

ЛАБОРАТОРНА РОБОТА № 2

ПОРІВНЯННЯ МЕТОДІВ КЛАСИФІКАЦІЇ ДАНИХ

Мета роботи: використовуючи спеціалізовані бібліотеки та мову програмування Python дослідити різні методи класифікації даних та навчитися їх порівнювати.

Хід роботи

Завдання 1. Класифікація за допомогою машин опорних векторів (SVM)

```
F1 score: 56.15%  
  
Process finished with exit code 1
```

Завдання 2. Порівняння якості класифікаторів SVM з нелінійними ядрами

Поліноміальне ядро

```
classifier = OneVsOneClassifier(SVC(random_state=0, kernel='poly', degree=8))
```

Гаусове ядро

```
classifier = OneVsOneClassifier(SVC(random_state=0, kernel='rbf'))
```

```
F1 score: 71.95%  
  
Process finished with exit code 1
```

Сигмоїдальне ядро

```
classifier = OneVsOneClassifier(SVC(random_state=0, kernel='sigmoid'))
```

```
C:\Users\Admin\AppData\Local\Microsoft\WindowsApps\python3.10.exe D:/LabsPoli/AI/Lab2/LR_2_task_2_3.py  
F1 score: 63.77%
```

Завдання 3. Порівняння якості класифікаторів на прикладі класифікації сортів ірисів

Форма масиву та зріз даних

```

*****shape*****
(150, 5)
*****Зріз даних*****
  sepal-length  sepal-width  petal-length  petal-width      class
0           5.1           3.5           1.4           0.2  Iris-setosa
1           4.9           3.0           1.4           0.2  Iris-setosa
2           4.7           3.2           1.3           0.2  Iris-setosa
3           4.6           3.1           1.5           0.2  Iris-setosa
4           5.0           3.6           1.4           0.2  Iris-setosa
5           5.4           3.9           1.7           0.4  Iris-setosa
6           4.6           3.4           1.4           0.3  Iris-setosa
7           5.0           3.4           1.5           0.2  Iris-setosa
8           4.4           2.9           1.4           0.2  Iris-setosa
9           4.9           3.1           1.5           0.1  Iris-setosa
10          5.4           3.7           1.5           0.2  Iris-setosa
11          4.8           3.4           1.6           0.2  Iris-setosa
12          4.8           3.0           1.4           0.1  Iris-setosa
13          4.3           3.0           1.1           0.1  Iris-setosa
14          5.8           4.0           1.2           0.2  Iris-setosa
15          5.7           4.4           1.5           0.4  Iris-setosa
16          5.4           3.9           1.3           0.4  Iris-setosa
17          5.1           3.5           1.4           0.3  Iris-setosa
18          5.7           3.8           1.7           0.3  Iris-setosa
19          5.1           3.8           1.5           0.3  Iris-setosa

