

Statystyczne metody rozpoznawania obrazu

Zadanie 1

Bartłomiej Gorzela

1) Wypisać dane zaczytane z repozytorium.

Do wykonania tego zadania posłużyłem się zestawem danych 'Iris' możliwym do zainportowania bezpośrednio z biblioteki scikit-learn. Biblioteką pandas posłużyłem się do wizualizacji zaczytanych danych.

Kod źródłowy:

```
Task_1_1.py X
StatisticalPatternRecognitionMethods > Task1 > Task_1_1.py > ...
1  #ZAD 1_1    Wypisać dane zaczytane z repozytorium
2  from sklearn.datasets import load_iris
3  import pandas as pd
4
5  # Wczytanie danych Iris z sklearn
6  iris = load_iris()
7  # Konwersja do DataFrame dla czytelności
8  data = pd.DataFrame(data=iris.data, columns=iris.feature_names)
9  data['class'] = iris.target
10 print("Dane zaczytane z repozytorium:")
11 print(data)
```

Wynik z konsoli:

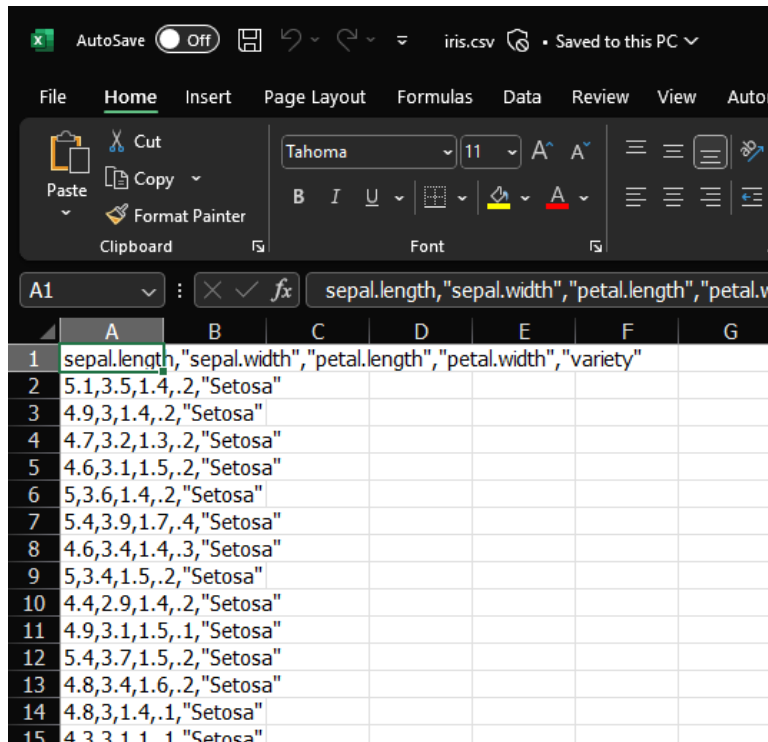
```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS
Python - StatisticalPatternRecognitionMethods
PS D:\Studia\StatisticalPatternRecognitionMethods> & C:/toolsx64/Python37/python.exe d:/Studia/StatisticalPatternRecognitionMethods/Task1/Task_1_1.py
• Dane zaczytane z repozytorium:
  sepal length (cm)  sepal width (cm)  petal length (cm)  petal width (cm)  class
0                5.1                3.5                1.4                0.2        0
1                4.9                3.0                1.4                0.2        0
2                4.7                3.2                1.3                0.2        0
3                4.6                3.1                1.5                0.2        0
4                5.0                3.6                1.4                0.2        0
..                ...                ...                ...                ...        ...
145               6.7                3.0                5.2                2.3        2
146               6.3                2.5                5.0                1.9        2
147               6.5                3.0                5.2                2.0        2
148               6.2                3.4                5.4                2.3        2
149               5.9                3.0                5.1                1.8        2

[150 rows x 5 columns]
PS D:\Studia\StatisticalPatternRecognitionMethods>
```

2) Wypisać dane pobrane z pliku .csv.

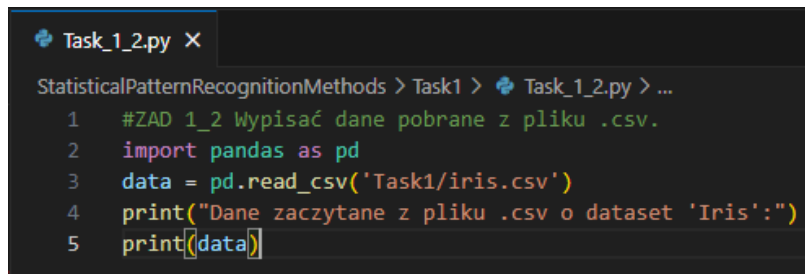
Ten sam zestaw danych został pobrany w formacie .csv i zaimportowany za pomocą biblioteki pandas.

