

POLITECHNIKA WROCŁAWSKA  
WYDZIAŁ INFORMATYKI I ZARZĄDZANIA

---

KIERUNEK: INFORMATYKA  
SPECJALNOŚĆ: DANOLOGIA  
KURS: INDUKCYJNE METODY ANALIZY DANYCH

Klasyfikacja k-najbliższych sąsiadów  
Dokumentacja ćwiczenia nr 4

AUTOR:  
Adam Dłubak

PROWADZĄCY:  
Dr inż. Paweł Myszkowski

# Spis treści

<b>1. Zbiory Badawcze</b>	<b>3</b>
1.1. Iris Dataset	3
1.2. Wine Dataset	4
1.3. Glass Dataset	6
1.4. Pima Indians Diabetes Dataset	7
1.5. User Knowledge Modeling Dataset	8
<b>2. Wstęp teoretyczny</b>	<b>10</b>
2.1. Algorytm KNN	10
2.2. Sposoby obliczania odległości	11
2.3. Miary jakości klasyfikatora	12
<b>3. Badania i analiza wyników</b>	<b>13</b>
3.1. Kroswalidacja	13
3.1.1. Wine Dataset	13
3.1.2. Glass Dataset	14
3.1.3. Pima Indians Diabetes Dataset	14
3.1.4. User Knowledge Dataset	15
3.2. Parametr liczebności sąsiadów	15
3.2.1. Wine Dataset	15
3.2.2. Glass Dataset	16
3.2.3. Pima Indians Diabetes Dataset	16
3.2.4. User Knowledge Dataset	17
3.3. Parametry mierzenia odległości i sposobu głosowania	17
3.3.1. Wine Dataset	17
3.3.2. Glass Dataset	18
3.3.3. Pima Indians Diabetes Dataset	18
3.3.4. User Knowledge Dataset	19
3.4. Porównanie najlepszych uzyskanych wyników	19
3.4.1. Glass Dataset	19

# Rozdział 1

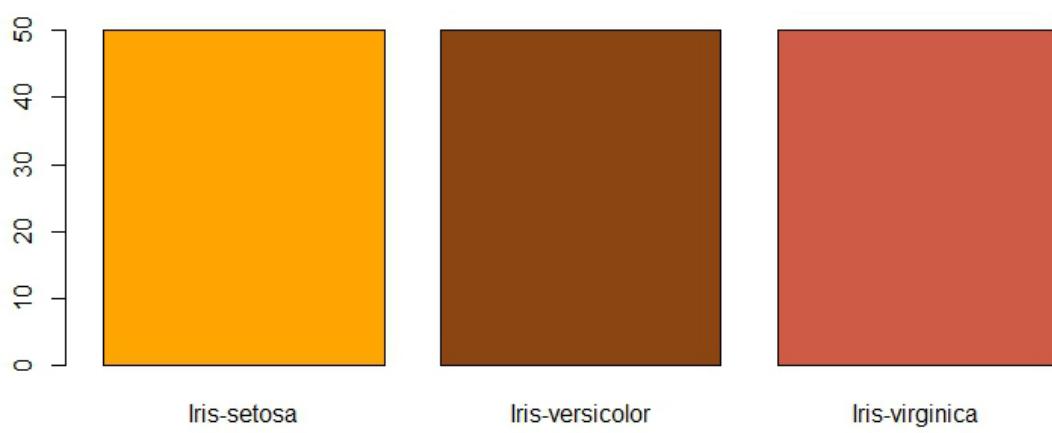
## Zbiory Badawcze

W ramach realizacji ćwiczenia, badania zostały oparte o 1 zbiór wykorzystany do wstępnej weryfikacji, 3 zbiory danych testowych wykorzystywanych już wcześniej, przeznaczonych głównie do zadania klasyfikacji oraz 1 zbiór, który jeszcze nie był badany i przeznaczony jest szczególnie do zadań klasteryzacji. Wykorzystane zbiory danych to:

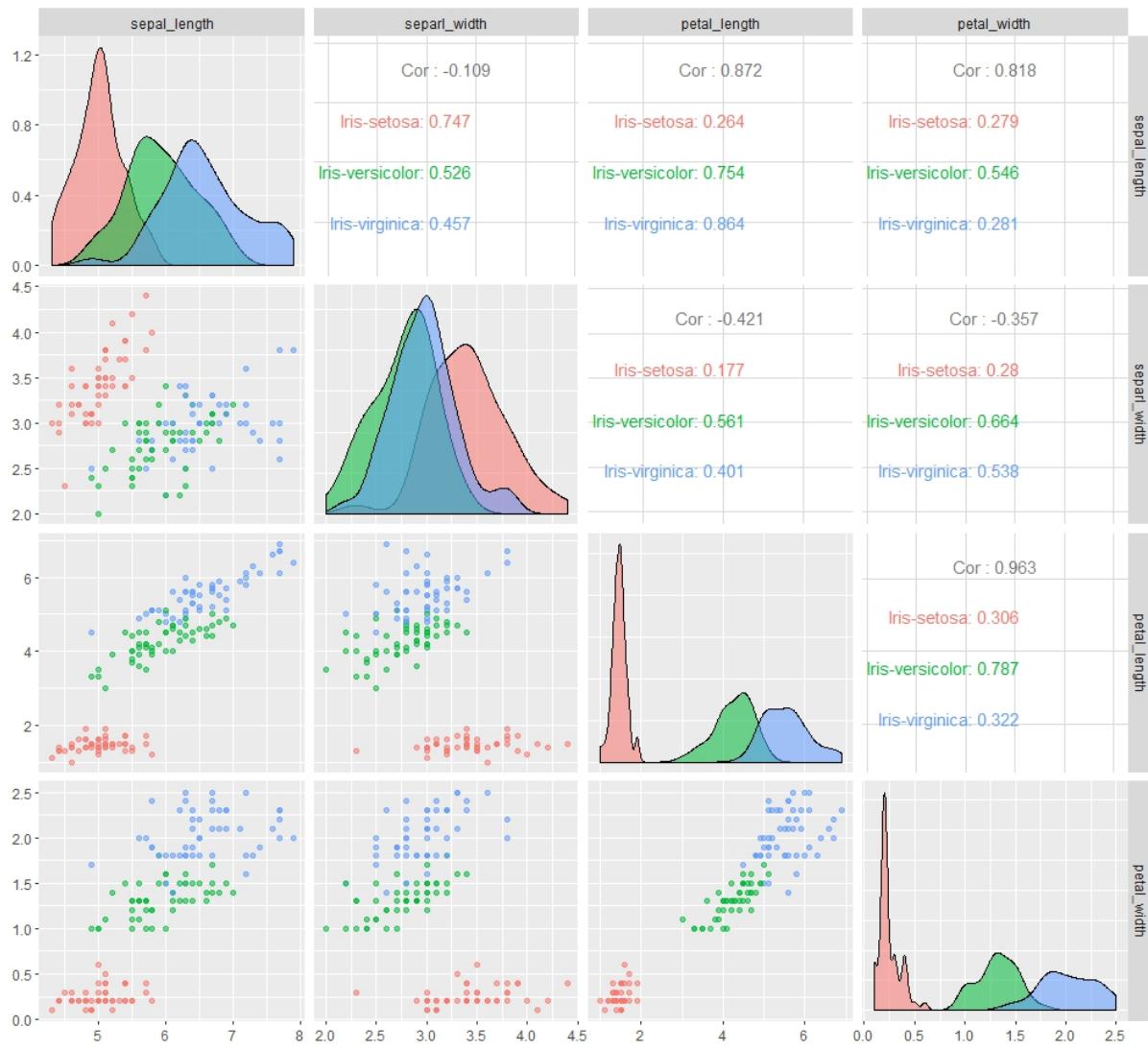
### 1.1. Iris Dataset

Zestaw pomiarów kwiatów irysa, udostępniony po raz pierwszy przez Ronaldiego Fishera w roku 1936. Jeden z najbardziej znanych zbiorów, a zarazem bardzo prosty i użyteczny. Zbiór irysów składa się z 4 wartości pomiarów jego płatków (szerokości i długość) oraz klasy do jakiej należy.

