

**LAPORAN HASIL TUGAS KECIL 2 IF2211 STRATEGI ALGORITMA  
SEMESTER II TAHUN 2021/2022**

**Implementasi Convex Hull untuk Visualisasi Tes Linear Separability  
Dataset dengan Algoritma Divide and Conquer**



**Disusun oleh :**

**Steven**

**13520131**

**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA  
INSTITUT TEKNOLOGI BANDUNG  
BANDUNG**

**2022**

## DAFTAR ISI

|   |           |
|---|-----------|
| <b>BAB I - ALGORITMA DIVIDE AND CONQUER SECARA GARIS BESAR.....</b> | <b>3</b>  |
| <b>BAB II - KODE PROGRAM DALAM BAHASA PYTHON.....</b>               | <b>5</b>  |
| <b>BAB III - INPUT/OUTPUT PROGRAM .....</b>                         | <b>8</b>  |
| <b>BAB IV - ALAMAT KODE PROGRAM.....</b>                            | <b>11</b> |
| <b>BAB V - TABEL PENILAIAN.....</b>                                 | <b>12</b> |

# BAB I

## ALGORITMA DIVIDE AND CONQUER SECARA GARIS BESAR

### 1.1 Pengertian Algoritma *Divide and Conquer*

Algoritma *divide and conquer* merupakan algoritma yang membagi persoalan menjadi beberapa upa-persoalan lebih kecil yang memiliki kemiripan dengan persoalan semula. Cara kerja algoritma ini adalah dengan menyelesaikan upa-persoalan lalu menggabungkan solusi dari upa-persoalan tersebut.

Contoh persoalan yang dapat diselesaikan dengan memanfaatkan algoritma *divide and conquer*

1. Persoalan MinMaks
2. Menghitung perpangkatan
3. *Sorting (Mergesort & Quicksort)*
4. Mencari pasangan titik terdekat
5. *Convex Hull*
6. Perkalian matriks
7. Perkalian bilangan bulat besar
8. Perkalian dua buah polinom

### 1.2 *Convex Hull* dan Kaitannya dengan Algoritma *Divide and Conquer*

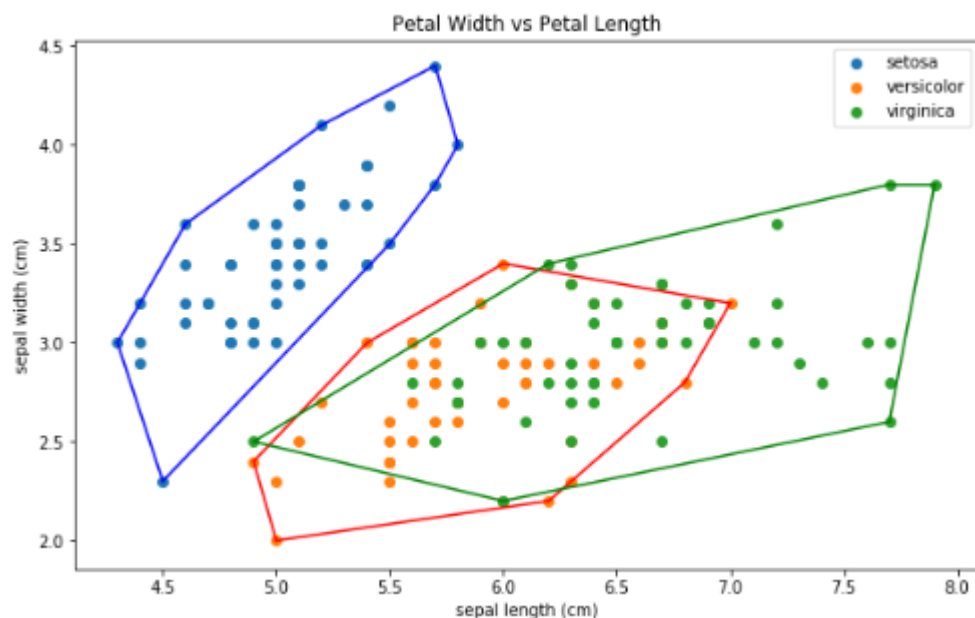
*Convex Hull* merupakan sebuah pembungkusan dari kumpulan titik. Syarat suatu bidang dikatakan *convex* adalah jika untuk sembarang pasang titik pada bidang tersebut, seluruh segmen garis tetap berada pada himpunan tersebut