Plik .csv:



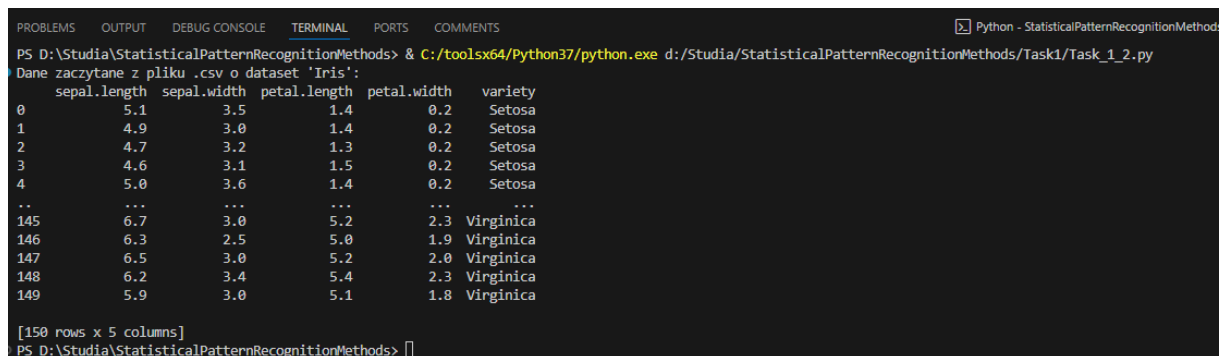
	A	B	C	D	E	F	G
1	sepal.length	sepal.width	petal.length	petal.width	variety		
2	5.1	3.5	1.4	.2	Setosa		
3	4.9	3.1	1.4	.2	Setosa		
4	4.7	3.2	1.3	.2	Setosa		
5	4.6	3.1	1.5	.2	Setosa		
6	5.3	6.1	4.7	.2	Setosa		
7	5.4	3.9	1.7	.4	Setosa		
8	4.6	3.4	1.4	.3	Setosa		
9	5.3	4.1	5.2	.2	Setosa		
10	4.4	2.9	1.4	.2	Setosa		
11	4.9	3.1	1.5	.1	Setosa		
12	5.4	3.7	1.5	.2	Setosa		
13	4.8	3.4	1.6	.2	Setosa		
14	4.8	3.1	1.4	.1	Setosa		
15	4.3	3.1	1.1	.1	Setosa		

Kod źródłowy:



```
Task_1_2.py X
StatisticalPatternRecognitionMethods > Task1 > Task_1_2.py > ...
1 #ZAD 1_2 Wypisać dane pobrane z pliku .csv.
2 import pandas as pd
3 data = pd.read_csv('Task1/iris.csv')
4 print("Dane załczytane z pliku .csv o dataset 'Iris':")
5 print(data)
```

Wynik z konsoli:



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS Python - StatisticalPatternRecognitionMethod
PS D:\Studia\StatisticalPatternRecognitionMethods> & C:/toolsx64/Python37/python.exe d:/Studia/StatisticalPatternRecognitionMethods/Task1/Task_1_2.py
Dane załczytane z pliku .csv o dataset 'Iris':
  sepal.length  sepal.width  petal.length  petal.width  variety
0          5.1          3.5          1.4          0.2    Setosa
1           4.9          3.0          1.4          0.2    Setosa
2           4.7          3.2          1.3          0.2    Setosa
3           4.6          3.1          1.5          0.2    Setosa
4           5.0          3.6          1.4          0.2    Setosa
..          ...          ...          ...          ...    ...
145          6.7          3.0          5.2          2.3  Virginica
146          6.3          2.5          5.0          1.9  Virginica
147          6.5          3.0          5.2          2.0  Virginica
148          6.2          3.4          5.4          2.3  Virginica
149          5.9          3.0          5.1          1.8  Virginica

[150 rows x 5 columns]
PS D:\Studia\StatisticalPatternRecognitionMethods>
```

3) Sztucznie wygenerować dane, np. dwuwymiarowe.

W celu wykonania tego zadania posłużyłem się używaną wcześniej biblioteką scikit-learn do stworzenia sztucznych danych oraz matplotlib w celu ich wizualizacji:

Kod źródłowy:

```
Task_1_3.py X
StatisticalPatternRecognitionMethods > Task1 > Task_1_3.py > ...
1  #ZAD 1_3 Sztucznie wygenerować dane, np. dwuwymiarowe.
2  import matplotlib.pyplot as plt
3  from sklearn.datasets import make_blobs
4
5  data, labels = make_blobs(n_samples=200, centers=3, n_features=2)
6  # Wypisanie danych
7  print("Wygenerowane dane:")
8  print(data)
9  print("Etykiety klas:")
10 print(labels)
11 # Wizualizacja danych
12 plt.scatter(data[:, 0], data[:, 1], c=labels, cmap='plasma', marker='o')
13 plt.title("Sztucznie wygenerowane dane dwuwymiarowe")
14 plt.show()
```

Sztucznie wygenerowane dane:

