

ALGORYTMY UCZENIA MASZYNOWEGO

---

## Heart Disease Prediction machine learning

---

Autor:  
inż. Wiktor Springer 248970

Prowadzący: mgr inż. Marcin Ochman

19 czerwca 2022



Politechnika  
Wrocławska

# Spis treści

<b>1</b>	<b>Opis problemu</b>	<b>1</b>
<b>2</b>	<b>Wizualizacja zbioru danych</b>	<b>2</b>
<b>3</b>	<b>Opis programu</b>	<b>8</b>
3.1	Proces uczenia modelu . . . . .	8
3.2	Działanie programu . . . . .	15
3.3	Przykładowe wywołania programu . . . . .	15

# 1 Opis problemu

Tematem projektu było stworzenie programu wykorzystującego algorytmy uczenia maszynowego w procesie predykcji chorób serca. W ramach projektu użyto zbioru danych, którego cechy zostały zaprezentowane poniżej:

1. age - wiek w latach
2. sex - płeć (1 = mężczyzna, 0 = kobieta)
3. cp - ból klatki piersiowej
  - 0: Typowy ból dławicowy: ból w klatce piersiowej spowodowany zmniejszeniem dopływu krwi do serca.
  - 1: Nietypowy ból dławicowy: ból w klatce piersiowej niepowiązany z sercem.
  - 2: Niedławicowy: typowe skurcze przełyku.
  - 3: Bezobjawowy: ból klatki piersiowej niewskazujący sygnał choroby.
4. trestbps - spoczynkowe ciśnienie krwi (w mm Hg przy przyjęciu do szpitala) wszystko powyżej 130-140 jest zwykle powodem do niepokoju
5. chol - cholesterol w surowicy w mg/dl
  - surowica = LDL + HDL + 0,2 \* triglicerydy
  - powyżej 200 to powód do niepokoju
6. fbs - (cukier we krwi na czczo  $\geq$  120 mg/dl) (1 = prawda; 0 = fałsz)
  - $> 126 \frac{mg}{dL}$  powód do niepokoju
7. restecg - wyniki elektrokardiograficzne w stanie spoczynku
  - 0: Nic szczególnego
  - 1: Nieprawidłowa morfologia ST-T
    - może wahać się od łagodnych objawów do poważnych problemów
    - sygnalizuje nieprawidłowe bicie serca
  - 2: Możliwy lub zdecydowany przerost lewej komory
    - Powiększona główna komora pompowania serca
8. thalach - maksymalne osiągnięte tętno
9. exang - dławica wysiłkowa (1 = tak; 0 = nie)
10. oldpeak - depresja ST wywołana wysiłkiem fizycznym w stosunku do odpoczynku

11. slope - nachylenie szczytowego odcinka ST
- 0: Upsloping: lepsze tętno podczas ćwiczeń
  - 1: Flatsloping: minimalna zmiana (typowe zdrowe serce)
  - 2: Downsloping: oznaki niezdrowego serca
12. ca - liczba głównych naczyń krwionośnych (0-3) pokolorowanych fluorozopią
- kolorowe naczynie oznacza, że lekarz widzi przepływającą krew
  - im większy przepływ krwi, tym lepiej (brak skrzepów)
13. thal - wynik stresu talowego
- 1,3: stan normalny
  - 6: naprawiona wada: kiedyś była wada, ale teraz ok
  - 7: wada odwracalna: brak prawidłowego przepływu krwi podczas ćwiczeń
14. target - osoba chora lub nie (1=tak, 0=nie) (= przewidywany atrybut)

## 2 Wizualizacja zbioru danych

W tym rozdziale przedstawiono podstawowe własności zbioru danych wykorzystanego w procesie testowania oraz nauki modelu.

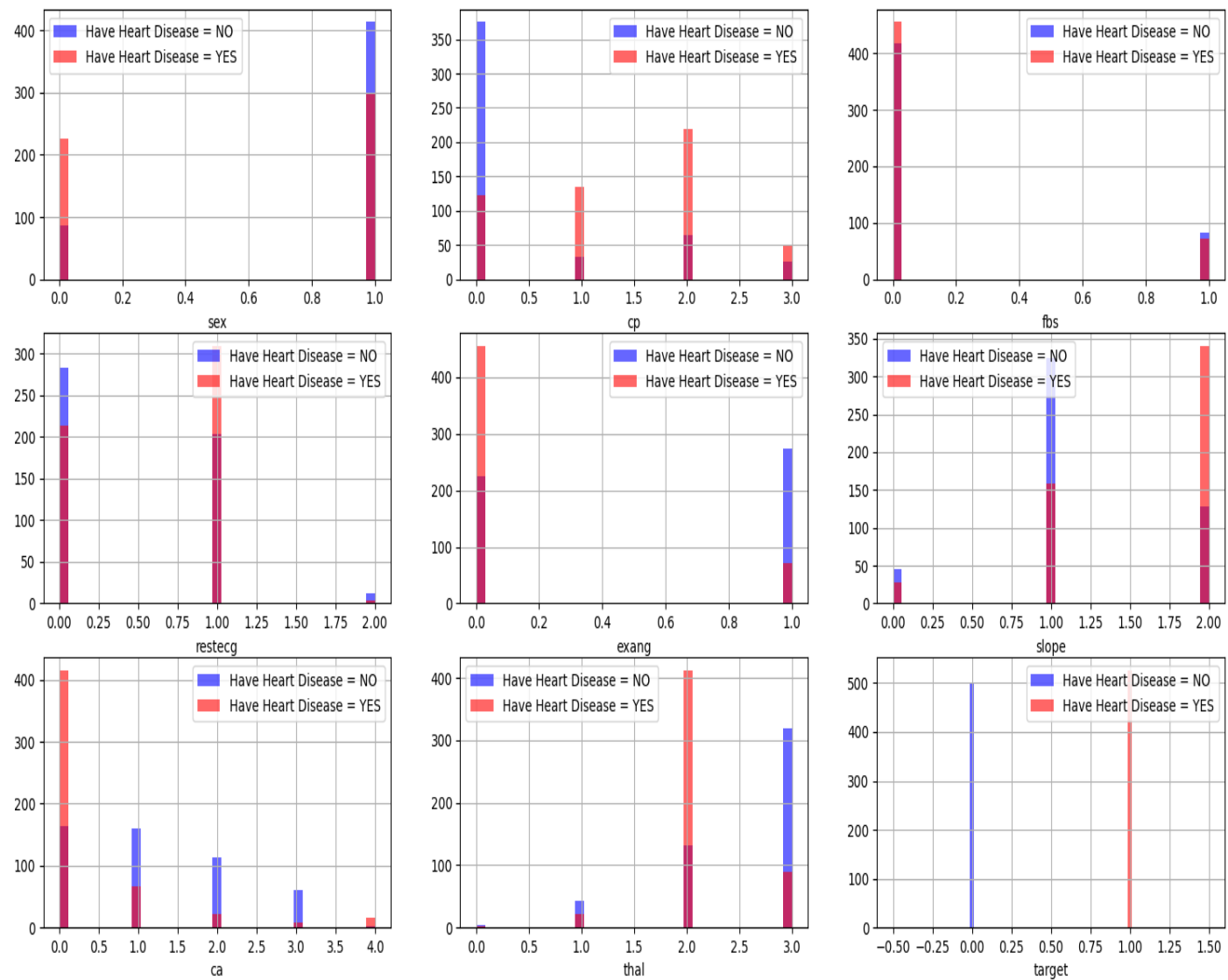
```
heart.csv
  age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang  oldpeak  slope  ca  thal  target
0   52   1   0     125    212    0         1     168     0      1.0    2   2     3       0
1   53   1   0     140    203    1         0     155     1      3.1    0   0     3       0
2   70   1   0     145    174    0         1     125     1      2.6    0   0     3       0
3   61   1   0     148    203    0         1     161     0      0.0    2   1     3       0
4   62   0   0     138    294    1         1     106     0      1.9    1   3     2       0
First 10 rows from csv file:
```