- Liczba atrybutów: 4
- Rodzaj atrybutów: wartości typu Float
- Liczba instancji: 150
- Liczba klas: 3



Rys. 1.1: Rozłożenie ilościowe poszczególnych klas zbioru Iris

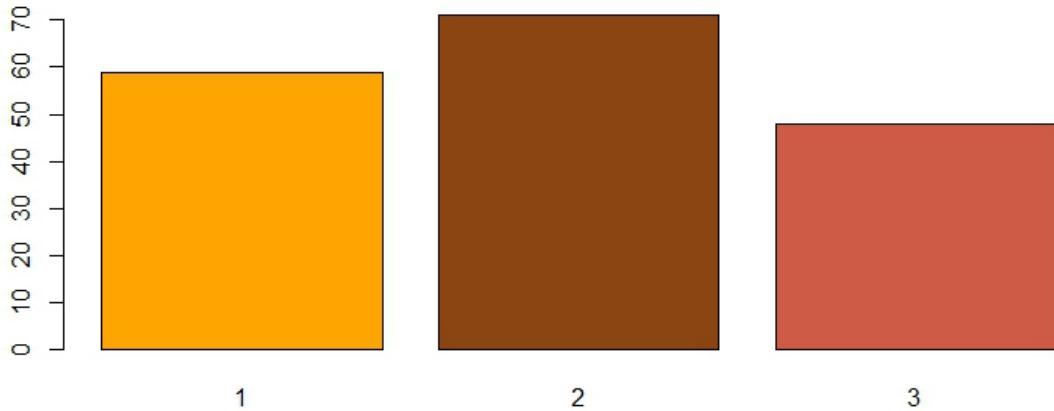


Rys. 1.2: Zestawienie zależności atrybutów i klas zbioru Iris

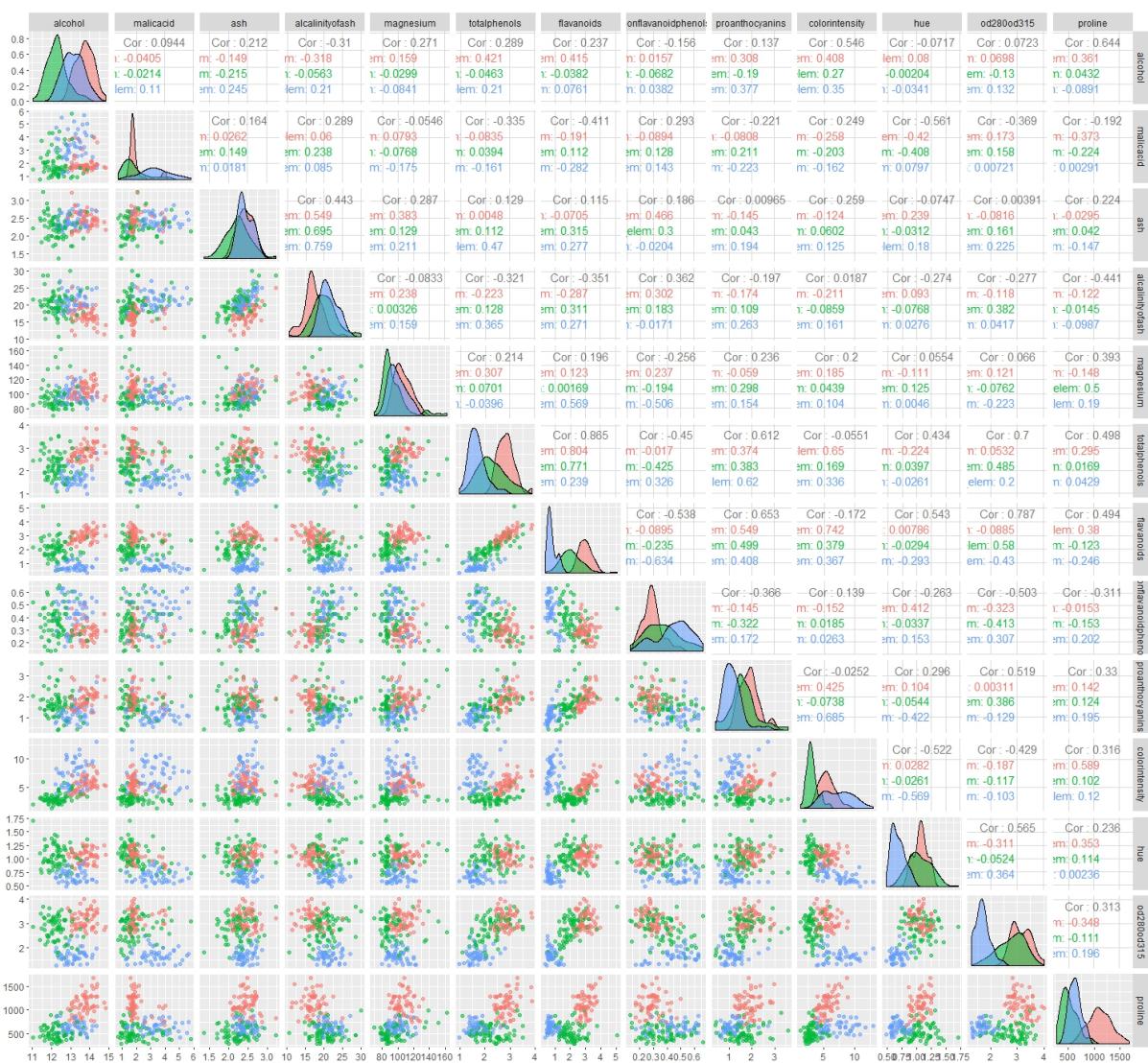
## 1.2. Wine Dataset

Zbiór danych jest wynikiem analizy chemicznej win uprawianych w tym samym regionie we Włoszech, ale uzyskanych z trzech różnych odmian. W analizie określono 13 składników znalezionych w każdym z trzech rodzajów win.

- Liczba atrybutów: 13.
- Rodzaj atrybutów: wartości typu Float i Intiger.
- Liczba instancji: 178.
- Liczba klas: 3



Rys. 1.3: Rozłożenie ilościowe poszczególnych klas zbioru Wine

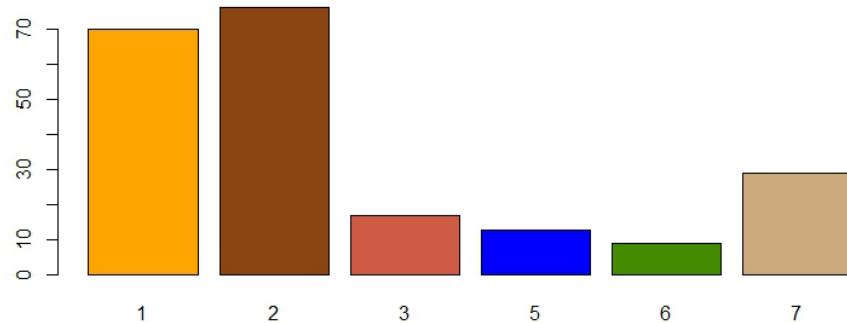


Rys. 1.4: Zestawienie zależności atrybutów i klas zbioru Wine

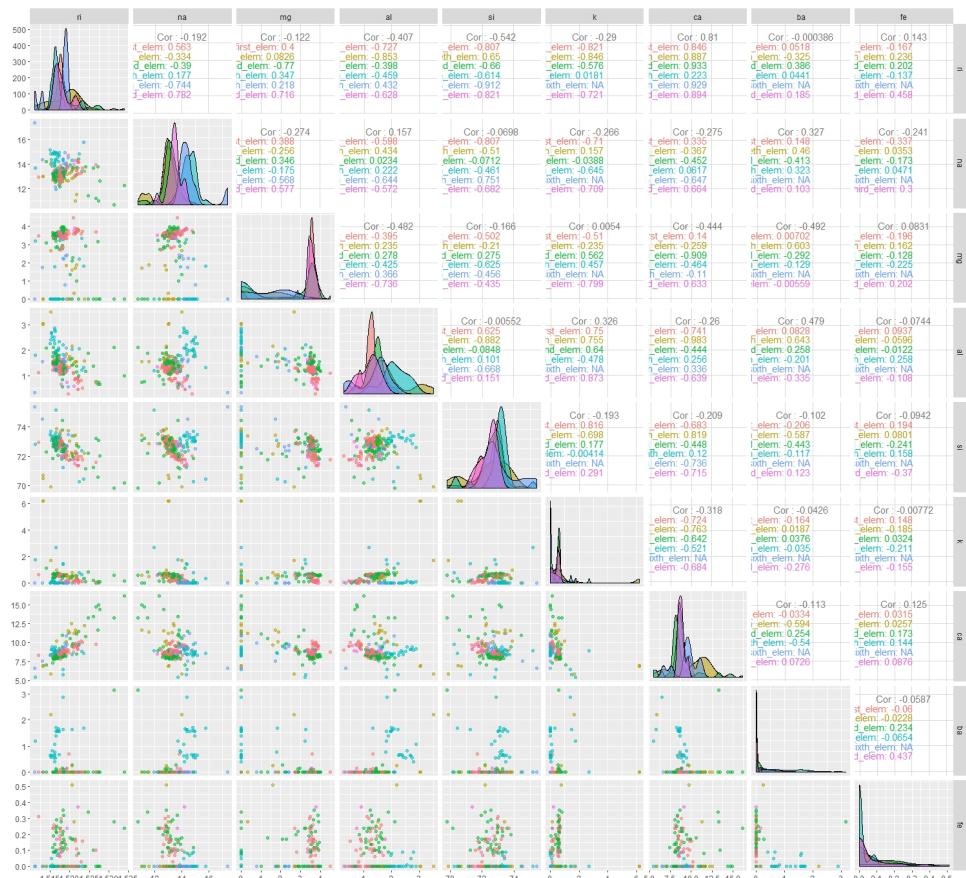
## 1.3. Glass Dataset

Zbiór danych powstał w wyniku motywacji badania dochodzeń kryminologicznych. Poprawne zidentyfikowanie rodzaju szkła znalezioneego na miejscu przestępstwa, na postawie jego składu pozwala na użycie go jako dowodu w sprawie.