Dalam kaitannya dengan algoritma *divide and conquer*, *convex hull* diselesaikan dengan menggunakan ide dasar dari algoritma *quicksort*. Langkah-langkahnya adalah dengan mengurutkannya dengan prioritas berupa nilai absis yang menaik lalu nilai ordinat yang menaik. Setelah itu, tentukan 2 buah titik ekstrim yang akan menjadi acuan algoritma ini. Setelah itu, tarik garis untuk menghubungkan kedua buah titik ekstrim tersebut sedemikian sehingga terbentuk 2

buah area pada himpunan yang dimana pembentukan 2 buah area inilah yang disebut dengan *divide* dalam algoritma *divide and conquer*. Tidak hanya itu, semua titik yang berada pada garis ekstrim juga tidak perlu diperiksa karena sudah pasti tidak akan membentuk *convex hull*. Lakukan proses ini berulang untuk area yang terkena *division*. Setelah semua area telah diselesaikan, kembalikan pasangan titik yang dihasilkan.

### 1.3 Deskripsi Masalah *Convex Hull* pada Tugas Kecil 2 IF2211 Strategi Algoritma

Pada Tugas Kecil 2 IF2211 Strategi Algoritma, masalah yang diberikan adalah mengimplementasikan *convex hull* untuk visualisasi tes *linear separability dataset* dengan algoritma *divide and conquer* dengan spesifikasi sebagai berikut. Membuat sebuah pustaka *myConvexHull* dalam bahasa Python yang dapat mengembalikan *convex hull* dari kumpulan data 2 dimensi. Selanjutnya, gunakan pustaka tersebut ke dalam program visualisasi data dan *convex hull* berikut ini. Misalnya program menerima dataset berlabel, misalnya iris yang memiliki 150 baris dan 5 kolom. Program akan mengembalikan *convexhull* dari setiap label lalu menampilkannya dengan warna yang berbeda. Hasil dari *convex hull* data iris dapat dilihat pada gambar di bawah ini.



## BAB II

### KODE PROGRAM DALAM BAHASA PYTHON

#### 2.1 myconvexhull.py

Berisi class MyConvexHull yang merupakan implementasi dari pustaka *convex hull*.

```
1  from typing import List, Sequence, Tuple
2
3  class MyConvexHull:
4
5      # Menerima seluruh data berupa posisi dari titik-titik
6      def __init__(self, points):
7          # Mengubah bentuk data ke generator of tuple
8          # sehingga dapat diolah
9          param = ((i[0], i[1]) for i in points)
10         res = self.make_hull(sorted(param))
11         self.x_coords = [i[0] for i in res]
12         self.x_coords.append(res[0][0])
13         self.y_coords = [i[1] for i in res]
14         self.y_coords.append(res[0][1])
15
16         # Membuat Convex Hull
17         def make_hull(self, points):
18             # Menangani edgecase
19             if len(points) <= 1:
20                 return List(points)
21
22             # Membuat array untuk menampung hasil dari
23             # pembagian area akibat algoritma divide and conquer
24             area_one = []
25             area_two = []
26
27             # Melakukan conquer sesuai dengan
28             # pengertian dari convex hull
29             for area in (area_one, area_two):
30                 if (area != area_one):
31                     points = reversed(points)
32
33                 for p in points:
34                     while len(area) >= 2:
35                         x_one, y_one = area[-1]
```

```

36         x_two, y_two = area[-2]
37         check1 = (x_one - x_two) * (p[1] - y_two)
38         check2 = (y_one - y_two) * (p[0] - x_two)
39         if (check1 >= check2):
40             area.pop()
41         else:
42             break
43         area.append(p)
44
45     area.pop()
46
47     if (len(area_one) == 1):
48         if (area_one == area_two):
49             area_two = []
50
51     # Menggabungkan hasil dari kedua buah area
52     area_one += area_two
53
54     # Mengembalikan hasil
55     return area_one

```

## 2.2 main.py

Merupakan program utama yang akan memanggil pustaka MyConvexHull lalu menerima input berupa dataset dan mengembalikan *cconvex hull* dari dataset tersebut.

```

1  import numpy as np
2  import pandas as pd
3  import matplotlib.pyplot as plt
4  from sklearn import datasets
5
6  from myconvexhull import MyConvexHull
7
8  # Untuk checking hasil
9  # from scipy.spatial import ConvexHull
10
11 if __name__ == "__main__":
12     print("List dataset yang tersedia:")
13     print("1. Iris")
14     print("2. Wine")
15     print("3. Breast Cancer")
16     option = int(input("Input dataset yang diinginkan: "))
17
18     if (option==1):
19         data = datasets.load_iris()
20         title = "Iris Dataset"
21     elif (option==2):
22         data = datasets.load_wine()
23         title = "Wine Dataset"
24     elif (option==3):
25         data = datasets.load_breast_cancer()
26         title = "Breast Cancer Dataset"
27     else:
28         print("Dataset tidak tersedia")
29         exit()