Rysunek 1: Pierwsze pięć wierszy z pliku csv

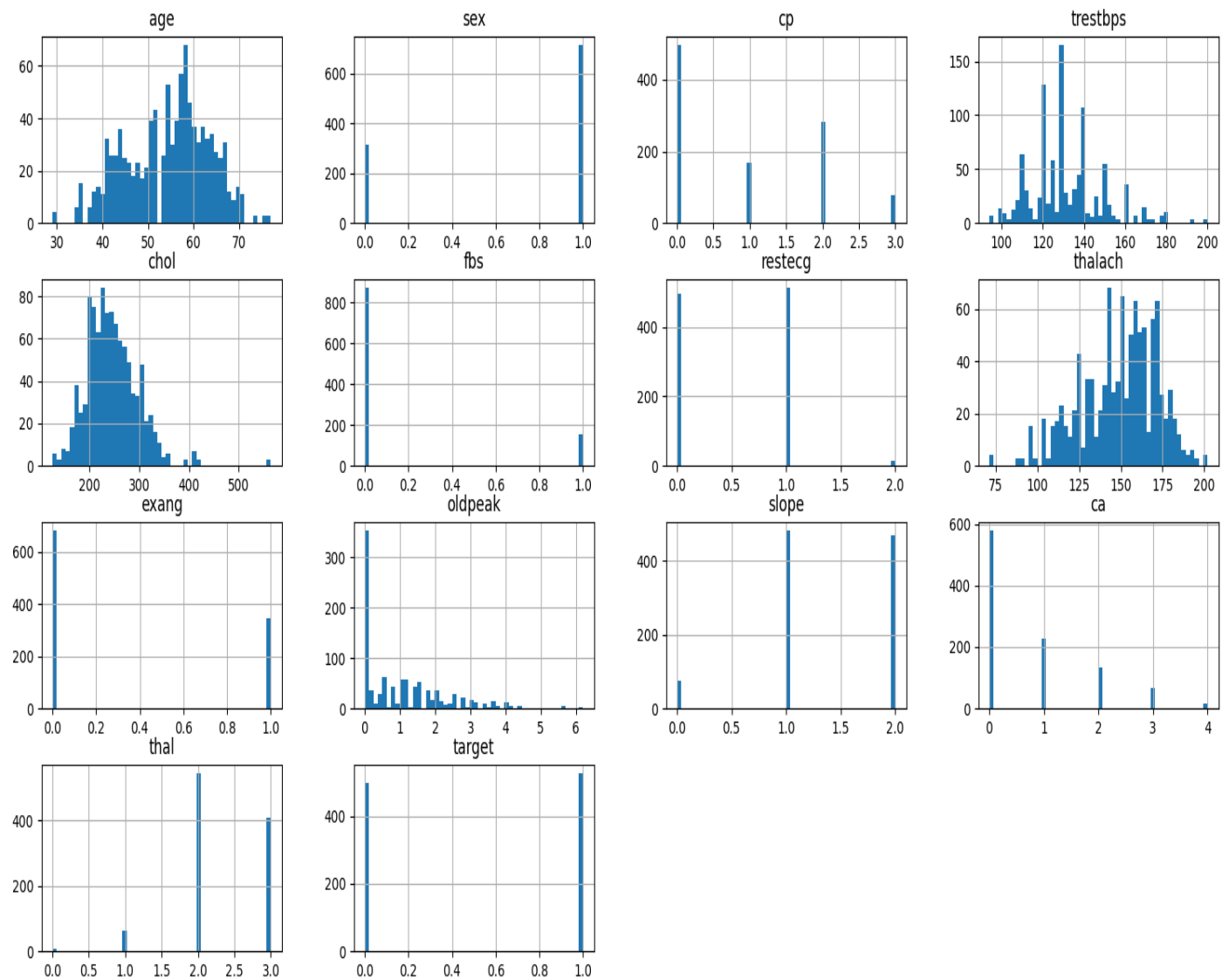
## Info

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 1025 entries, 0 to 1024  
Data columns (total 14 columns):  
#   Column      Non-Null Count  Dtype  
---  -  
0   age         1025 non-null   int64  
1   sex         1025 non-null   int64  
2   cp          1025 non-null   int64  
3   trestbps    1025 non-null   int64  
4   chol        1025 non-null   int64  
5   fbs         1025 non-null   int64  
6   restecg     1025 non-null   int64  
7   thalach     1025 non-null   int64  
8   exang       1025 non-null   int64  
9   oldpeak     1025 non-null   float64  
10  slope       1025 non-null   int64  
11  ca          1025 non-null   int64  
12  thal        1025 non-null   int64  
13  target      1025 non-null   int64  
dtypes: float64(1), int64(13)  
memory usage: 112.2 KB  
None
```

Rysunek 2: Podstawowe informacje o danych zawartych w pliku csv