- Liczba atrybutów: 9.
- Rodzaj atrybutów: realistyczne, ciągłe.
- Liczba instancji: 214.
- Liczba klas: 7.



Rys. 1.5: Rozłożenie ilościowe poszczególnych klas zbioru Glass

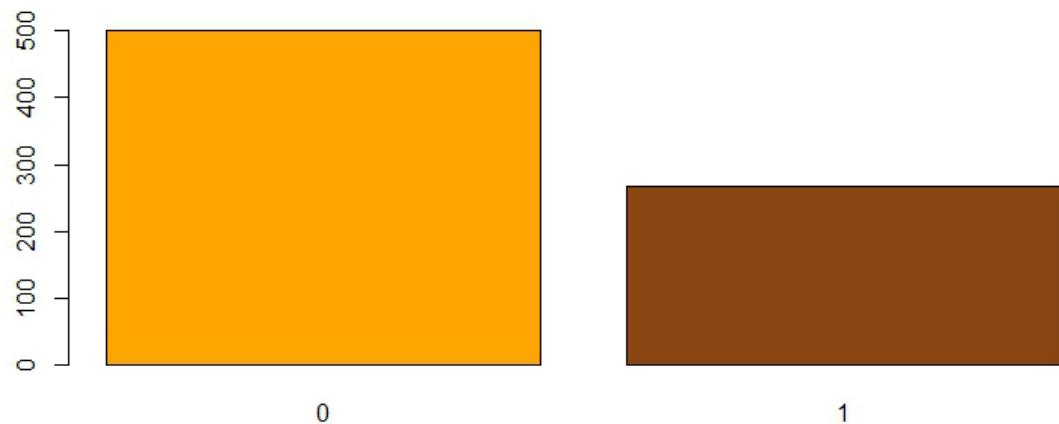


Rys. 1.6: Zestawienie zależności atrybutów i klas zbioru Glass

## 1.4. Pima Indians Diabetes Dataset

Zbiór danych Pima Indian Diabetes przewidywanie wystąpienie cukrzycy w oparciu o badania diagnostyczne. Pochodzi on z *National Institute of Diabetes and Digestive and Kidney Diseases*. Zawiera on dane dotyczące zachorowań na cukrzycę wśród kobiet z indiańskiego plemienia Pima. Każdy z 768 obiektów zbioru opisany jest przy pomocy 8 cech zawierających następujące informacje: ile razy pacjentka była w ciąży, test tolerancji glukozy, ciśnienie rozkurczowe, grubość zagięcia skóry, poziom insuliny, masę ciała, czy ktoś w rodzinie był chory na cukrzycę oraz wiek pacjentki. Każdy z obiektów przynależy do jednej z dwóch klas. Pierwsza klasa oznacza, że pacjentka nie choruje na cukrzycę, a druga klasa oznacza, że dana kobieta jest diabetykiem.

- Liczba atrybutów: 8.
- Rodzaj atrybutów: realistyczne, ciągłe i typu Intiger (wiek i ilość dotychczasowych ciąży).
- Liczba instancji: 768.
- Liczba klas: 2 - wartość 1 (pozytywna) lub 0 (negatywna).



Rys. 1.7: Rozłożenie ilościowe poszczególnych klas zbioru Diabetes

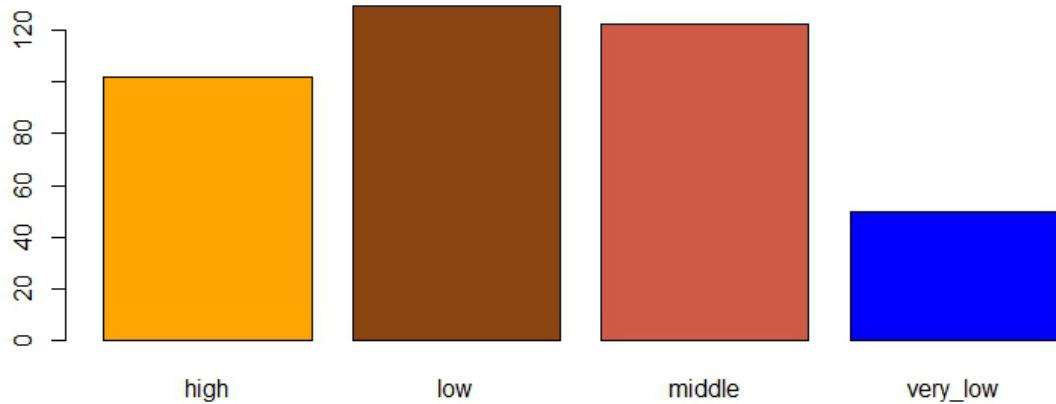


Rys. 1.8: Zestawienie zależności atrybutów i klas zbioru Diabetes

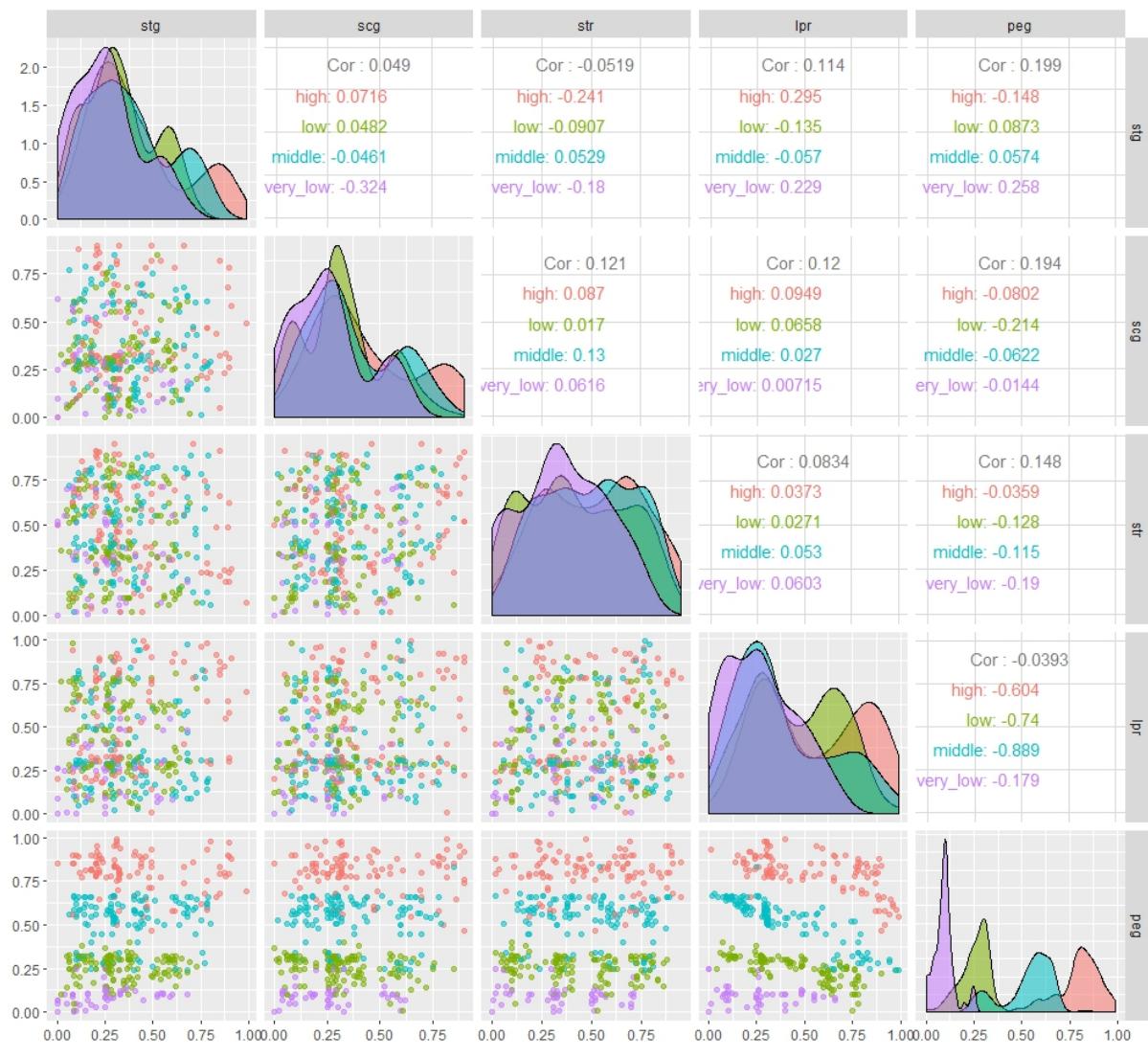
## 1.5. User Knowledge Modeling Dataset

Zbiór danych User Knowledge został sklasyfikowany przez autorów za pomocą intuicyjnego klasyfikatora wiedzy (technika hybrydowego uczenia maszynowego metod k-NN i meta-heurystycznych) oraz algorytmu k najbliższych sąsiadów. Zawiera 5 atrybutów opisujących m.in. stopień czasu nauki oraz liczbę powtórzeń koniecznych do osiągnięcia danego celu, czy chociażby osiągnięte rezultaty na egzaminie.

