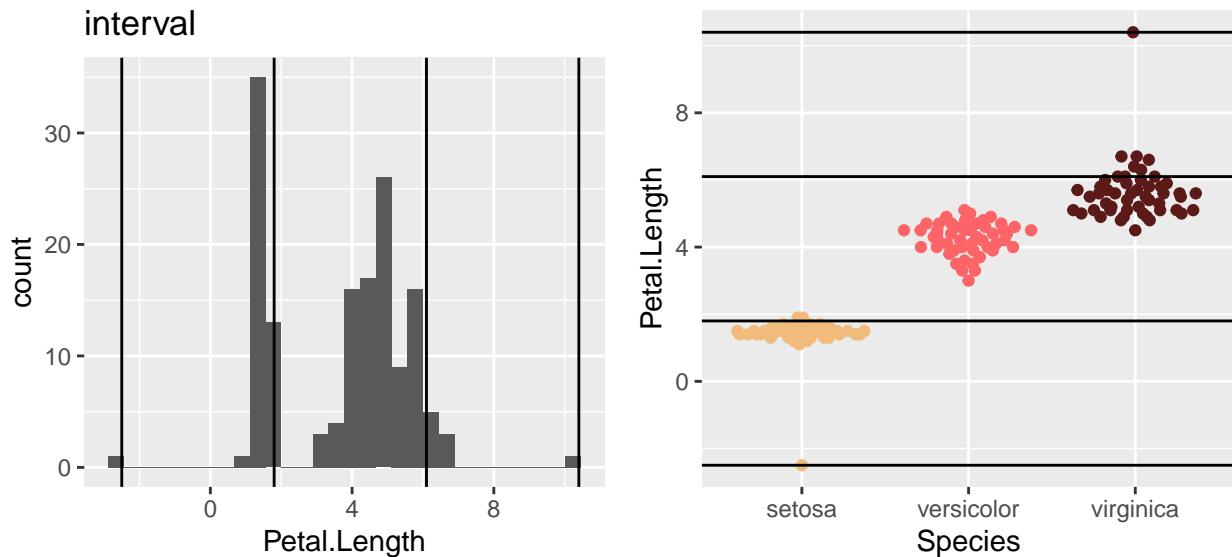
Dla obu zmiennych każda z metod wypada równie dobrze, przy czym, najlepsze wyniki dają metody równej częstości oraz k-średnich.

2.3 Metody dyskretyzacji z wartościami odstającymi

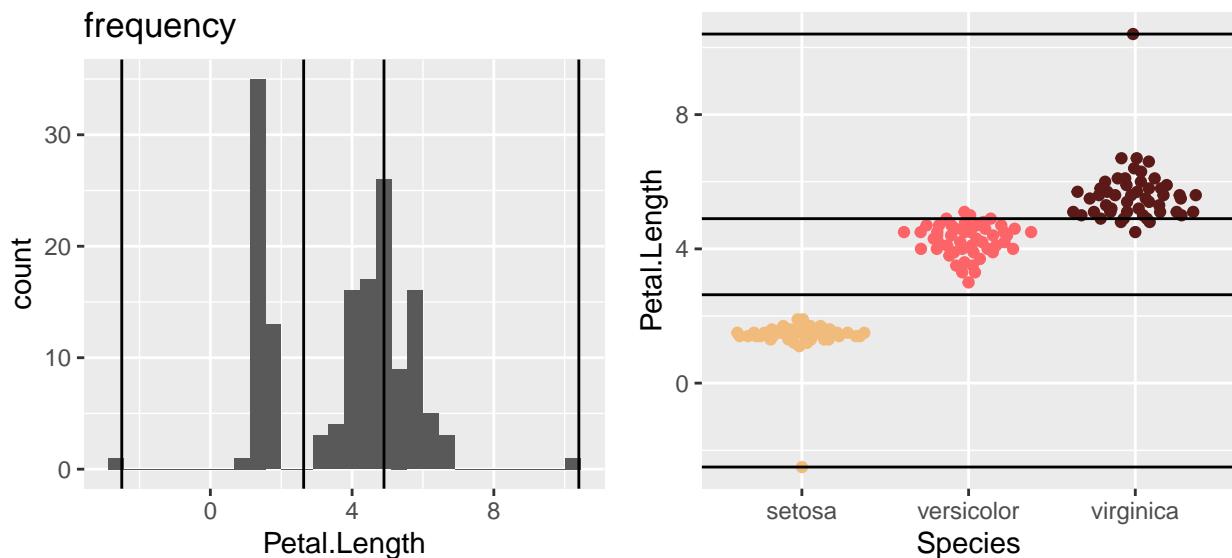
Rozpatrzmy teraz dyskretyzację przy dodaniu sztucznie wartości odstających.

2.3.1 Zmienna Petal.Length

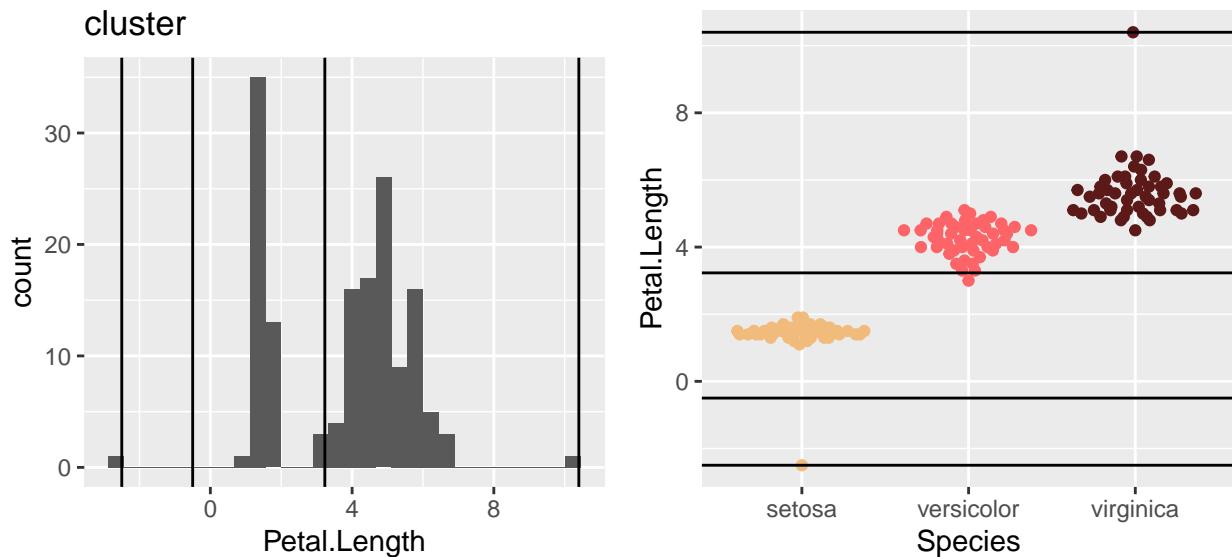
Zacznijmy znowu od zmiennej Petal.Length.