```

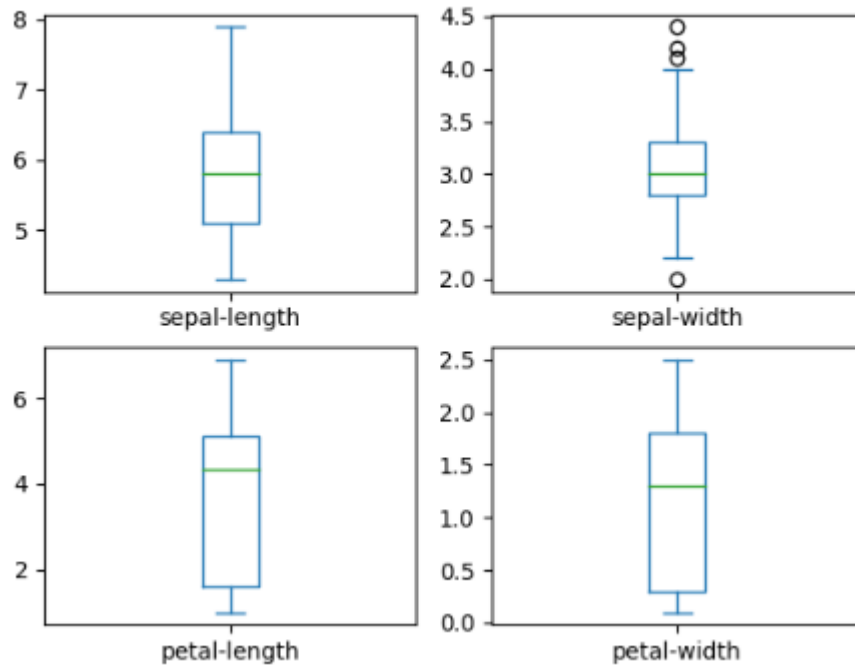
Статистичне зведення та розподіл за класом.

```

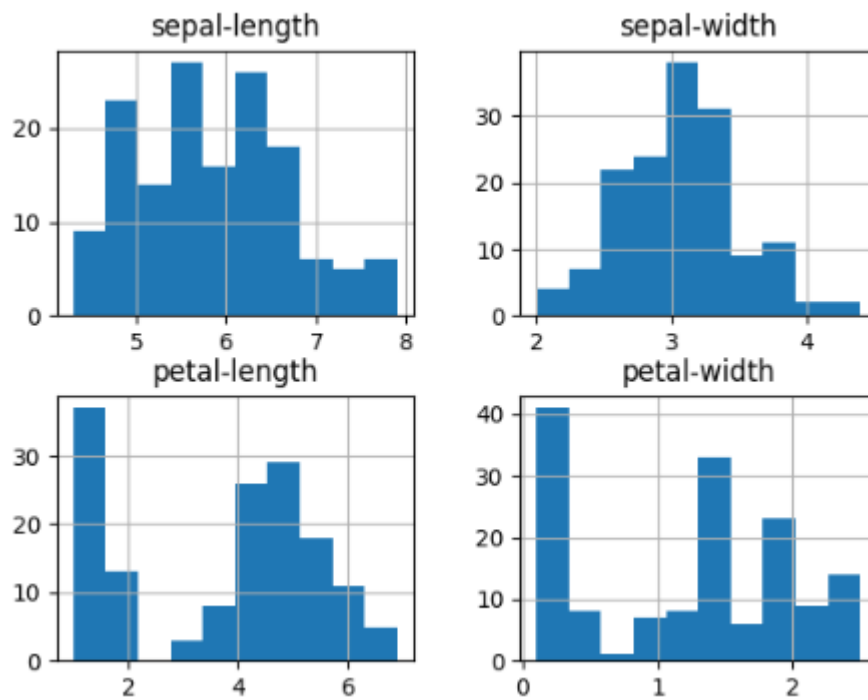
*****Опис даних*****
  sepal-length  sepal-width  petal-length  petal-width
count    150.000000    150.000000    150.000000    150.000000
mean       5.843333     3.054000     3.758667     1.198667
std        0.828066     0.433594     1.764420     0.763161
min        4.300000     2.000000     1.000000     0.100000
25%        5.100000     2.800000     1.600000     0.300000
50%        5.800000     3.000000     4.350000     1.300000
75%        6.400000     3.300000     5.100000     1.800000
max        7.900000     4.400000     6.900000     2.500000
*****Розподіл за класом*****
class
Iris-setosa      50
Iris-versicolor  50
Iris-virginica   50
dtype: int64