```

```

30
31 #create a DataFrame
32 df = pd.DataFrame(data.data, columns=data.feature_names)
33 df['Target'] = pd.DataFrame(data.target)
34 # print(df)
35 df.head()
36 plt.figure(figsize = (10, 6))
37 colors = ['b','r','g', 'c', 'm', 'y', 'k', 'w', 'aquamarine', 'mediumseagreen']
38 plt.title(title)
39 plt.xlabel(data.feature_names[0])
40 plt.ylabel(data.feature_names[1])
41 for i in range(len(data.target_names)):
42     bucket = df[df['Target'] == i]
43     bucket = bucket.iloc[:,[0,1]].values
44     # hull = ConvexHull(bucket)
45     # print(bucket)
46     hull = MyConvexHull(bucket)
47     plt.plot(hull.x_coords, hull.y_coords, colors[i])
48     plt.scatter(bucket[:, 0], bucket[:, 1], label=data.target_names[i])
49     # plt.plot(hull[0], hull[1], colors[i])
50     # print(hull.simplices)
51     # for simplex in hull.simplices:
52     #     print(simplex)
53     #     plt.plot(bucket[simplex, 0], bucket[simplex, 1], colors[i])
54 plt.legend()
55 plt.show()

```

## BAB III

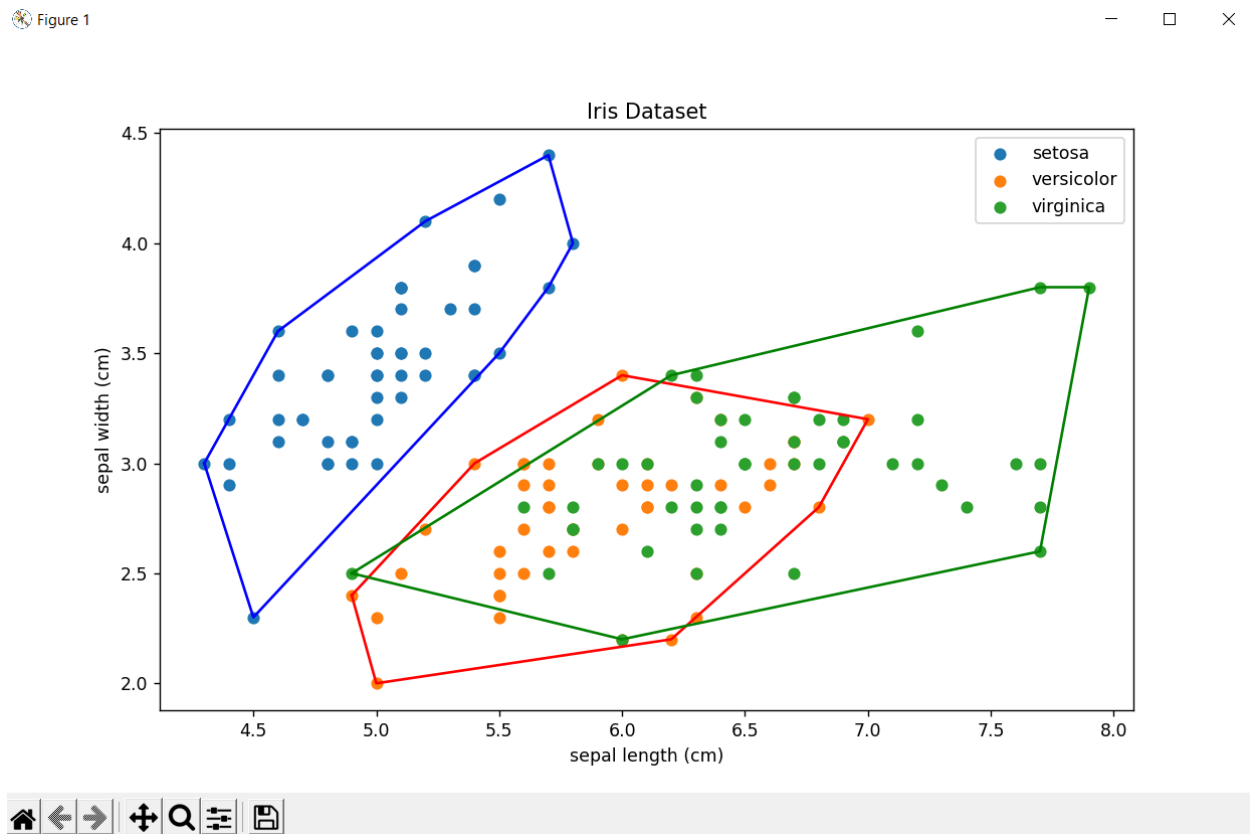
### INPUT/OUTPUT PROGRAM

#### Test Case 1 (Dataset Iris)

##### Input

```
(venv) PS C:\Users\Asus\desktop\github\Tucil-2-Stima-13520131\src> py main.py
List dataset yang tersedia:
1. Iris
2. Wine
3. Breast Cancer
Input dataset yang diinginkan: 1
```

##### Output



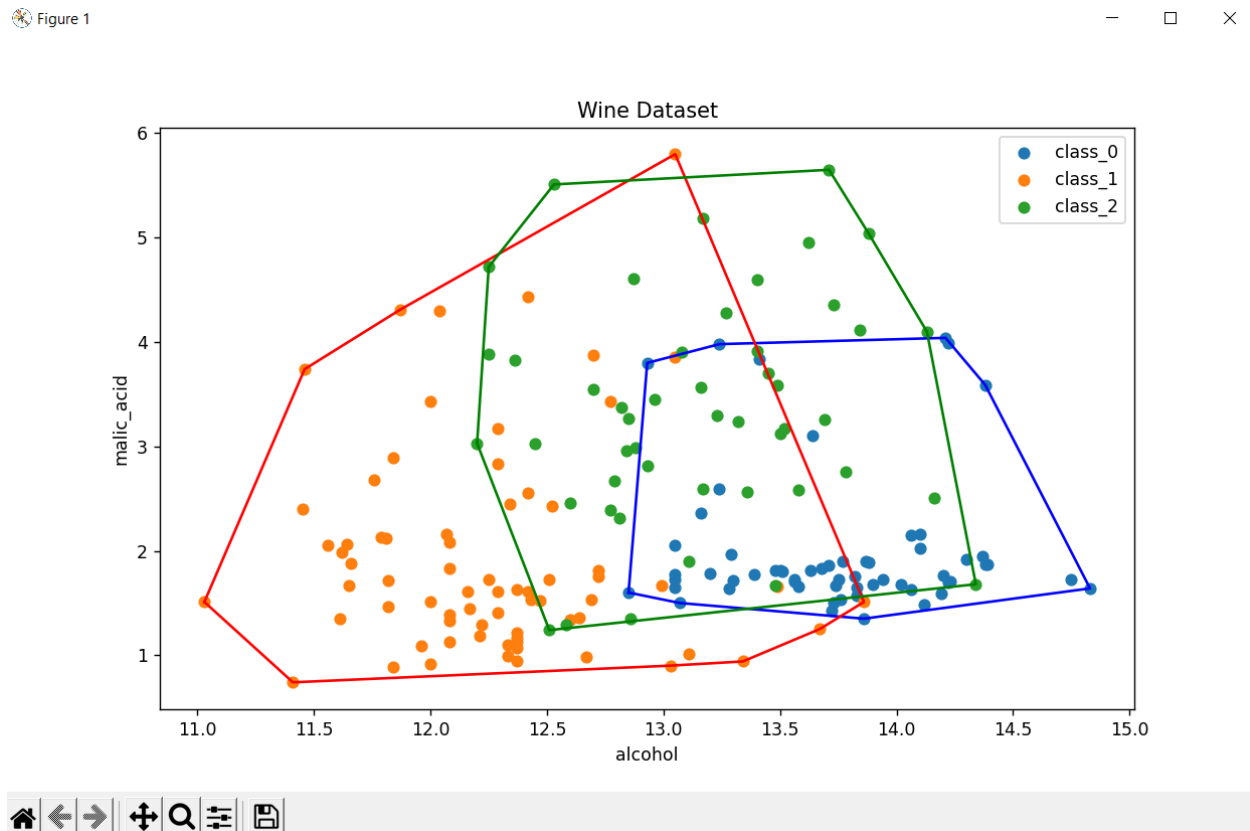


## Test Case 2 (Dataset Wine)

### Input

```
(venv) PS C:\Users\Asus\desktop\github\Tucil-2-Stima-13520131\src> py main.py
List dataset yang tersedia:
1. Iris
2. Wine
3. Breast Cancer
Input dataset yang diinginkan: 2
```