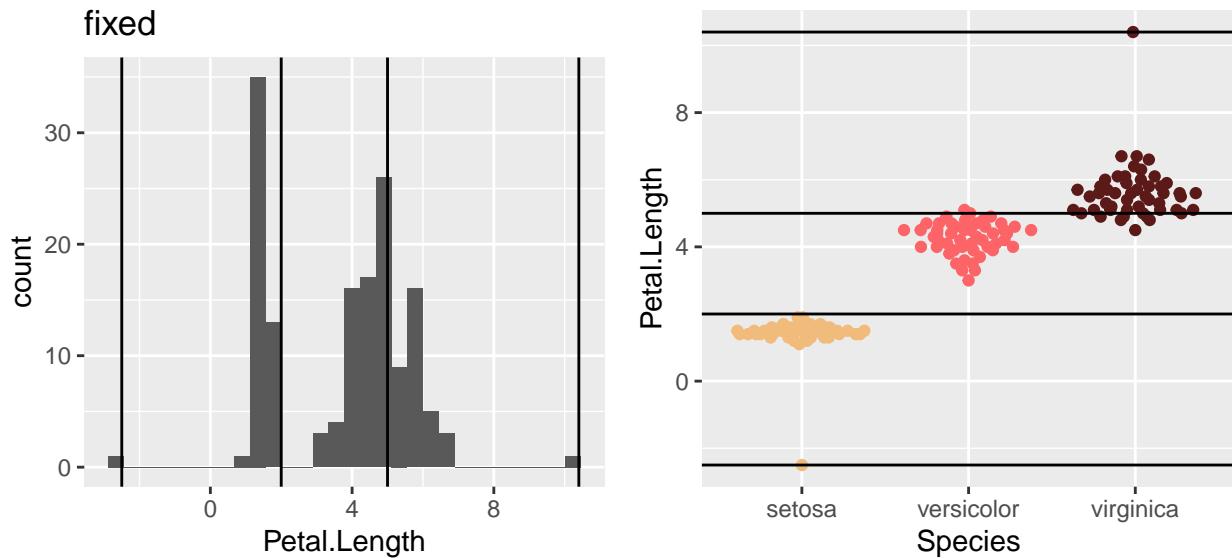
```
## Cases in matched pairs: 71.33 %
```



```
## Cases in matched pairs: 95.33 %
```



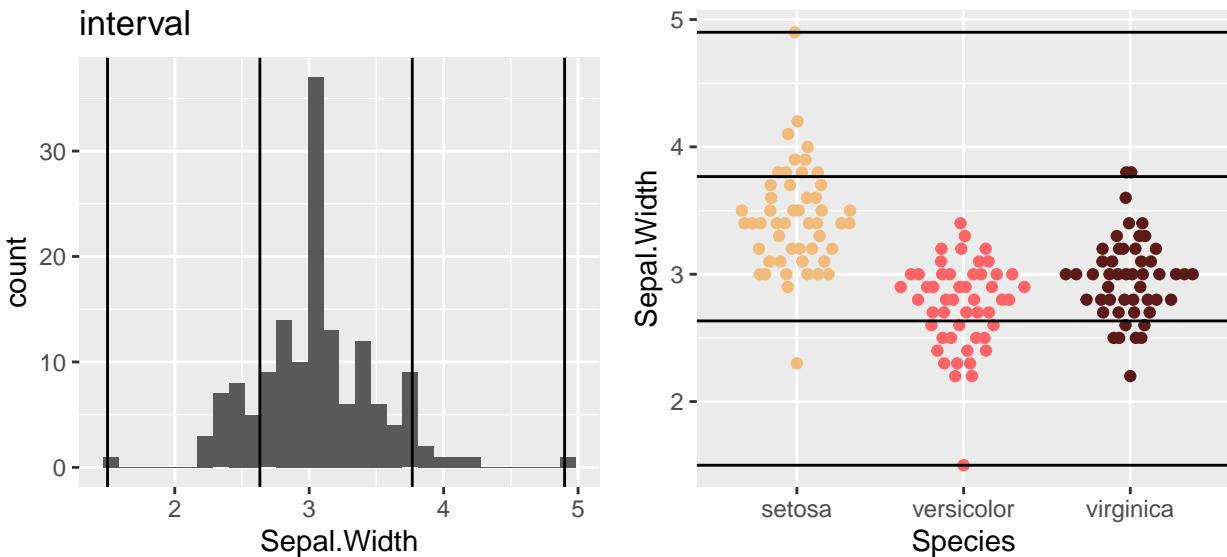
```
## Cases in matched pairs: 66.67 %
```



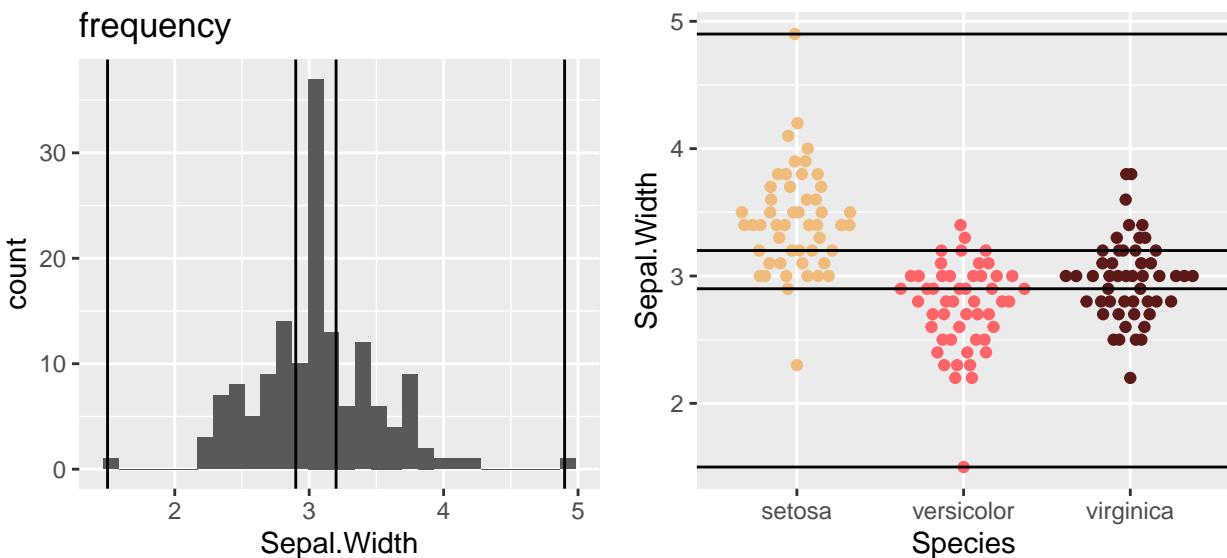
```
## Cases in matched pairs: 94.67 %
```

2.3.2 Zmienna Sepal.Width

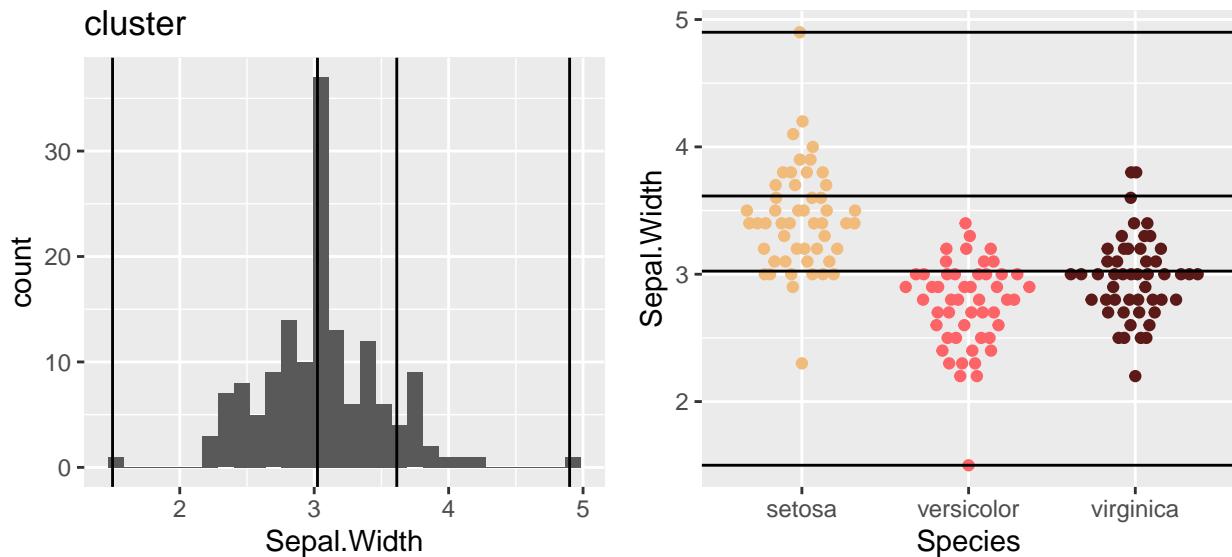
Dla zmiennej Sepal.Width po dodaniu wartości odstających dyskretyzacja wygląda następująco:



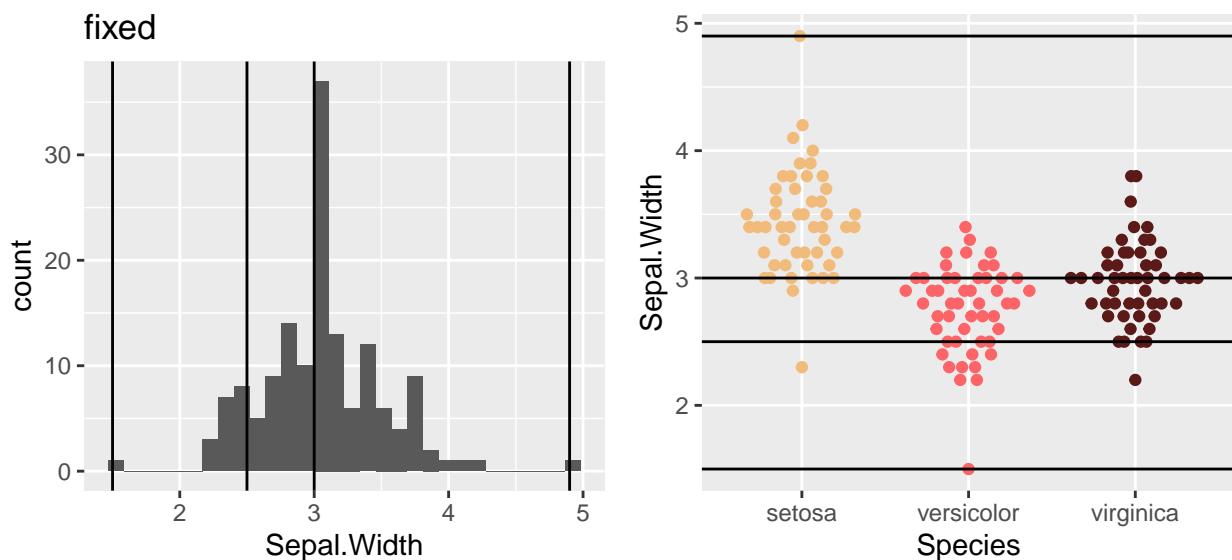
Cases in matched pairs: 44.67 %



Cases in matched pairs: 55.33 %



```
## Cases in matched pairs: 56 %
```



```
## Cases in matched pairs: 54.67 %
```

Nie powinien dziwić fakt, że największa zmiana w poprawności predykcji dotknęła metodę przedziałów równej długości, gdyż pojedyncza obserwacja całkowicie zmienia dobór miejsc partycji przedziału.

3 Zadanie 2

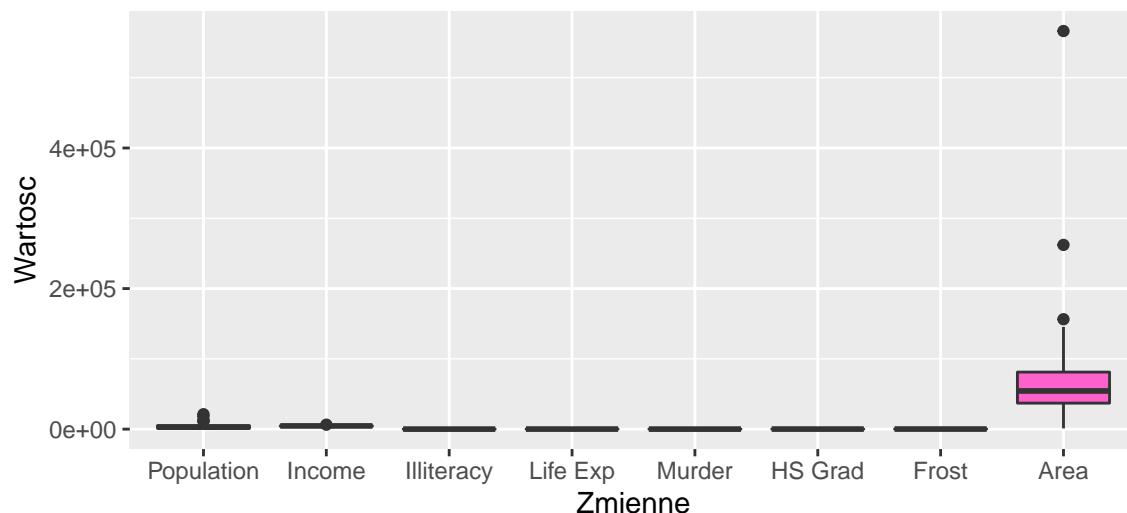
3.1 Wczytanie i przygotowanie danych

Teraz naszym zadaniem jest dokonanie analizy składów głównych (PCA) dla zbioru `state.x77`, który zawiera informacje o wskaźnikach terytorialno-społecznych dla wszystkich amerykańskich stanów.

Najpierw wczytajmy dane i uzupełnijmy je o informacje geograficzne.

```
data(state)
state <- as.data.frame(state.x77)
state$region <- state.region
state$division <- state.division
state.subset <- subset(state, select=-c(region, division))
```

By rozstrzygnąć, czy potrzebna jest normalizacja danych, przeanalizujemy wykresy pudełkowe oraz wyznaczmy odchylenia standardowe i współczynniki zmienności.



Rysunek 2: Wykresy pudełkowe dla zmiennych ze zbioru state.x77

Tabela 2: Odchylenie standardowe i współczynnik zmienności dla zmiennych

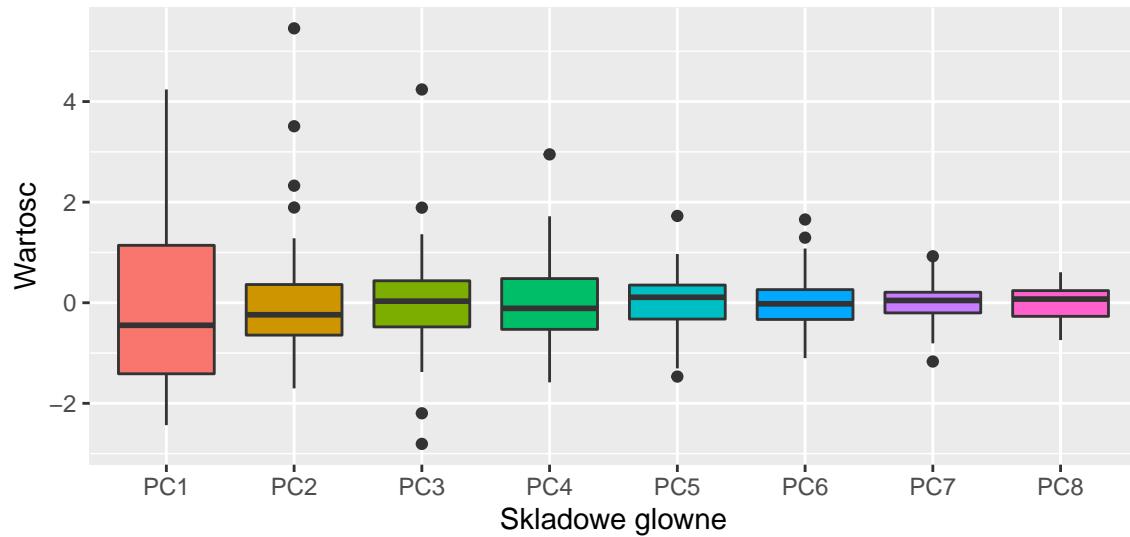
	Population	Income	Illiteracy	Life Exp	Murder	HS Grad	Frost	Area
Odchylenie standardowe	4464.491	614.470	0.610	1.342	3.692	8.077	51.981	85327.300
Współczynnik zmienności	1.051	0.139	0.521	0.019	0.500	0.152	0.498	1.206

Widać, że zmienne wymagają standaryzacji — ich wariancje zbyt mocno się różnią.