Rysunek 3: Histogramy poszczególnych cech z uwzględnieniem zachorowania

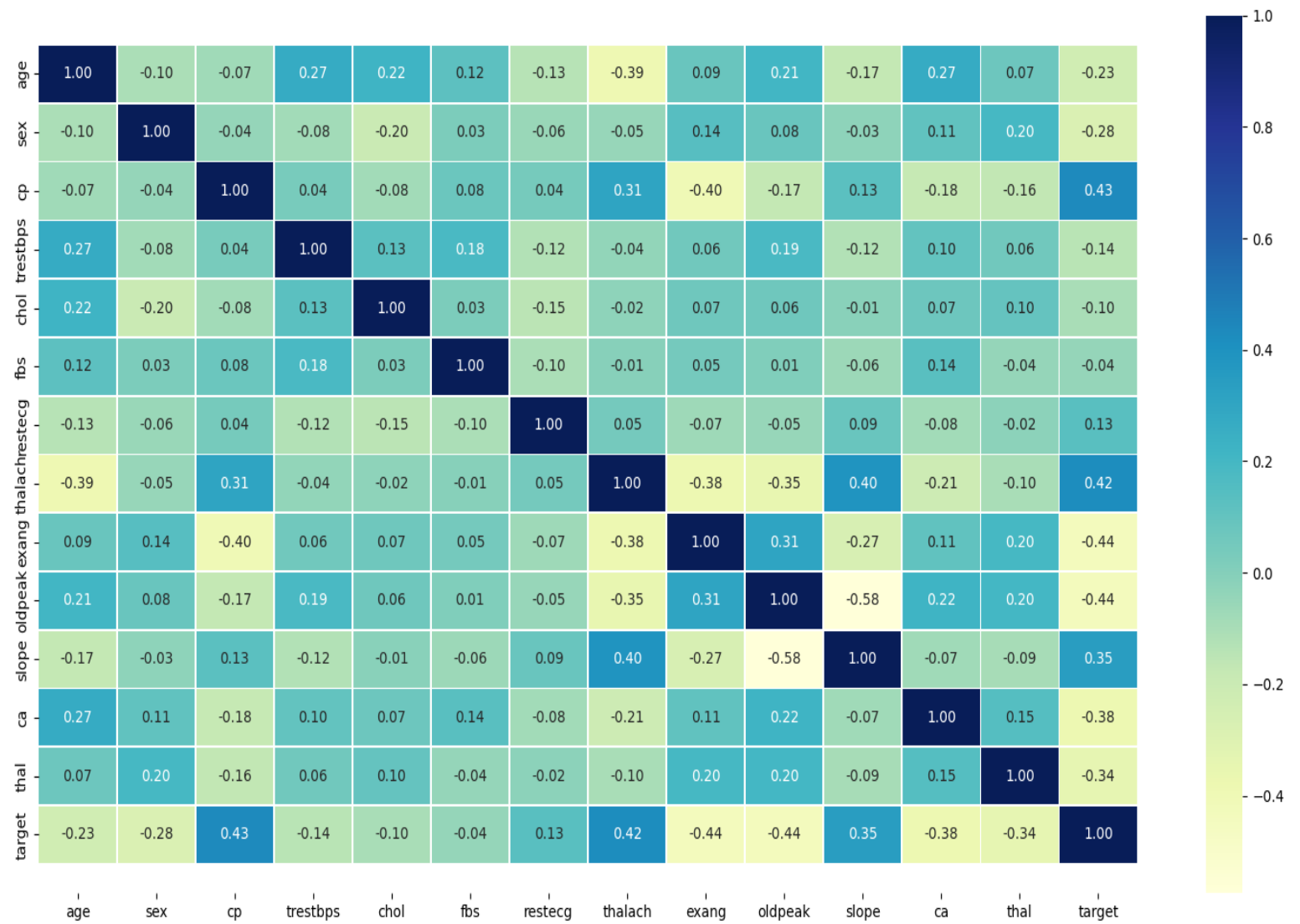


Rysunek 4: Histogramy cech zawartych



Rysunek 5: Choroba serca w funkcji wieku i maksymalnego tętna





Rysunek 6: Macierz korelacji

## 3 Opis programu

Na potrzeby projektu stworzono program pozwalający na:

- Wyświetlenie właściwości zbioru danych zawartych w pliku csv w postaci histogramów i wykresów oraz w formie tekstu.
- Naukę modelu za pomocą poniższych algorytmów.
  - DecisionTreeClassifier
  - SVC
  - KNeighborsClassifier
- Wizualizację procesu nauczania.
- Zapisywanie wyuczonego modelu.
- Obliczenie podstawowej miary jakości wyuczonego modelu.
- Wykonanie predykcji na podstawie danych zawartych w pliku csv oraz na podstawie danych wprowadzonych ręcznie poprzez terminal.

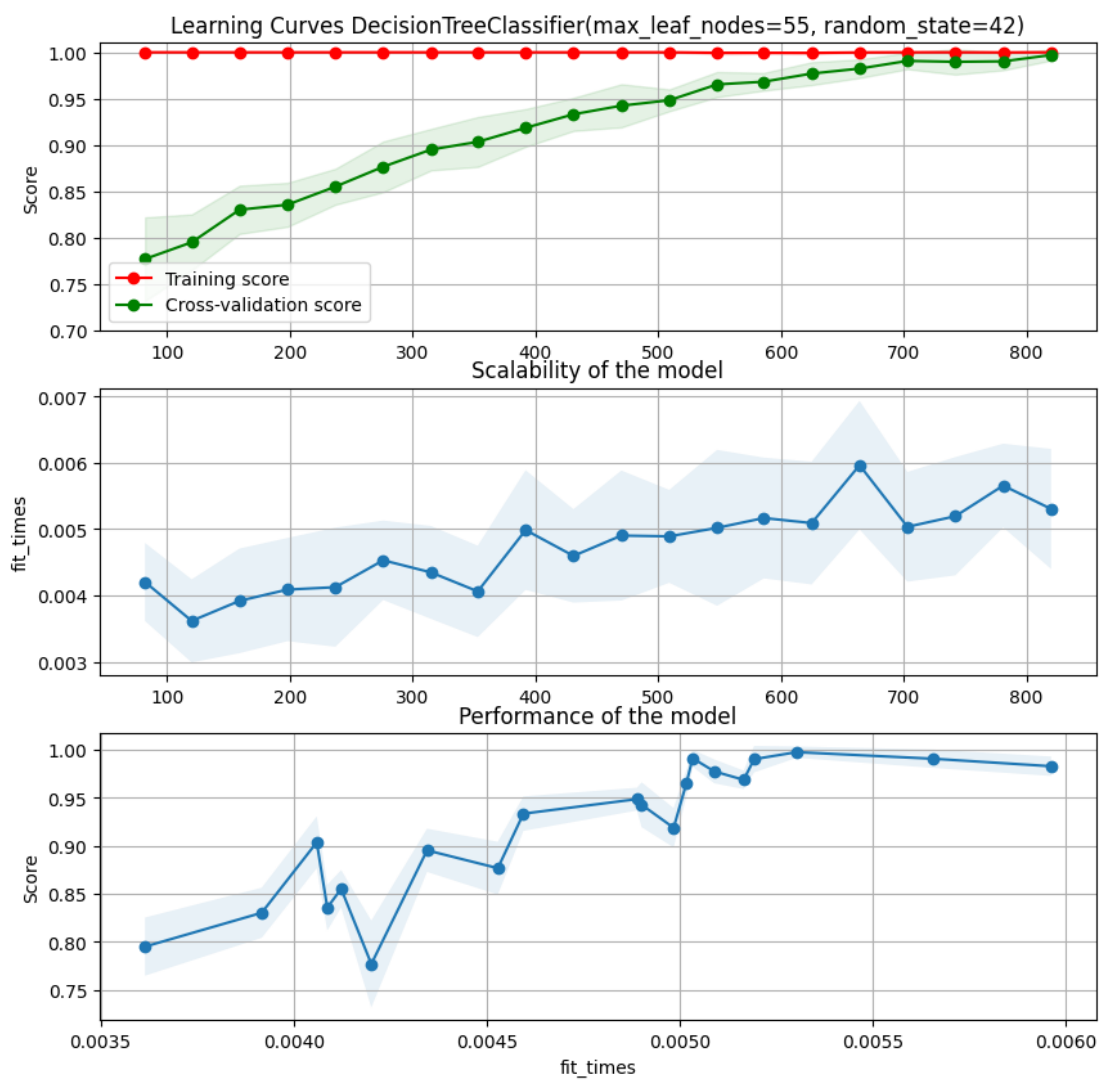
### 3.1 Proces uczenia modelu

Program pozwala na naukę modelu za pomocą trzech wcześniej wymienionych algorytmów. Proces nauki składa się również z etapu przygotowania danych, który polega na konwertowaniu wartości kategorycznych na wskaźniki. Dodatkowo pozostałe wartości są skalowane za pomocą **StandardScaler()**.

Użytkownik chcąc stworzyć model zostaje zapytany o stosunek zbioru testowego do treningowego. Następnie program dokonuje nauki modelu za pomocą wybranego algorytmu. Dobór parametrów zostaje wykonany automatycznie za pomocą funkcji **GridSearchCV**. Po zakończeniu procesu wyszukiwania hiperparametrów program wyświetla krzywe uczenia w celu wizualizacji tego procesu oraz oblicza miarę jakości działania modelu na zbiorze testowym i treningowym. Informacja o wybranych parametrach zostaje wyświetlona na wyjściu standardowym oraz w tytule obrazu zawierającego krzywe uczenia.

Poniżej przedstawiono efekty działania programu dla tego samego zbioru danych uczących, testowych i różnych algorytmów.

- DecisionTreeClassifie



Rysunek 7: krzywe uczenia DecisionTreeClassifie