- Liczba atrybutów: 5.
- Rodzaj atrybutów: realistyczne, ciągłe.
- Liczba instancji: 403.
- Liczba klas: 4.



Rys. 1.9: Rozłożenie ilościowe poszczególnych klas zbioru User Knowledge



Rys. 1.10: Zestawienie zależności atrybutów i klas zbioru User Knowledge

# Rozdział 2

## Wstęp teoretyczny

### 2.1. Algorytm KNN

Algorytm k najbliższych sąsiadów (*ang. k nearest neighbours*) to jeden z algorytmów regresji nieparametrycznej używanych w statystyce do prognozowania wartości pewnej zmiennej losowej. Może również być używany do klasyfikacji.

**Założenia:**

1. Dany jest zbiór uczący zawierający obserwacje z których każda ma przypisany wektor zmiennych objaśniających  $X_1 \dots X_n$  oraz wartość zmiennej objaśnianej  $Y$ .
2. Dana jest obserwacja  $C$  z przypisany wektorem zmiennych objaśniających  $X_1 \dots X_n$  dla której chcemy prognozować wartość zmiennej objaśnianej  $Y$ .

**Algorytm polega na:**

1. Porównaniu wartości zmiennych objaśniających dla obserwacji  $C$  z wartościami tych zmiennych dla każdej obserwacji w zbiorze uczącym.
2. Wyborze  $k$  (ustalona z góry liczba) najbliższych do  $C$  obserwacji ze zbioru uczącego.
3. Uśrednieniu wartości zmiennej objaśnianej dla wybranych obserwacji, w wyniku czego uzyskujemy prognozę.

Definicja „najbliższych obserwacji” w punkcie 2 sprowadza się do minimalizacji pewnej metryki, mierzącej odległość pomiędzy wektorami zmiennych objaśniających dwóch obserwacji. Zwykle stosowana jest tu metryka euklidesowa lub metryka Mahalanobisa. Można również zamiast średniej arytmetycznej stosować np. medianę.

Algorytm k najbliższych sąsiadów jest użyteczny szczególnie wtedy, gdy zależność między zmiennymi objaśniającymi a objaśnianymi jest złożona lub nietypowa (np. niemonotoniczna), czyli trudna do modelowania w klasyczny sposób. W przypadku, gdy zależność ta jest łatwa do interpretacji (np. liniowa), a zbiór nie zawiera obserwacji odstających, metody klasyczne (np. regresja liniowa) dadzą zwykle dokładniejsze wyniki.

**Cechy algorytmu:**

- Bardziej odporny na szумy - w poprzednim algorytmie obiekt najbliższy klasyfikowanemu może być zniekształcony - tak samo zostanie zaklasyfikowany nowy obiekt.
- Konieczność ustalenia liczby najbliższych sąsiadów.
- Wyznaczenie miary podobieństwa wśród obiektów (wiele miar podobieństwa).
- Dobór parametru  $k$  - liczby sąsiadów: jeśli  $k$  jest małe, algorytm nie jest odporny na szumy – jakość klasyfikacji jest niska. Jeśli  $k$  jest duże, czas działania algorytmu rośnie - większa złożoność obliczeniowa. Należy wybrać  $k$ , które daje najwyższą wartość klasyfikacji.

**Parametry funkcji *KNeighborsClassifier()* w Python:**

- **nneighbors** - liczba sąsiadów, których należy brać pod uwagę w trakcie klasyfikacji,
- **weights** - funkcja wagowa (głosowania) używana w prognozowaniu. Możliwe wartości:
  - **uniform** - wszystkie punkty w każdej okolicy są równomiernie ważone,
  - **distance** - punkty ciężkości przez odwrotność ich odległości. w tym przypadku bliżsi sąsiedzi punktu zapytania będą mieli większy wpływ niż sąsiedzi, którzy znajdują się dalej,
  - **callable** - funkcja zdefiniowana przez użytkownika, która akceptuje tablicę odległości i zwraca tablicę o tym samym kształcie zawierającym wagi. Realizacja w tym projekcie własnej funkcji głosowania polegała na losowym ustalaniu wag głosowania dla tablicy odległości.
- **metric** - string lub callable, domyślnie ‘*minkowski*’. Metryka odległości używana dla budowy drzewa. Wykorzystane metryki omówione zostaną w dalszej części.

## 2.2. Sposoby obliczania odległości

**• Euclidean**

Odległość Euklidesowa obliczana jest według następującej formuły:

$$dist_{x,y} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.1)$$

Jest to najbardziej powszechna, znana i najczęściej wykorzystywana metoda obliczania odległości.

**• Manhattan**

Metryka miejska zwana także metryką Manhattan obliczana według następującej formuły:

$$dist_{x,y} = \sum_{i=1}^n |x_i - y_i| \quad (2.2)$$

Jest to suma wartości bezwzględnych różnic współrzędnych punktów  $x$  oraz  $y$ .

- **Chebyshev**

Metryka obliczana według następującej formuły:

$$dist_{x,y} = \max_i |x_i - y_i| \quad (2.3)$$

Miara ta została wprowadzona przez Pafnutija Czebyszewa i jest specjalnym przypadkiem odległości Minkowskiego. W szachach jest to odległość między polami szachownicy wyrażona w ruchach, które musi wykonać figura króla. Stąd pochodzi jej angielska nazwa *chessboard distance*.

- **Minkowski**

Metryka obliczana według następującej formuły:

$$dist_{x,y} = \left( \sum_{i=1}^n |x_i - y_i|^p \right)^{1/p} \quad (2.4)$$

Uogólniona miara odległości między punktami przestrzeni euklidesowej. Można o niej myśleć jako o uogólnieniu odległości euklidesowej, miejskiej oraz Czebyszewa.

## 2.3. Miary jakości klasyfikatora

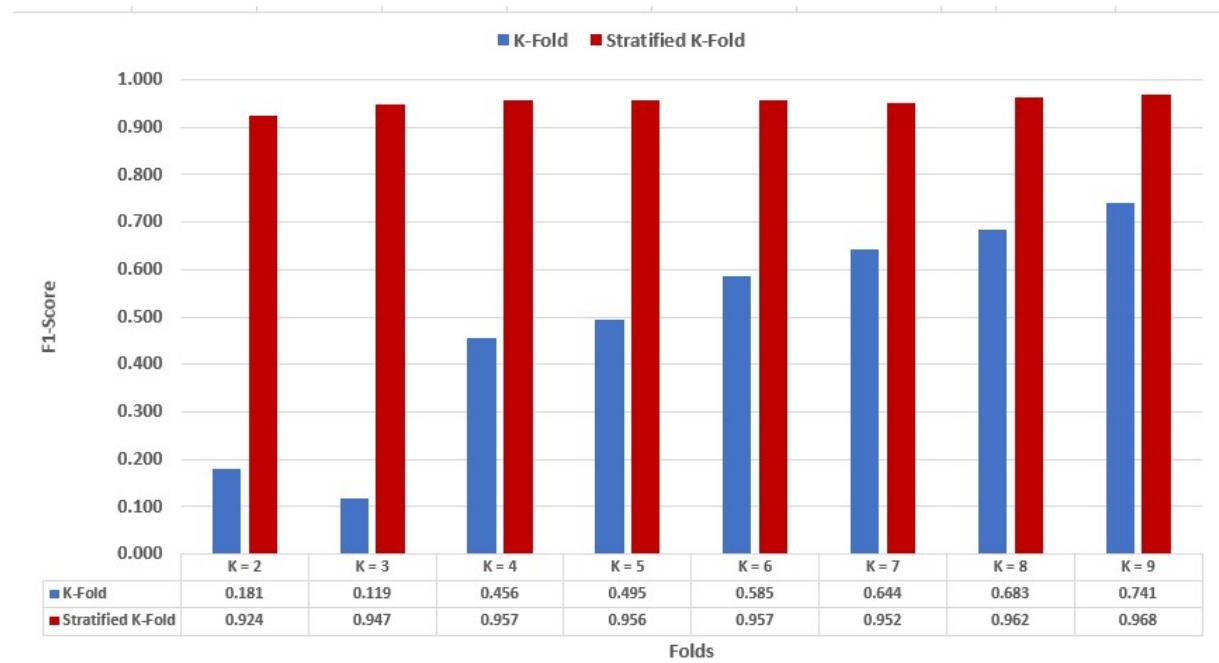
- **Trafność (ang. Accuracy)** - określa, jaka część prognozowanych etykiet jest zgodna z rzeczywistymi wynikami. Oznacza to procent poprawnie sklasyfikowanych etykiet.
- **Precyzja (ang. Precision)** - określa liczbę adekwatnych elementów w zbiorze wyników. W kontekście klasyfikacji jest to liczba poprawnych etykiet z wszystkich zbiorów sklasyfikowanych etykiet. Wyniki są uśredniane dla wszystkich etykiet.
- **Czułość (ang. Recall)** - określa liczbę poprawnych wyników względem liczby wszystkich poprawnych etykiet. W kontekście klasyfikacji jest to liczba poprawnie sklasyfikowanych etykiet w zbiorze podzielona przez łączną liczbę etykiet ze zbioru. Wyniki są uśredniane.
- **Wskaźnik F1 (ang. F1 Score)** - jest to średnia harmoniczna precyzji i czułości. Najczęściej stosowana jest dla niezrównoważonych zbiorów danych w celu ustalenia, czy klasyfikator działa dobrze dla wszystkich klas.