```

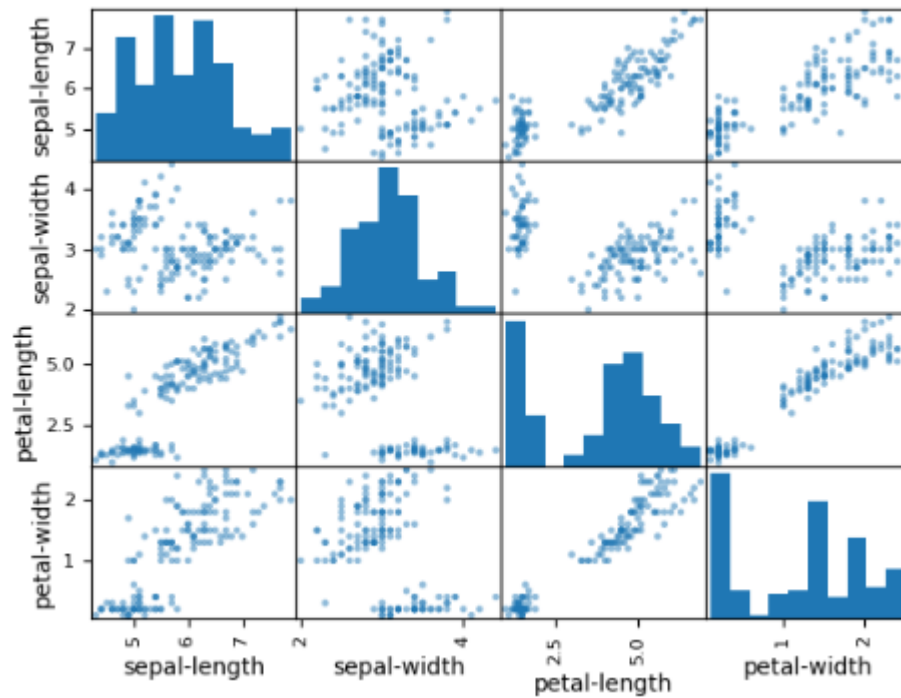
Діаграма розмаху



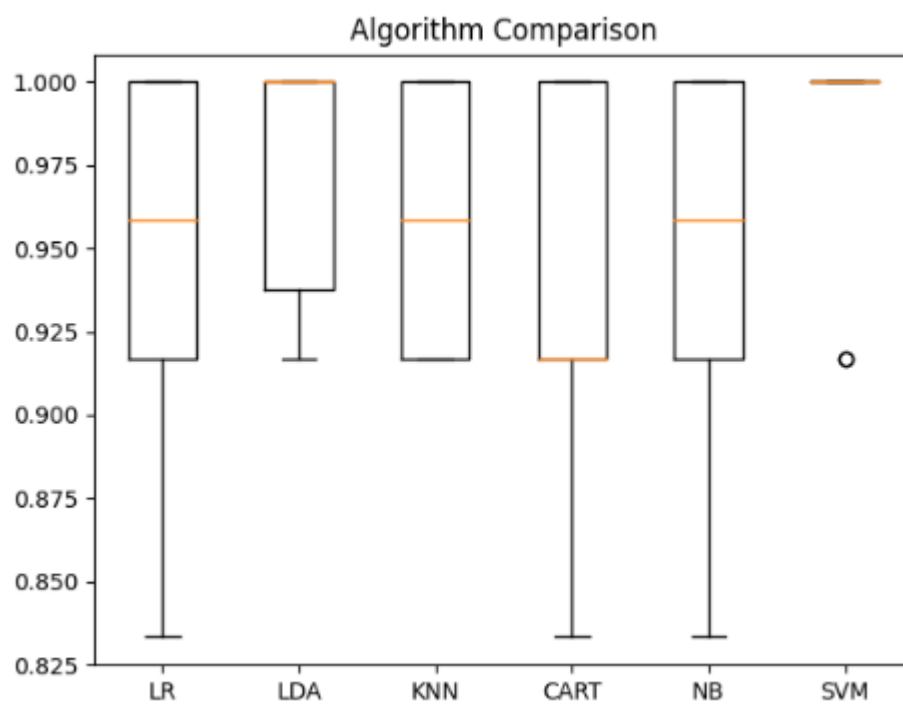
Гістограма розподілу атрибутів датасета



Матриця діаграм розсіювання



Порівняння алгоритмів



```

*****Оцінка моделей*****
LR: 0.941667 (0.065085)
LDA: 0.975000 (0.038188)
KNN: 0.958333 (0.041667)
CART: 0.941667 (0.053359)
NB: 0.950000 (0.05277)
SVM: 0.983333 (0.033333)

```

Найкращий результат має алгоритм SVM.