```

Test result:

#####

Accuracy: 98.05194805194806%

-----

Report:

```

	0	1	accuracy	macro avg	weighted avg
precision	0.963636	1.000000	0.980519	0.981818	0.981228
recall	1.000000	0.959732	0.980519	0.979866	0.980519
f1-score	0.981481	0.979452	0.980519	0.980467	0.980500
support	159.000000	149.000000	0.980519	308.000000	308.000000

```

-----

Confusion Matrix:
[[159  0]
 [ 6 143]]
#####

```

Rysunek 8: Raport Test

```

Results of training:

#####

Accuracy: 100.0%

-----

Report:

```

	0	1	accuracy	macro avg	weighted avg
precision	1.0	1.0	1.0	1.0	1.0
recall	1.0	1.0	1.0	1.0	1.0
f1-score	1.0	1.0	1.0	1.0	1.0
support	340.0	377.0	1.0	717.0	717.0

```

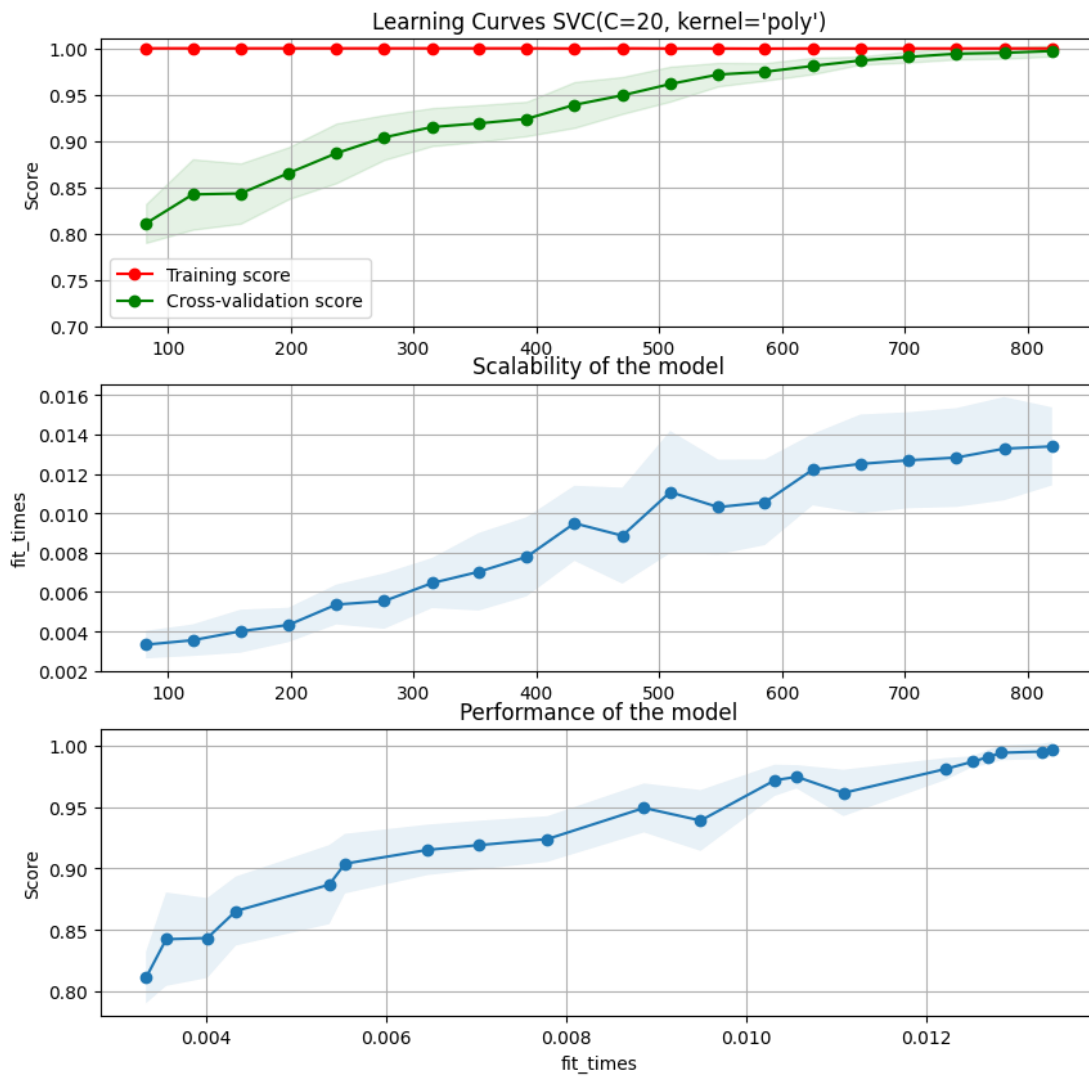
-----

Confusion Matrix:
[[340  0]
 [ 0 377]]
#####

```

Rysunek 9: Raport Treningowy

- SVC



Rysunek 10: krzywe uczenia SVC