3.2 Składowe główne i ich analiza

Wyznaczymy teraz składowe główne i przedstawimy ich rozrzut, wykorzystując wykresy pudełkowe.

```
after.pca <- prcomp(state.subset, retx = T, center = T, scale. = T)
```



Rysunek 3: Wykresy pudelkowe dla składowych głównych

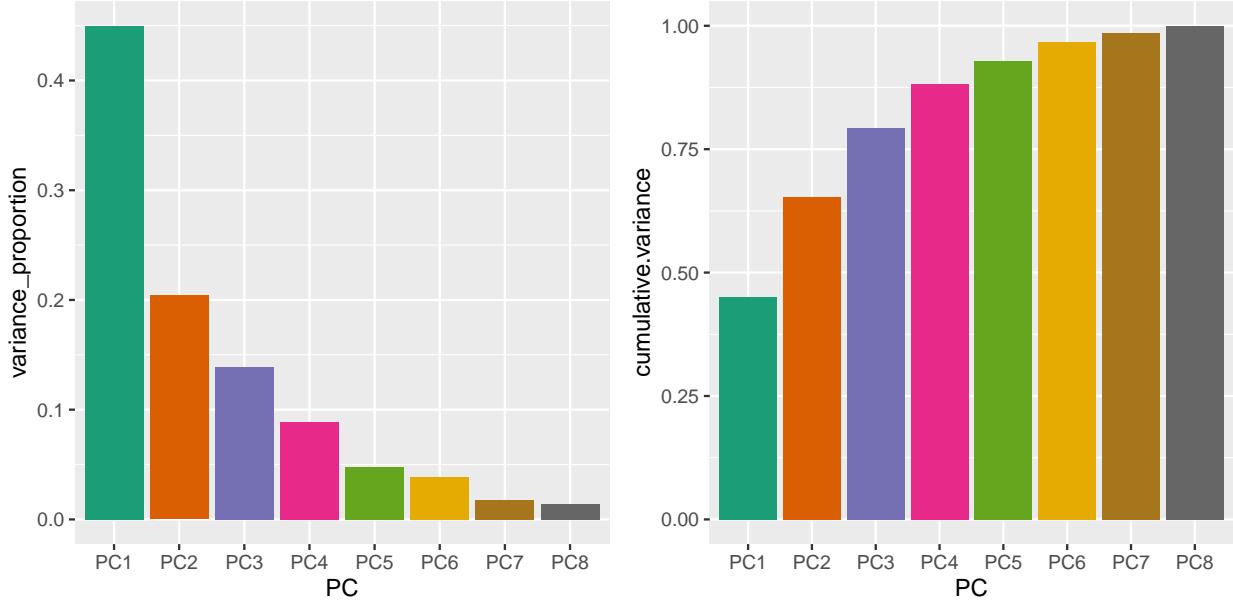
Przypatrzmy się teraz wektorom ładunków dla trzech pierwszych składowych głównych.

Tabela 3: Wektory ladunkow dla trzech pierwszych PC

	PC1	PC2	PC3
Population	0.126	0.411	-0.656
Income	-0.299	0.519	-0.100
Illiteracy	0.468	0.053	0.071
Life Exp	-0.412	-0.082	-0.360
Murder	0.444	0.307	0.108
HS Grad	-0.425	0.299	0.050
Frost	-0.357	-0.154	0.387
Area	-0.033	0.588	0.510

- W przypadku pierwszej składowej głównej, największy wkład mają zmienne **Illiteracy**, **Murder**, **HS Grad** i **Life Exp**. Dwie pierwsze mają ten sam znak, możemy więc wnioskować, że są ze sobą powiązane. **HS Grad** i **Life Exp** mają znak przeciwny — stąd te dwie pary są ze sobą negatywnie skorelowane. Jest to dość oczywisty rodzaj zależności między stopniem analfabetyzmu a procentem ilości mających ukończoną szkołę średnią i między ilością morderstw a średnią długością życia.
- W przypadku drugiej składowej głównej, największą wagę mają zmienne **Area**, **Population** i **Income**. Zależność między **Area** a **Population** jest dość oczywista, natomiast zależność tych zmiennych od **Income** już niekoniecznie da się łatwo wy tłumaczyć.

Zbadajmy teraz jaka część wyjaśnionej wariancji odpowiada kolejnym składowym głównym.



Rysunek 4: Wariancja wyjaśniana przez poszczególne składowe główne i wariancja skumulowana.

Tabela 4: Procent wyjaśnianej wariancji i skumulowana wariancja

	PC1	PC2	PC3	PC4	PC5
Proporcja wariancji	0.45	0.204	0.139	0.088	0.048
Skumulowana wariancja	0.45	0.654	0.793	0.881	0.929

Zauważamy, że:

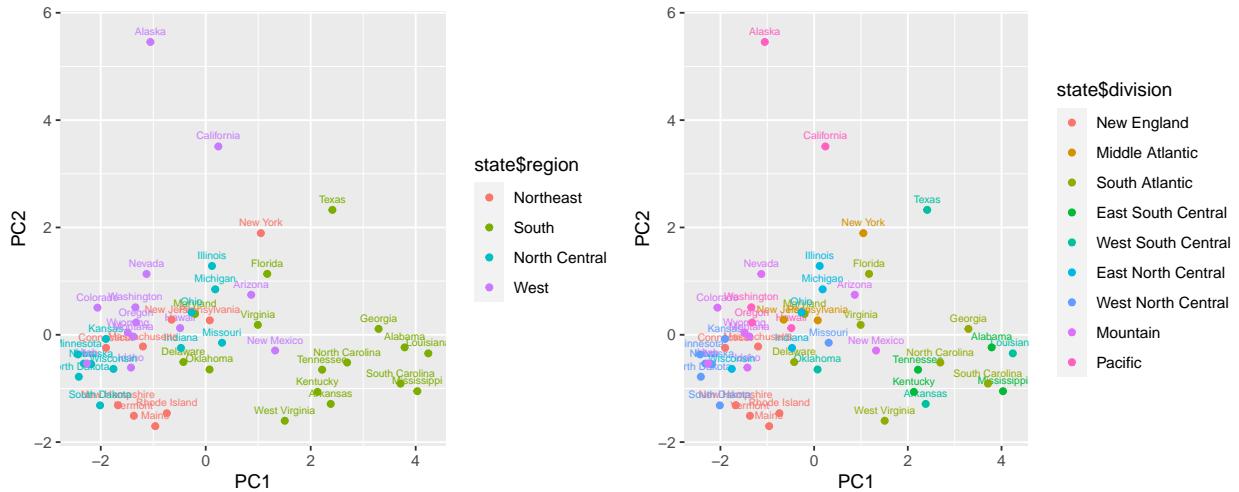
- PC1 wyjaśnia 45% wyjaśnianej wariancji, PC2 prawie 25%;
- 80% całkowitej wariancji jest wyjaśniane przez pierwsze cztery składowe główne (trzy pierwsze wyjaśniają niewiele mniej), 90% jest wyjaśniane zaś przez pierwszych 5.

3.3 Wizualizacja danych

W tej części wygenerujemy dwuwymiarowe wykresy rozrzutu (5) dla dwóch pierwszych składowych głównych. Skorzystamy z danych dotyczących lokalizacji poszczególnych stanów, by być w stanie wyciągnąć interesujące wnioski.

Obserwacje:

- Stany zlokalizowane w południowych częściach USA są stosunkowo blisko względem siebie położone — możemy więc wnioskować o ich dużym podobieństwie. Są one też często dość oddalone od pozostałych obserwacji.



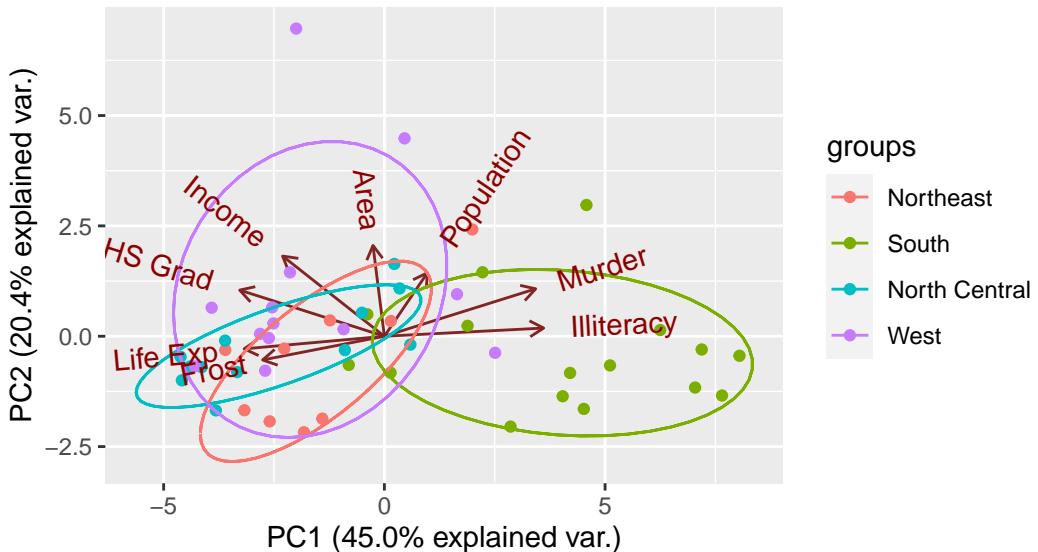
Rysunek 5: Wykresy rozrzutu

- Są dwie obserwacje, które znaczaco różnią się od pozostałych. Są to Alaska i Kalifornia. Alaska jest stanem o największej powierzchni, dochód na jednego mieszkańców jest tam również najwyższy. Natomiast w Kalifornii mieszka najwięcej ludzi (stan ten charakteryzuje się także dużą powierzchnią).

Przygotowaliśmy także wykresy 3d — kod umieściliśmy w dodatkowym skrypcie.

3.4 Korelacja zmiennych

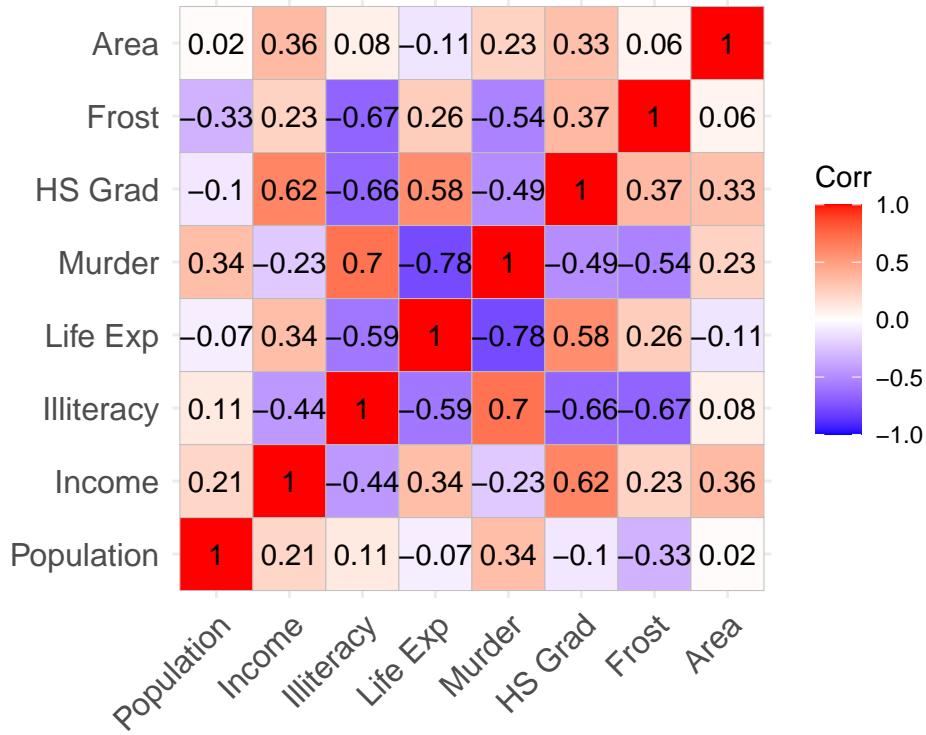
Zbadamy teraz korelację między zmiennymi. Najpierw skorzystamy z dwuwykresu.



Rysunek 6: Dwuwykres dla danych state.x77

Możemy zaobserwować, że zmienne **Murder** i **Illiteracy** są ze dodatnio i silnie skorelowane. Podobnie zachowują się zmienne **Income** i **HS Grad**. Ujemna korelacja jest możliwa do zaobserwowania pomiędzy zmiennymi **Life Exp** i **Murder**, **HS Grad** i **Illiteracy**. Zmienna **Frost** jest ujemnie skorelowane z **Illiteracy** i **Murder**.

Te wnioski potwierdzają się, jeżeli popatrzymy na mapę ciepła korelacji zmiennych.



Rysunek 7: Mapa ciepła korelacji zmiennych

3.5 Wnioski do zadania 2

Dzięki zastosowaniu metody analizy składowych głównych udało się nam otrzymać ciekawe wnioski dotyczące stanów USA.

Przede wszystkim, stany zlokalizowane na południu kraju są bardzo do siebie podobne. Charakteryzują się największym stopniem analfabetyzmu, największą ilością morderstw, najniższym stopniem wykształcenia. Są to też na ogół cieplejsze stany, co tłumaczy ujemną korelację pomiędzy zmiennymi **Murder**, **Illiteracy** a **Frost**.

Inne stany (poza dwoma wyjątkami — Alaską i Kalifornią) znajdująły się dość blisko siebie. Stąd trudno o ogólne wnioski na temat znaczących różnic pomiędzy nimi.

4 Zadanie 3

4.1 Wybrany zbiór danych

Wybranym przez nas zbiorem danych jest **Stars**, który dostępny jest na Kaggle'u (link). Zawiera on 240 obserwacji i 7 następujących zmiennych:

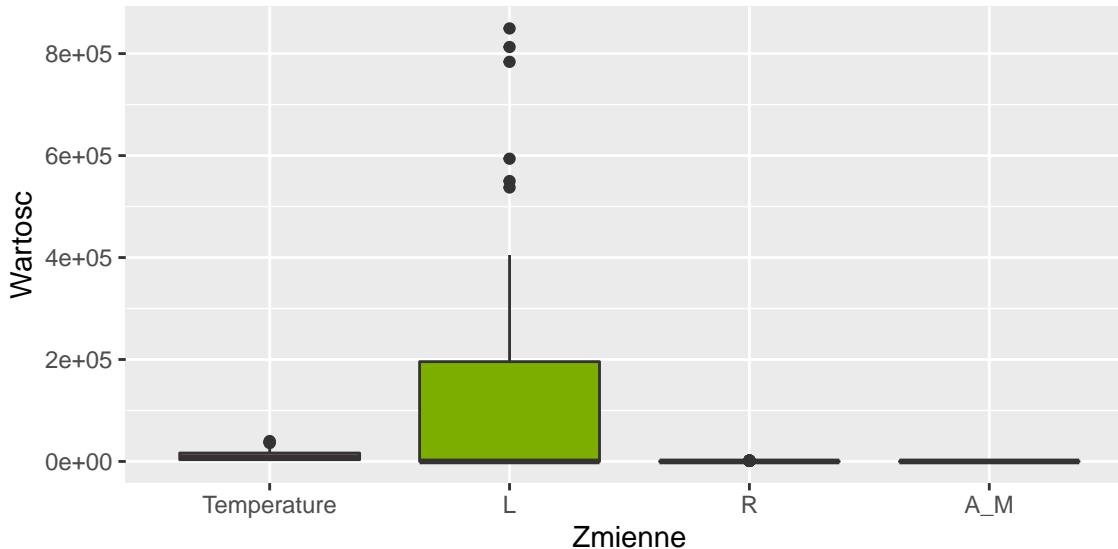
- **Temperature** — temperatura gwiazdy wyrażona w Kelwinach (ilościowa),
- **L** — stosunek jasności gwiazdy i jasności Słońca (ilościowa),
- **R** — stosunek promienia gwiazdy do promienia Słońca (ilościowa),
- **A_M** — absolutna wielkość gwiazdowa (ilościowa),
- **Color** — kolor gwiazdy (jakościowa),
- **Spectral_Class** — typ spektralny gwiazdy (jakościowa),
- **Type** — typ gwiazdy — jedno z: czerwony, biały, brązowy karzeł, gwiazda ciągu głównego, nadolbrzym, hiperolbrzym (jakościowa).