Мій прогноз

```
from sklearn.neighbors import KNeighborsClassifier
import numpy as np
knn = KNeighborsClassifier(n_neighbors=1)

knn.fit(X_train, Y_train)
X_new = np.array([[5, 2.9, 1, 0.2]])
print("Форма масиву X_new: {}".format(X_new.shape))
prediction = knn.predict(X_new)
print("Прогноз: {}".format(prediction))
print("Спрогнозована метка: {}".format(dataset['class']))
```

```
Форма масиву X_new: (1, 4)
Прогноз: ['Iris-setosa']
Спрогнозована метка: 0      Iris-setosa
```

Квітка належить до класу Iris-setosa

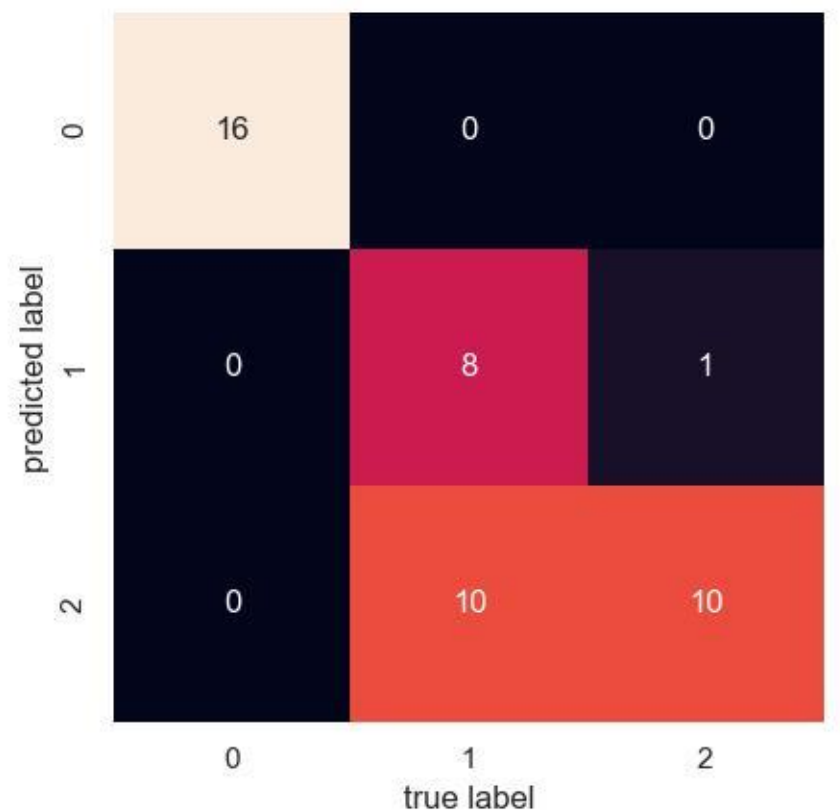
Завдання 4. Класифікація даних лінійним класифікатором Ridge

```
Accuracy: 0.7556
Precision: 0.8333
Recall: 0.7556
F1 Score: 0.7503
Cohen Kappa Score: 0.6431
Matthews Corrccoef: 0.6831

Classification Report:

```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	16
1	0.44	0.89	0.59	9
2	0.91	0.50	0.65	20
accuracy			0.76	45
macro avg	0.78	0.80	0.75	45
weighted avg	0.85	0.76	0.76	45



Ця картинка показує, що програма 16 із 16 класів 0, 8 з 18 класів 1 та 10 з 11 класів 2.

Посилання на Git: <https://github.com/Grum74/AI>

Висновок

Я, використовуючи спеціалізовані бібліотеки та мову програмування Python дослідив різні методи класифікації даних та навчився їх порівнювати.