```

Test result:

#####

Accuracy: 99.02597402597402%

-----

Report:

```

	0	1	accuracy	macro avg	weighted avg
precision	0.981481	1.000000	0.99026	0.990741	0.990440
recall	1.000000	0.979866	0.99026	0.989933	0.990260
f1-score	0.990654	0.989831	0.99026	0.990242	0.990256
support	159.000000	149.000000	0.99026	308.000000	308.000000

```

-----

Confusion Matrix:
[[159  0]
 [ 3 146]]
#####

```

Rysunek 11: Raport Test

```

Results of training:

#####

Accuracy: 100.0%

-----

Report:

```

	0	1	accuracy	macro avg	weighted avg
precision	1.0	1.0	1.0	1.0	1.0
recall	1.0	1.0	1.0	1.0	1.0
f1-score	1.0	1.0	1.0	1.0	1.0
support	340.0	377.0	1.0	717.0	717.0

```

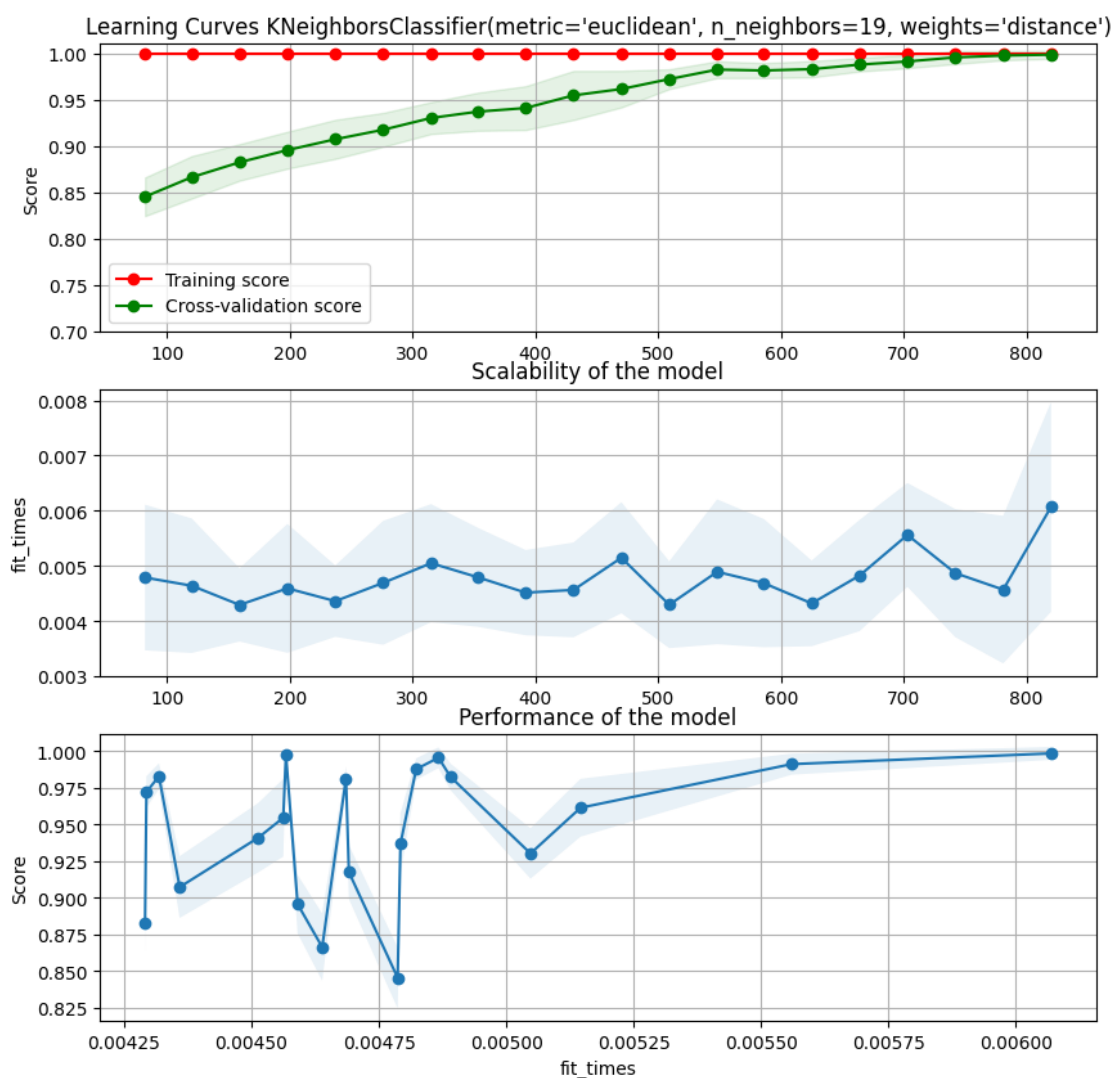
-----

Confusion Matrix:
[[340  0]
 [ 0 377]]
#####

```

Rysunek 12: Raport Treningowy

- KNeighborsClassifier



Rysunek 13: krzywe uczenia SVC

```

Test result:

#####

Accuracy: 100.0%

-----

Report:
           0         1  accuracy  macro avg  weighted avg
precision    1.0     1.0         1.0         1.0         1.0
recall       1.0     1.0         1.0         1.0         1.0
f1-score      1.0     1.0         1.0         1.0         1.0
support     159.0   149.0         1.0        308.0        308.0
-----

Confusion Matrix:
[[159   0]
 [  0 149]]
#####

```

Rysunek 14: Raport Test

```

Results of training:

#####

Accuracy: 100.0%

-----

Report:
           0         1  accuracy  macro avg  weighted avg
precision    1.0     1.0         1.0         1.0         1.0
recall       1.0     1.0         1.0         1.0         1.0
f1-score      1.0     1.0         1.0         1.0         1.0
support     340.0   377.0         1.0       717.0       717.0
-----

Confusion Matrix:
[[340   0]
 [  0 377]]
#####

```

Rysunek 15: Raport Treningowy