# Rozdział 3

## Badania i analiza wyników

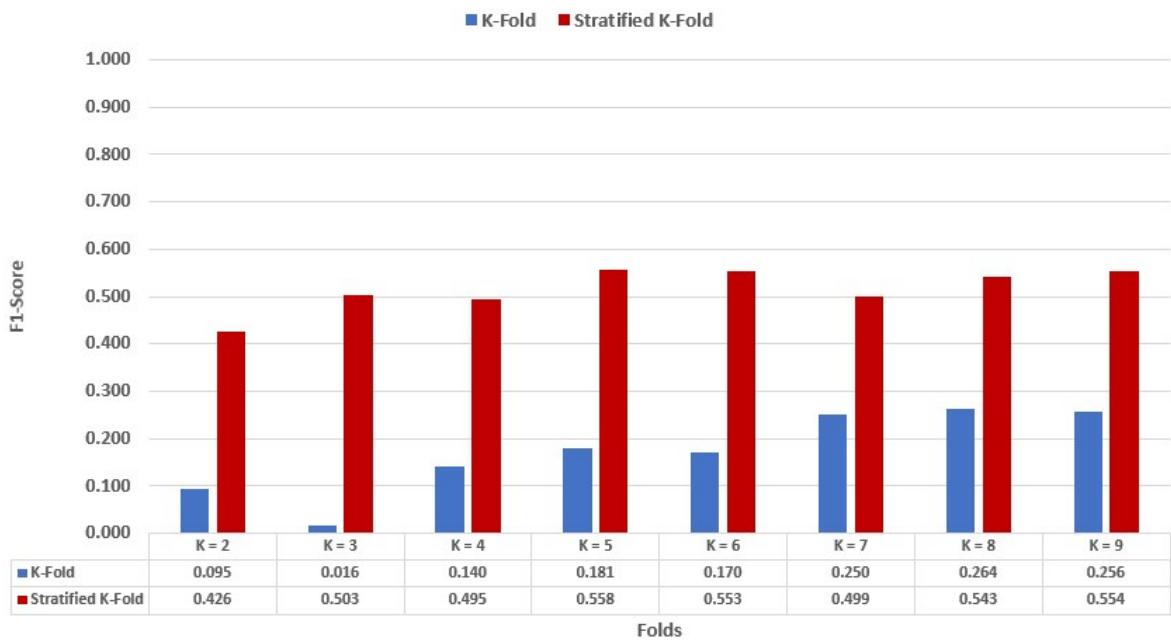
### 3.1. Kroswalidacja

#### 3.1.1. Wine Dataset



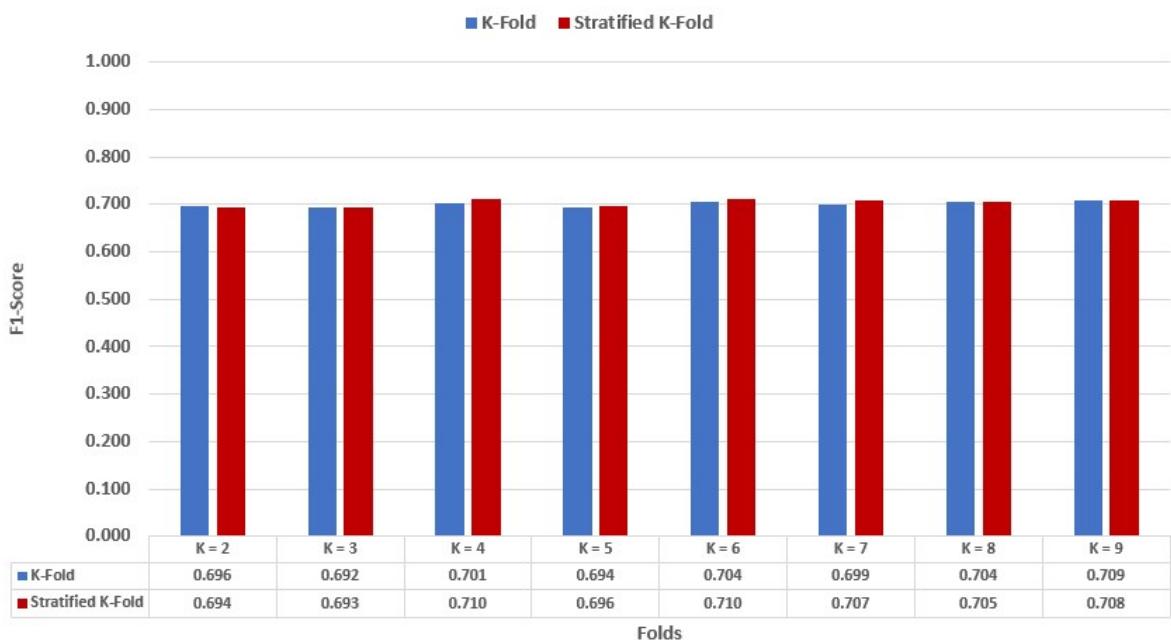
Rys. 3.1: Wartość parametru K dla kroswalidacji zbioru Wine

### 3.1.2. Glass Dataset



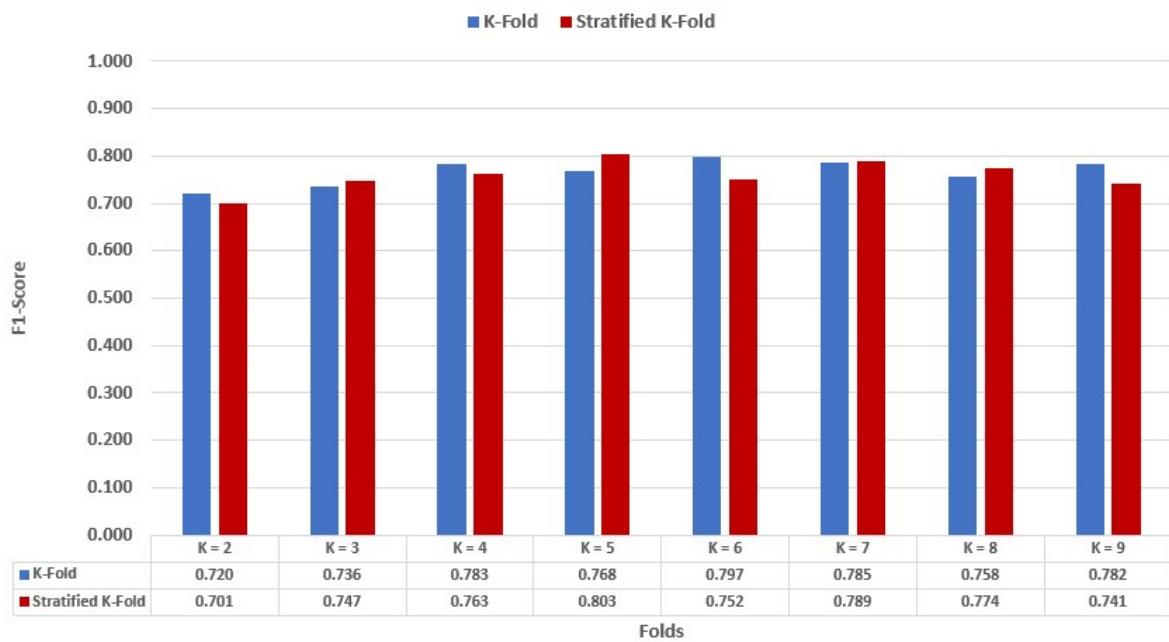
Rys. 3.2: Wartość parametru K dla kroswalidacji zbioru Glass