Zmienną **Type** będziemy traktować jako zmienną grupującą.

Wczytamy teraz dane (z całego zbioru wybieramy losowo 100 obserwacji), przygotujemy je do skalowania wielowymiarowego i wyznaczmy macierz niepodobieństwa.

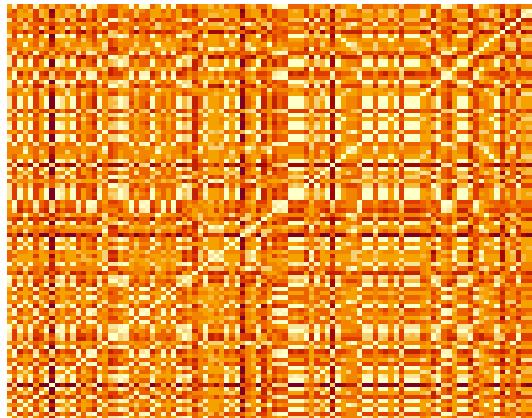
```
set.seed(42)
stars <- sample_n(read.csv("Stars.csv"), 100)
```

Przy wyznaczaniu odległości między obserwacjami za pomocą funkcji **daisy** dane poddajemy standaryzacji. O konieczności tej operacji świadczą wykresy pudełkowe zmiennych ciągłych.



Rysunek 8: Wykresy pudelkowe dla zmiennych ciągłych ze zbioru Stars

Poniżej także wygenerowaliśmy mapę ciepła macierzy niepodobieństwa.



Rysunek 9: Mapa ciepła macierzy niepodobienstwa

4.2 Redukcja wymiaru na bazie MDS i analiza jej jakości

Porównamy teraz jakość odwzorowania MDS w zależności od wielkości wymiaru d przestrzeni docelowej. Przedstawimy na wykresie wartości funkcji STRESS, jak i wykonamy diagramy Sheparda. Porównamy ze sobą skalowanie klasyczne (funkcja `cmdscale`) i niemetryczne — wykorzystujące metrykę Kruskala-Sheparda (funkcja `isoMDS`).

```
d.max <- 6

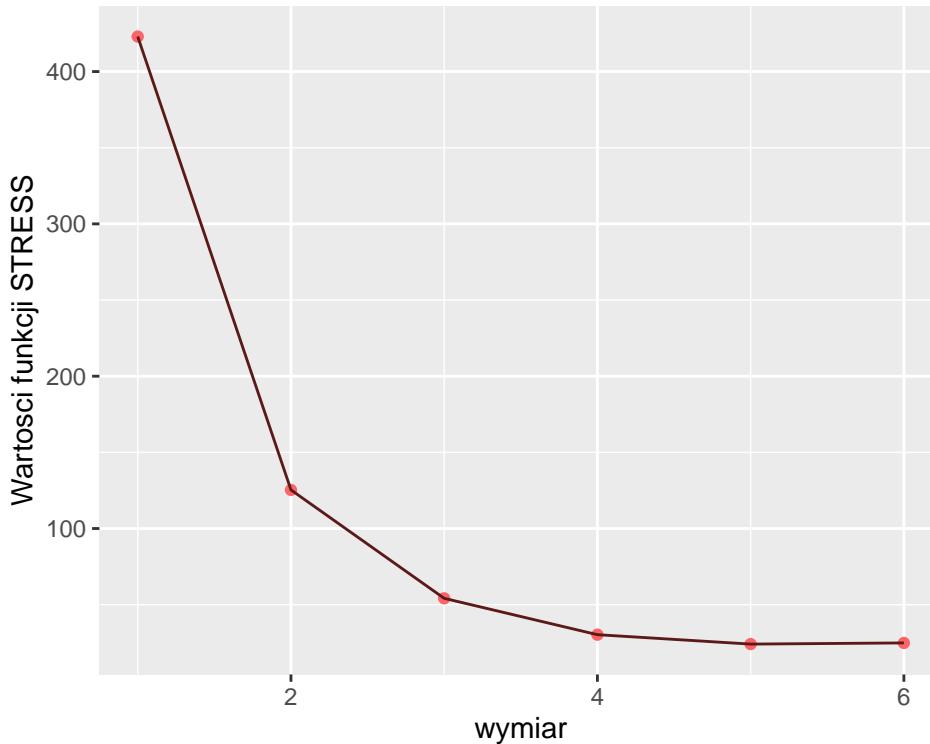
data.for.shepard <- data.frame('original' = as.vector(dist.matrix))
STRESS.values <- numeric(d.max)

for (i in 1:d.max) {
  mds.k <- as.matrix(cmdscale(dist.matrix, k = i))
  new.dist.matrix <- as.matrix(dist(mds.k, method = "euclidean"))

  STRESS.values[i] <- sum((new.dist.matrix - dist.matrix)^2)

  data.for.shepard[paste0("d=", i)] = as.vector(new.dist.matrix)
}
```

Poniżej znajduje się wykres funkcji STRESS dla skalowania klasycznego i odpowiadające mu diagramy Sheparda (11).



Rysunek 10: Wykres funkcji STRESS dla skalowania klasycznego

Przyglądając się diagramom Sheparda, możemy dojść do wniosku, że klasyczne MDS nie działa dla naszych danych idealnie. Świadczy o tym również wartość funkcji STRESS dla maksymalnego wymiaru $d.\max = 6 = 24.9$. Na podstawie tych wykresów możemy też stwierdzić, że optymalnym wyborem wymiaru byłoby $d = 4$, dla którego wartość funkcji stresu wynosi 30.3 — niewiele mniej niż dla $d.\max$.