## 3.2 Działanie programu

Proces korzystania z programu polega na wywoływaniu odpowiednich pliku `mani.py` z odpowiednimi argumentami. Poniżej lista możliwych do wykonania operacji.

- `-h, --help` : wyświetlenie listy możliwych do użycia parametrów.
- `-p PATH, --path PATH`: ścieżka do pliku z danymi w formacie csv
- `-his, --histograms`: wyświetlenie histogramów związanych z danymi w pliku csv
- `-m, --more_text_info`: wyświetlenie informacji o zbiorze danych w postaci tekstu
- `-l tree,SVC,kne, --learn_model tree,SVC,kne`: proces nauki modelu za pomocą wybranego algorytmu
- `-s SAVE_MODEL, --save_model SAVE_MODEL`: zapisanie wyuczonego modelu do pliku `joblib`
- `-lm LOAD_MODEL, --load_model LOAD_MODEL`: wczytanie modelu zapisanego w pliku `joblib`

## 3.3 Przykładowe wywołania programu

```
wiktor@wiktor-Inspiron-5482:~/Desktop/MGR/ML$ python3 main.py -p heart.csv -his
```

Rysunek 16: Wyświetlenie informacji graficznych o zbiorze danych

```
wiktor@wiktor-Inspiron-5482:~/Desktop/MGR/ML$ python3 main.py -p heart.csv -m
```

Rysunek 17: Wyświetlenie informacji tekstowych o zbiorze danych

```
wiktor@wiktor-Inspiron-5482:~/Desktop/MGR/ML$ python3 main.py -p heart.csv -l tree
```

Rysunek 18: Proces uczenia modelu na podstawie wybranego algorytmu i zbioru danych

```
wiktor@wiktor-Inspiron-5482:~/Desktop/MGR/ML$ python3 main.py -p heart.csv -l tree -s model_1
```

Rysunek 19: Proces uczenia modelu na podstawie wybranego algorytmu i zbioru danych oraz zapisanie efektów do pliku `joblib`

```
wiktor@wiktor-Inspiron-5482:~/Desktop/MGR/ML$ python3 main.py -p heart.csv -lm model_1
```

Rysunek 20: Proces predykcji za pomocą wczytanego modelu. Predykcja zostaje dokonana na danych zawartych w pliku csv

```
wiktor@wiktor-Inspiron-5482:~/Desktop/MGR/ML$ python3 main.py -lm model_1
```

Rysunek 21: Proces predykcji za pomocą wczytanego modelu. Predykcja zostaje dokonana na danych wpisanych przez użytkownika