### 3.1.3. Pima Indians Diabetes Dataset



Rys. 3.3: Wartość parametru K dla kroswalidacji zbioru Pima Indians

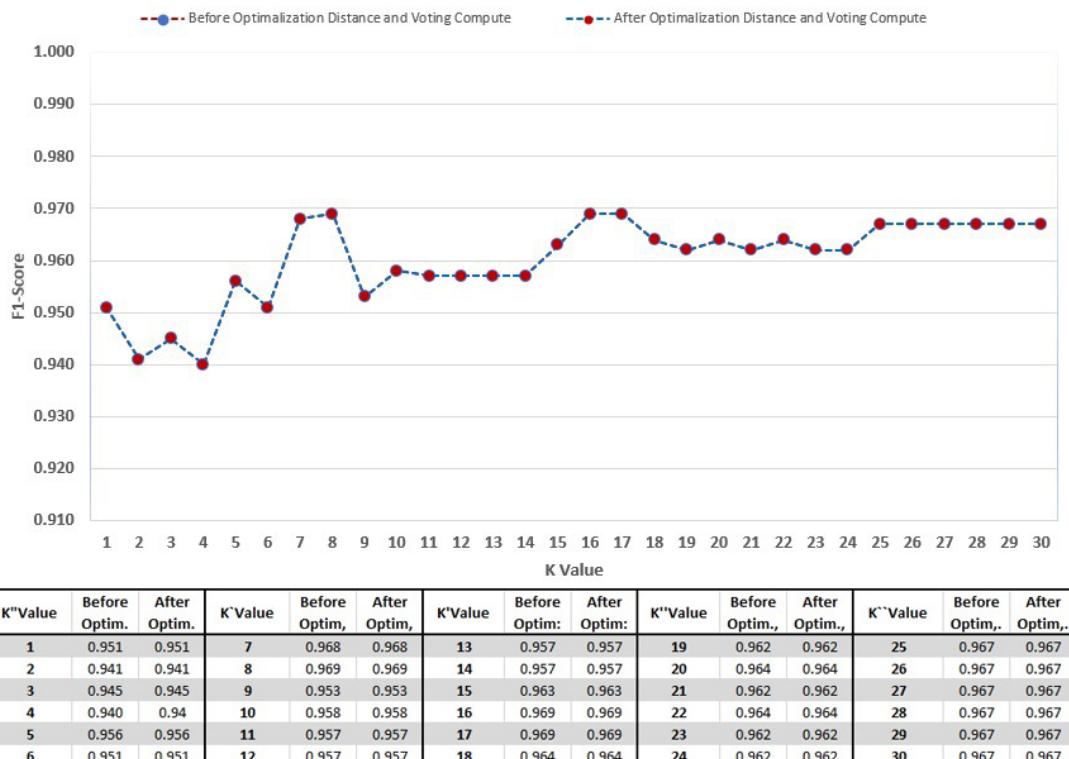
### 3.1.4. User Knowledge Dataset



Rys. 3.4: Wartość parametru K dla kroswalidacji zbioru Knowledge

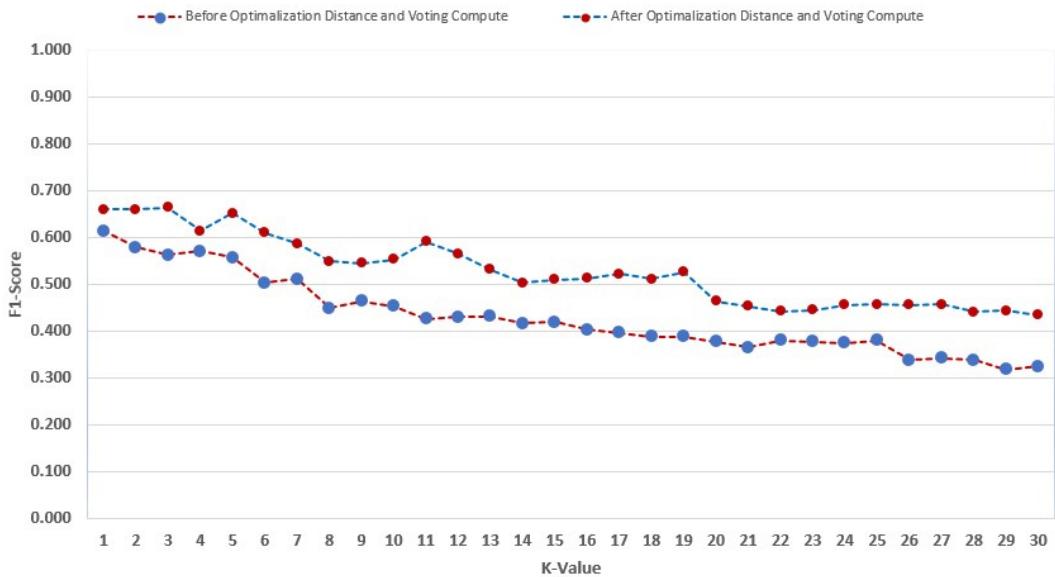
## 3.2. Parametr liczebności sąsiadów

### 3.2.1. Wine Dataset



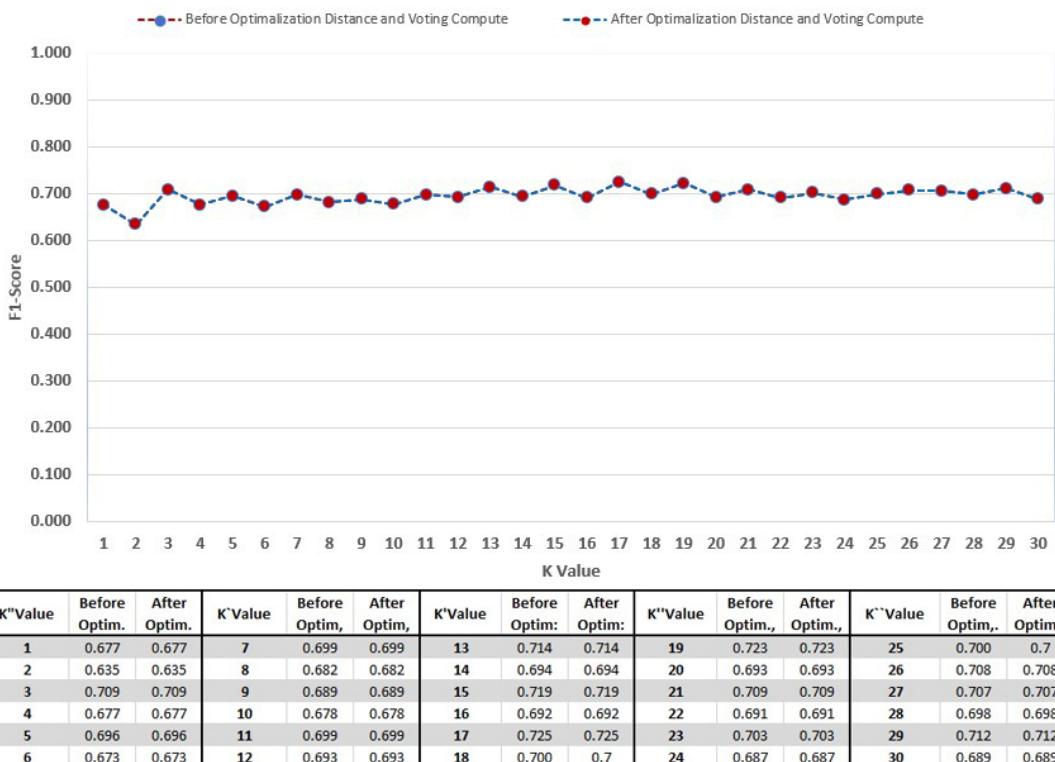
Rys. 3.5: Wartość parametru K dla algorytmu KNN zbioru Wine

### 3.2.2. Glass Dataset



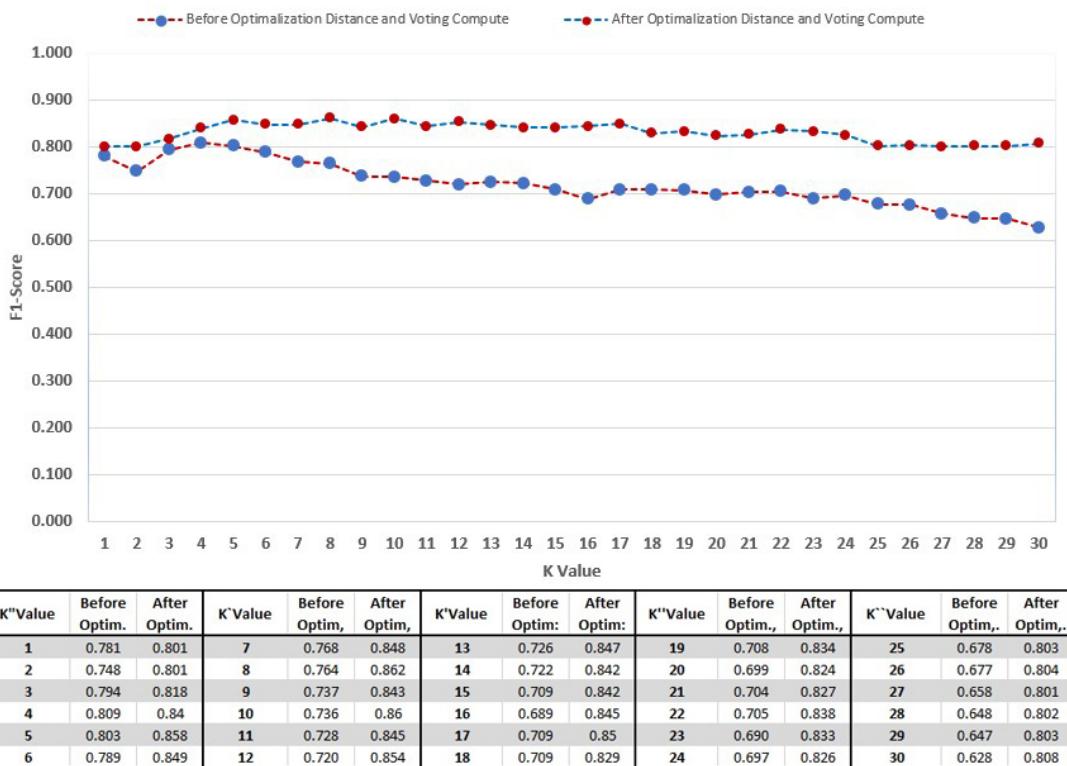
Rys. 3.6: Wartość parametru K dla algorytmu KNN zbioru Glass

### 3.2.3. Pima Indians Diabetes Dataset



Rys. 3.7: Wartość parametru K dla algorytmu KNN Pima Indians

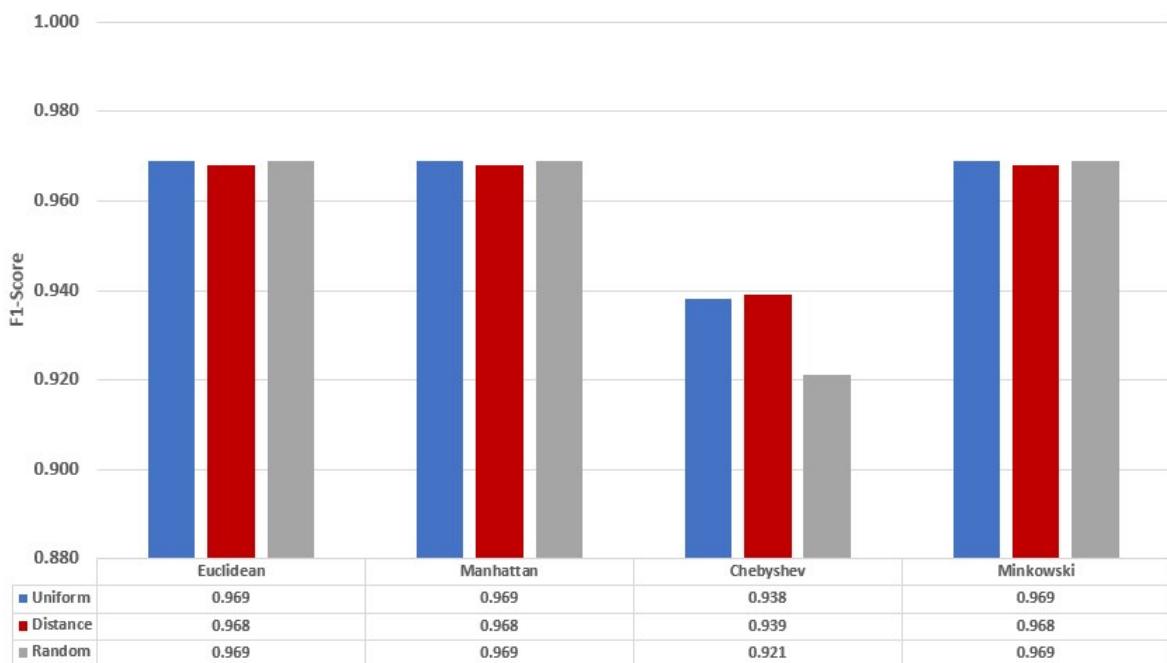
### 3.2.4. User Knowledge Dataset



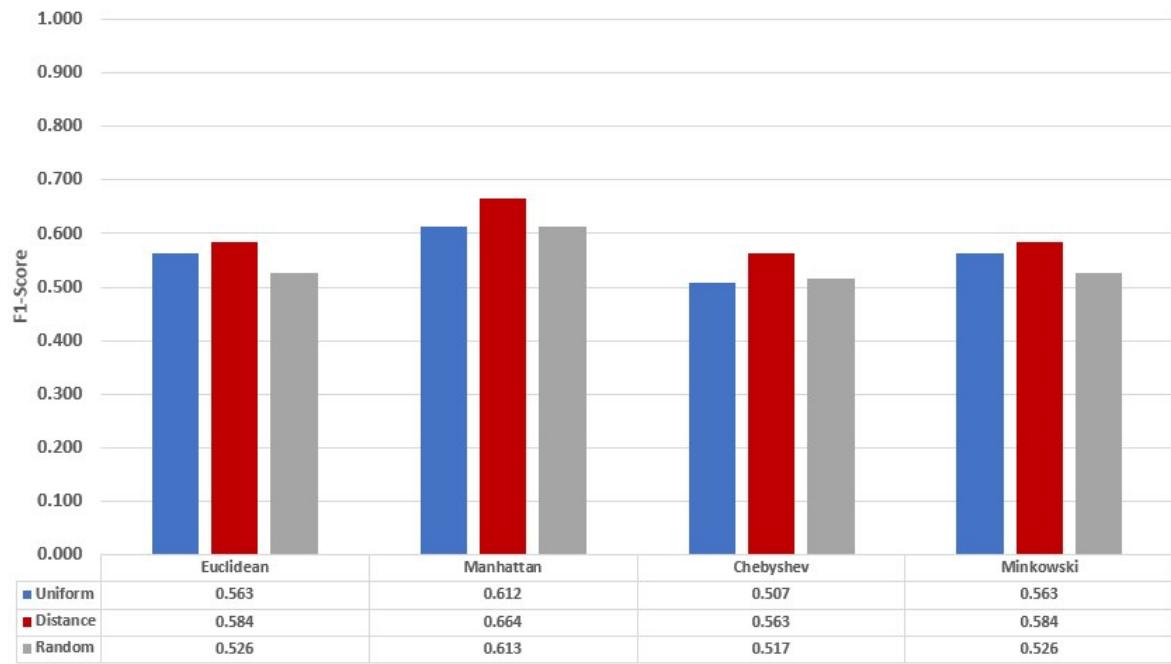
Rys. 3.8: Wartość parametru K dla algorytmu KNN Knowledge