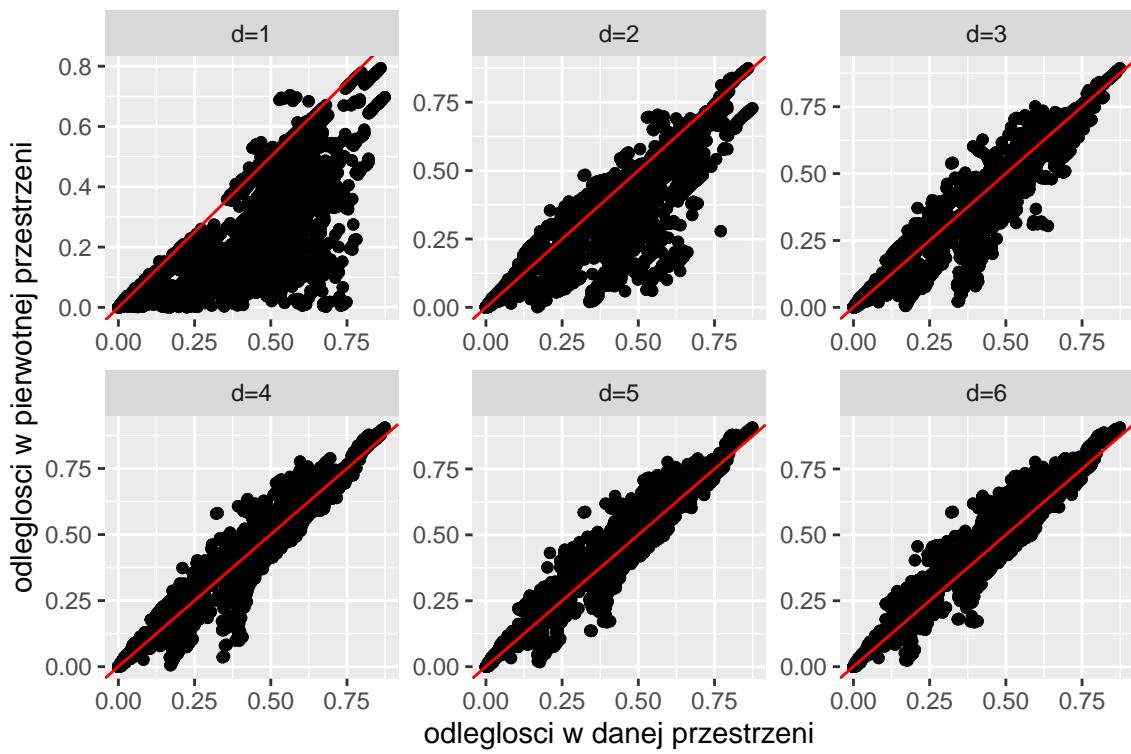
Teraz powtórzmy te czynności dla skalowania Kruskala-Sheparda — będą to wykresy (12) i (13).

```
kruskal.stress <- numeric(d.max)
data.for.shepard.2 <- data.frame('original' = as.vector(dist.matrix))

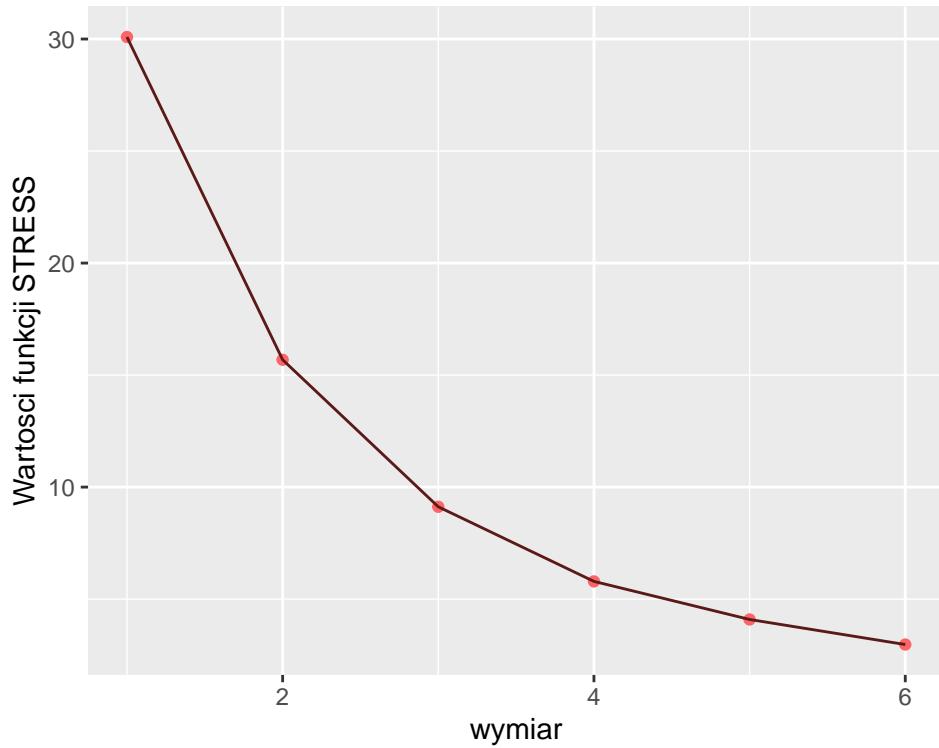
for (i in 1:d.max) {
  mds.k <- isoMDS(dist.matrix, k = i, trace = FALSE)
  new.dist.matrix <- as.matrix(dist(mds.k$points, method = "euclidean"))

  kruskal.stress[i] <- mds.k$stress

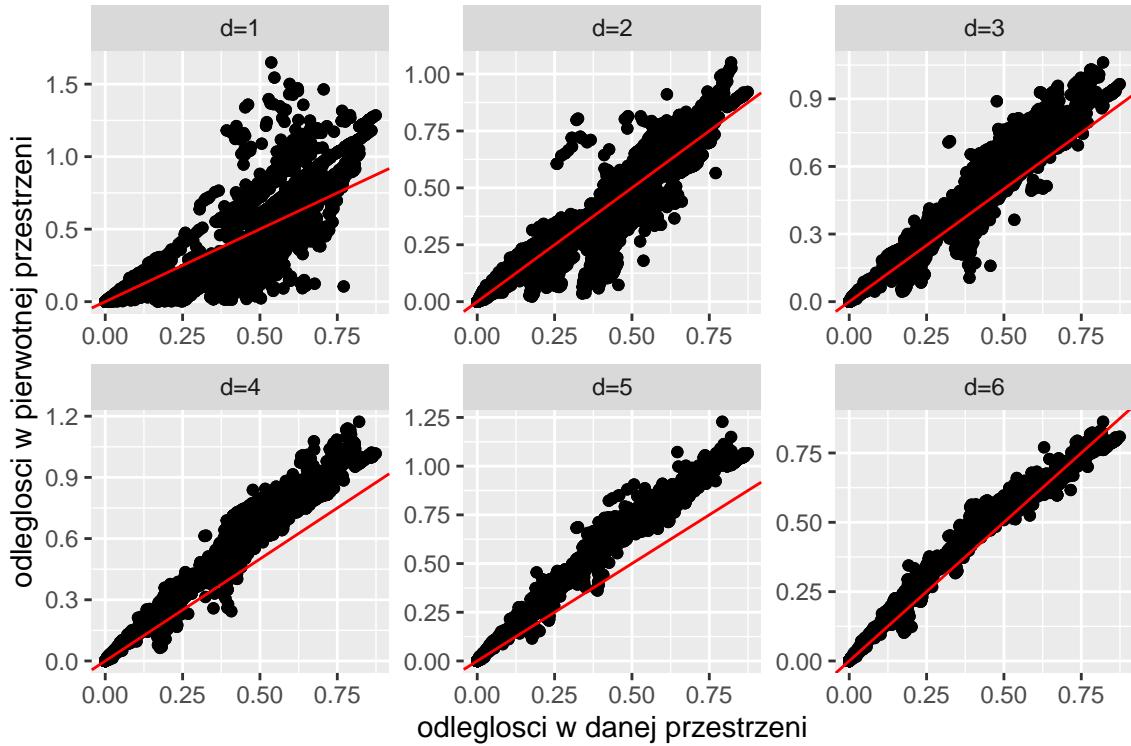
  data.for.shepard.2[paste0("d=", i)] = as.vector(new.dist.matrix)
}
```



Rysunek 11: Diagramy Sheparda dla kolejnych wymiarów



Rysunek 12: Wykres funkcji STRESS dla skalowania Kruskala



Rysunek 13: Diagramy Sheparda dla kolejnych wymiarów dla skalowania Kruskala

Możemy zauważyć, że funkcja STRESS przyjmuje mniejsze wartości dla tego rodzaju skalowania. Jednak zmiany tej funkcji dla skalowania Kruskala są bardziej gładkie — optymalnym wymiarem docelowym również byłoby $d = 4$.