### Output



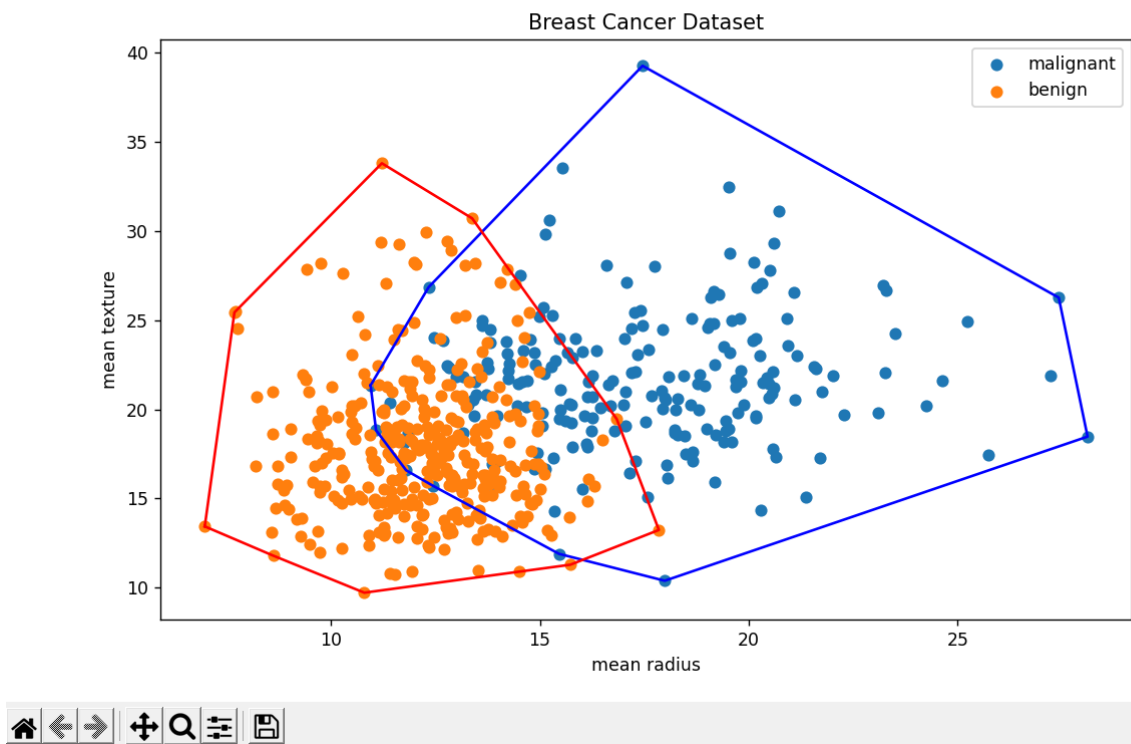
## Test Case 3 (Dataset Breast Cancer)

### Input

```
(venv) PS C:\Users\Asus\desktop\github\Tucil-2-Stima-13520131\src> py main.py
List dataset yang tersedia:
1. Iris
2. Wine
3. Breast Cancer
Input dataset yang diinginkan: 3
```

## Output

Figure 1



### Test Case 4 (Dataset random)

#### Input & Output

```
(venv) PS C:\Users\Asus\desktop\github\Tucil-2-Stima-13520131\src> py main.py
List dataset yang tersedia:
1. Iris
2. Wine
3. Breast Cancer
Input dataset yang diinginkan: 999
Dataset tidak tersedia
```

**BAB IV**  
**ALAMAT KODE PROGRAM**

<https://github.com/loopfree/Tucil-2-Stima-13520131>

## BAB V

### TABEL PENILAIAN

| Poin   | Ya | Tidak |
|--|----|-------|
| 1. Pustaka MyConvexHull berhasil dibuat dan tidak ada kesalahan  | √  |       |
| 2. <i>Convex hull</i> yang dihasilkan sudah benar  | √  |       |
| 3. Pustaka MyConvexHull dapat digunakan untuk menampilkan <i>convex hull</i> dengan warna yang berbeda | √  |       |
| 4. <b>BONUS:</b> program dapat menerima input dan menuliskan output untuk dataset lainnya.             | √  |       |