## 3.3. Parametry mierzenia odległości i sposobu głosowania

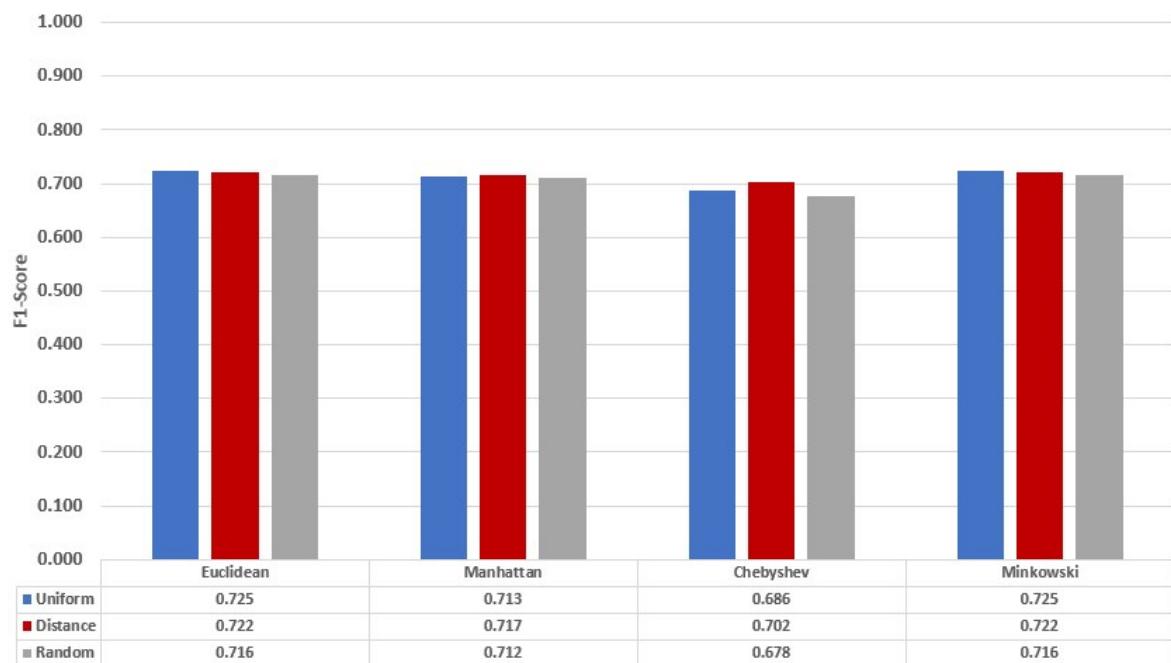
### 3.3.1. Wine Dataset

Rys. 3.9: Wartości parametrów *weight* i *metric* dla zbioru Wine

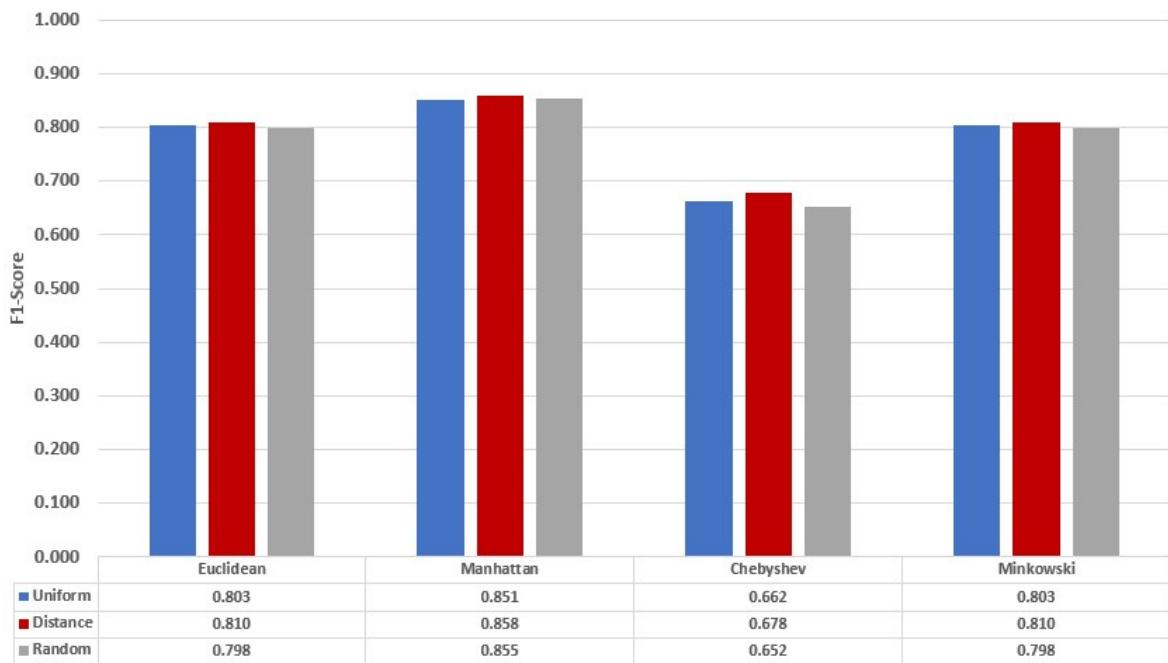
### 3.3.2. Glass Dataset

Rys. 3.10: Wartości parametrów *weight* i *metric* dla zbioru Glass

### 3.3.3. Pima Indians Diabetes Dataset

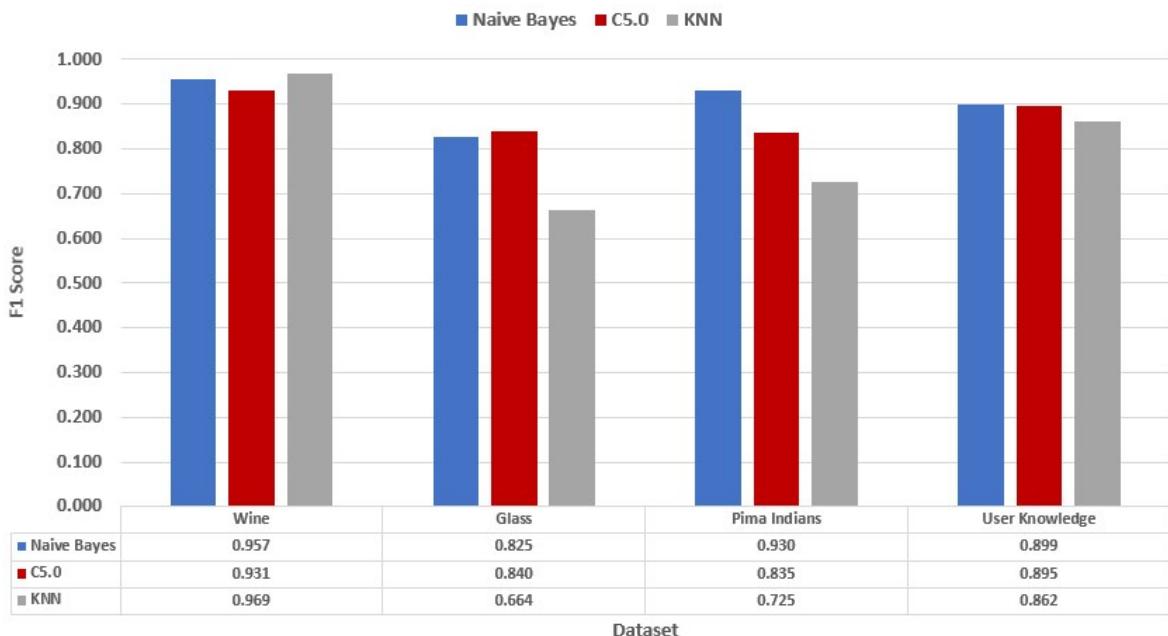
Rys. 3.11: Wartości parametrów *weight* i *metric* dla zbioru Pima Indians

### 3.3.4. User Knowledge Dataset

Rys. 3.12: Wartości parametrów *weight* i *metric* dla zbioru Knowledge

## 3.4. Porównanie najlepszych uzyskanych wyników

### 3.4.1. Glass Dataset



Rys. 3.13: Porównanie najlepszych wyników metod: Bayes, C5.0 i KNN

# Spis rysunków

1.1.	Rozłożenie ilościowe poszczególnych klas zbioru Iris . . . . .	3
1.2.	Zestawienie zależności atrybutów i klas zbioru Iris . . . . .	4
1.3.	Rozłożenie ilościowe poszczególnych klas zbioru Wine . . . . .	5
1.4.	Zestawienie zależności atrybutów i klas zbioru Wine . . . . .	5
1.5.	Rozłożenie ilościowe poszczególnych klas zbioru Glass . . . . .	6
1.6.	Zestawienie zależności atrybutów i klas zbioru Glass . . . . .	6
1.7.	Rozłożenie ilościowe poszczególnych klas zbioru Diabetes . . . . .	7
1.8.	Zestawienie zależności atrybutów i klas zbioru Diabetes . . . . .	8
1.9.	Rozłożenie ilościowe poszczególnych klas zbioru User Knowledge . . . . .	9
1.10.	Zestawienie zależności atrybutów i klas zbioru User Knowledge . . . . .	9
3.1.	Wartość parametru K dla kroswalidacji zbioru Wine . . . . .	13
3.2.	Wartość parametru K dla kroswalidacji zbioru Glass . . . . .	14
3.3.	Wartość parametru K dla kroswalidacji zbioru Pima Indians . . . . .	14
3.4.	Wartość parametru K dla kroswalidacji zbioru Knowledge . . . . .	15
3.5.	Wartość parametru K dla algorytmu KNN zbioru Wine . . . . .	15
3.6.	Wartość parametru K dla algorytmu KNN zbioru Glass . . . . .	16
3.7.	Wartość parametru K dla algorytmu KNN Pima Indians . . . . .	16
3.8.	Wartość parametru K dla algorytmu KNN Knowledge . . . . .	17
3.9.	Wartości parametrów <i>weight</i> i <i>metric</i> dla zbioru Wine . . . . .	17
3.10.	Wartości parametrów <i>weight</i> i <i>metric</i> dla zbioru Glass . . . . .	18
3.11.	Wartości parametrów <i>weight</i> i <i>metric</i> dla zbioru Pima Indians . . . . .	18
3.12.	Wartości parametrów <i>weight</i> i <i>metric</i> dla zbioru Knowledge . . . . .	19
3.13.	Porównanie najlepszych wyników metod: Bayes, C5.0 i KNN . . . . .	19