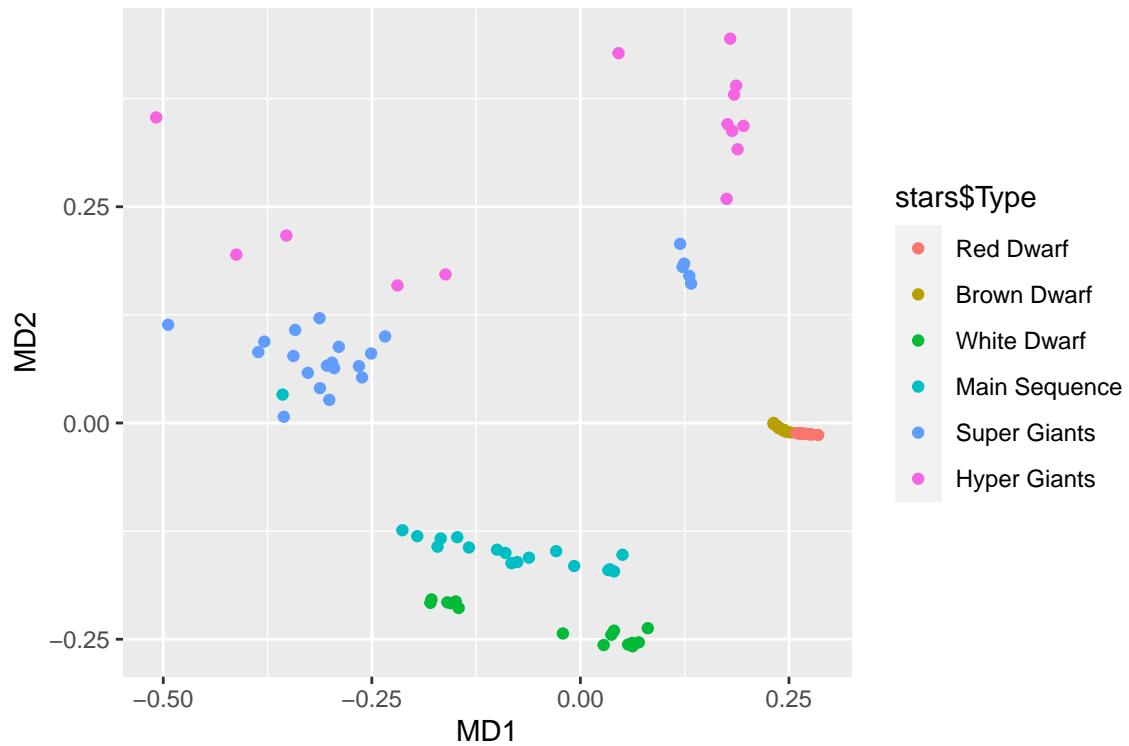
4.3 Wizualizacja danych

Wykorzystamy teraz metodę skalowania wielowymiarowego do przedstawienia naszych danych w przestrzeni dwuwymiarowej (przygotowaliśmy także wykresy w przestrzeni trójwymiarowej — znajdują się one w dołączonym do sprawozdania skrypcie).

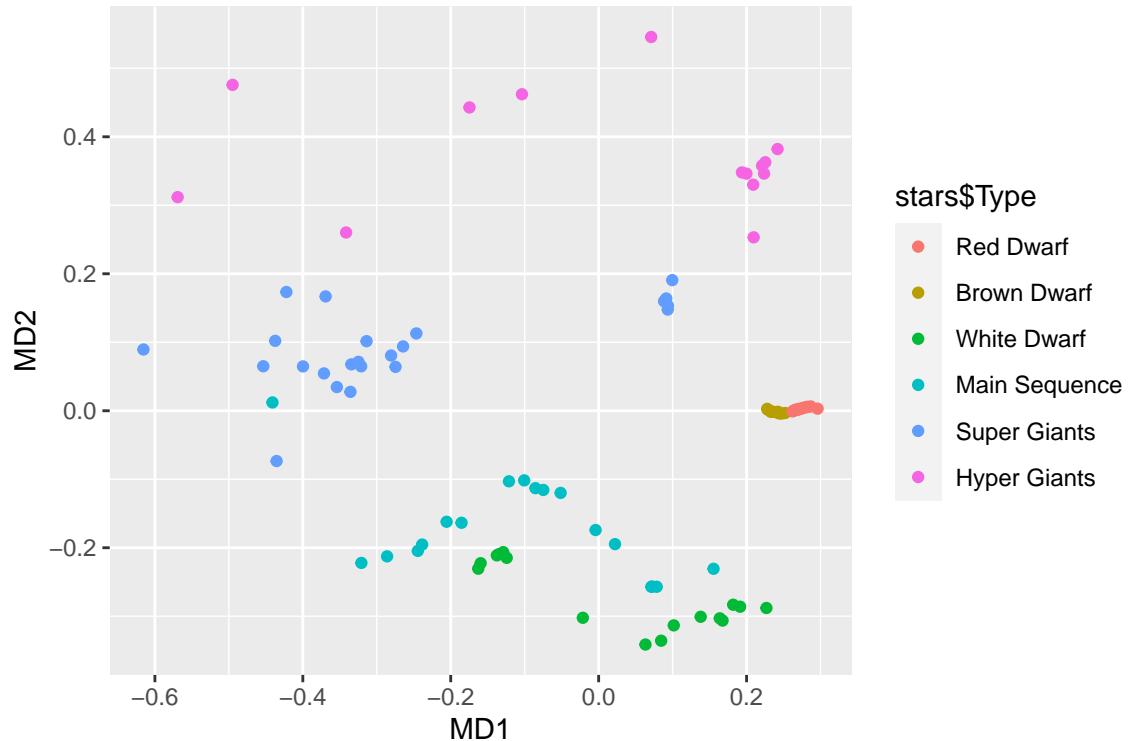
Dla obu metod widać wyraźny podział na klastry, które odpowiadają prawdziwym typom gwiazd. Podział jest tak wyraźny mimo naszych wcześniejszych wniosków dotyczących jakości skalowania klasycznego. Ciekawą obserwacją jest bardzo bliskie położenie czerwonych i brązowych karłów, które wręcz na siebie nachodzą. Odpowiada to rzeczywistości — brązowe i czerwone karły to obiekty o podobnych właściwościach (np. niska temperatura).

Na obu wykresach rozrzutu można zauważać obserwację odstającą — jest to jedna z gwiazd z ciągu głównego. Znajduje się ona na wykresach w klastrze odpowiadającym nadolbrzymom.

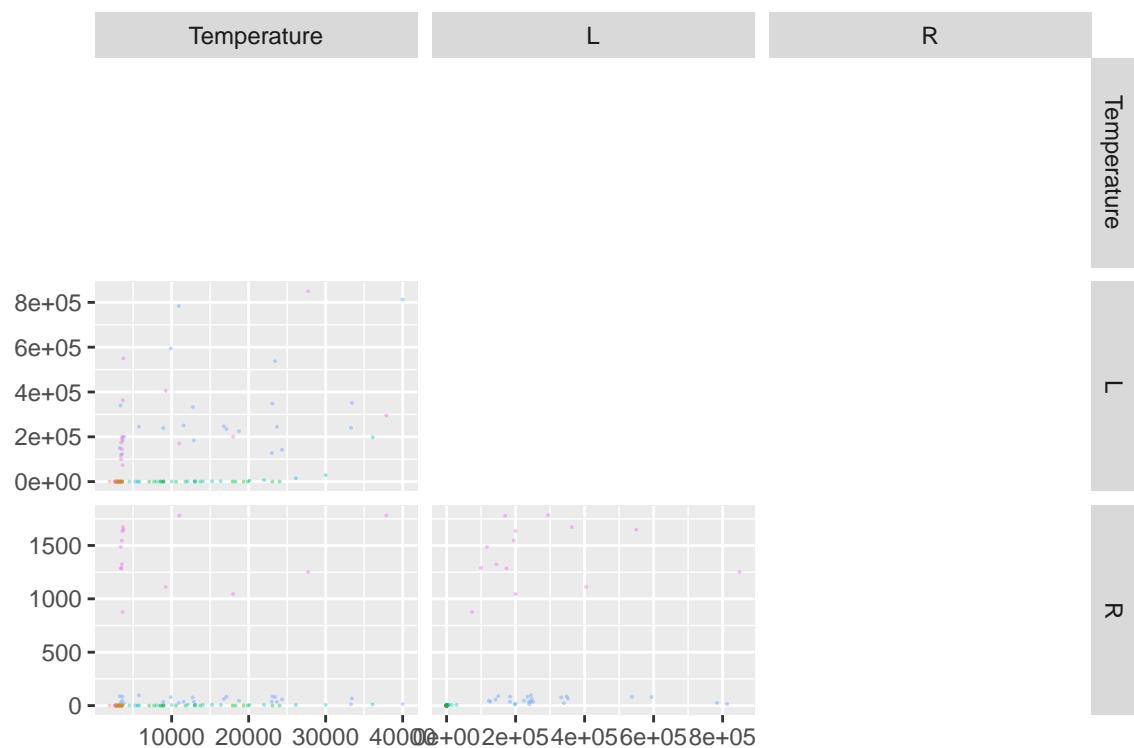
Zastosowanie MDS umożliwiło nam znacznie wartościową analizę wybranych przez nas danych. Na wykresach rozrzutu (16) nie widać tak dobrego podziału na klastry jak przy zastosowaniu metody skalowania wielowymiarowego.



Rysunek 14: Wykres rozrzutu w przestrzeni dwuwymiarowej — skalowanie klasyczne



Rysunek 15: Wykres rozrzutu w przestrzeni dwuwymiarowej — skalowanie Kruskala



Rysunek 16: Wykresy rozrzutu dla zmiennych ciągzych