

LAPORAN TUGAS KECIL 2

IF2211 STRATEGI ALGORITMA

Atabik Muhammad Azfa Shofi – 13520159

[Link Repository](#)

1. Algoritma Divide n Conquer pada myConvexHull

Algoritma divide and conquer adalah algoritma yang idenya adalah membagi persoalan menjadi sub persoalan lalu digabung kembali setelah solusi ditemukan. Salah satu penerapannya dapat dilakukan pada program convex-hull. Pada tugas kecil kali ini, penulis diminta untuk membuat sendiri fungsi myConvexHull untuk melakukan convexhull kepada dataset yang sudah diberikan.

Dalam fungsi myConvexHull yang penulis buat, idenya adalah menentukan titik ekstrem maks dan min dari input array of point lalu membagi titik-titik lain sesuai posisinya terhadap garis pMin-pMax. Titik yang berada di atas akan berada di subarray atas dan sebaliknya. Lalu, untuk tiap sub bagian ditentukan titik maksnya yang selanjutnya dari kumpulan titik-titik tersebut dan titik maksnya, dapat dilakukan subHull untuk tiap sub bagian.

Subhull perlu dibagi menjadi atas dan bawah karena untuk setiap sub bagian akan terus melakukan hull ke arah yang sama terus menerus hingga ditemukan titik-titik terluarnya. Misal, subhull bagian atas akan terus melakukan subhull atas untuk tiap subbagiannya dan begitupun sebaliknya. Setelah ditemukan solusi dari sub atas dan bawah, maka fungsi myConvexHull akan mengeluarkan hasil sub tersebut masing-masing ke arrOutUpper dan arrOutBottom.

Return fungsi perlu dibagi untuk mempermudah melakukan plot data, sehingga plot data hanya perlu iterasi dari indeks 0 hingga indeks terakhir - 1, pada tiap iterasi akan melakukan plot titik array[indeks] dengan array[indeks+1], masing-masing dilakukan pada arrOutUpper dan arrOutBottom.

Contoh penggunaan myConvexHull dan plotnya sebagai berikut:

```
hullUpper, hullBottom = myConvexHull(bucket)
idx = 0
while idx < len(hullUpper)-1:
    p1 = hullUpper[idx]
    p2 = hullUpper[idx+1]
    plt.plot([p1[0],p2[0]], [p1[1],p2[1]], colors[i])
    idx += 1
idx = 0
while idx < len(hullBottom)-1:
    p1 = hullBottom[idx]
    p2 = hullBottom[idx+1]
    plt.plot([p1[0],p2[0]], [p1[1],p2[1]], colors[i])
    idx += 1
```

2. Source-code Program

Source-code utama program myConvexHull berada pada file myConvexHull.py, di dalamnya terdapat beberapa fungsi lainnya untuk membantu fungsi utama myConvexHull, isi source-code dapat dilihat sendiri pada file program karena akan memakan banyak ruang pada laporan. Namun, secara garis besar fungsi-fungsinya dapat dijelaskan sebagai berikut :

- myConvexHull(array) -> arrayUpper, arrayBottom:
fungsi myConvexHull adalah fungsi utama untuk menentukan garis utama pointMin ke pointMax, yang didapat dari elemen pertama dan terakhir dari array yang sudah disort sesuai point. Lalu, fungsi akan membagi-membagi titik yang berada di atas garis ke arrayUpper dan titik di bawah garis ke arrayBottom. Dari seluruh titik di array masing-masing akan dicari titik maksimumnya terhadap garis pMin-pMax, lalu dilakukan subHull atas (boolean=true) pada arrayUpper dan subHull bawah (boolean=false) pada arrayBottom. Fungsi akan mengembalikan arrayUpper dan arrayBottom setelah dilakukan subHull.
- subHull(array, pointMax, boolean) -> arrayHull:
fungsi subHull dibuat untuk membedakan hull yang perlu dilakukan pada bagian atas dan bawah garis sesuai masukan boolean, true untuk atas dan false untuk bawah. Lalu, fungsi akan

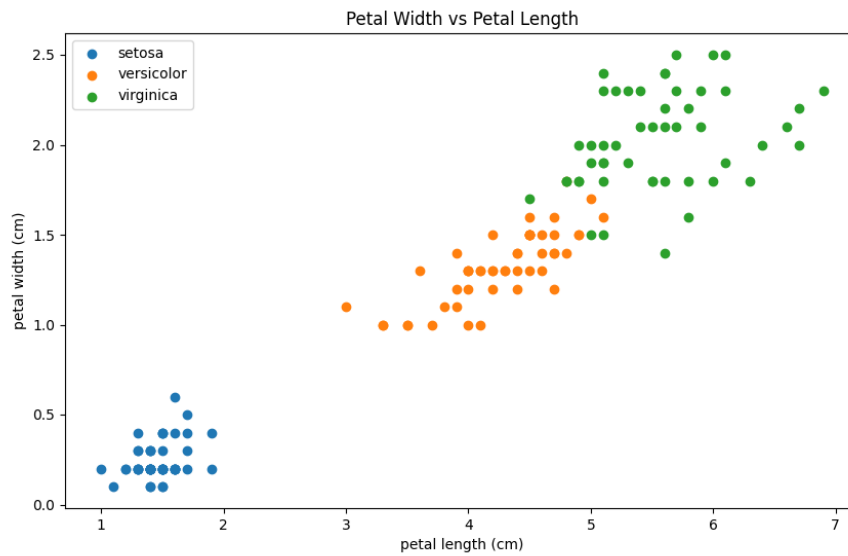
melakukan hull hingga ditemukan titik-titik paling ujung dari bagian atas/bawah. Setelah itu array of hull akan dikembalikan.

- `getAngle(A,B,C) -> float:`
fungsi `getAngle` akan membantu mendapatkan sudut ABC pada titik B dalam derajat. Fungsi ini digunakan untuk menentukan titik maks dari garis jika terdapat beberapa titik yang jaraknya sama.
- `getDeterminant(A,B,C) -> float:`
fungsi `getDeterminant` digunakan untuk mendapat determinan dari matriks $\begin{bmatrix} A_x & A_y & 1 \\ B_x & B_y & 1 \\ C_x & C_y & 1 \end{bmatrix}$, untuk menentukan apakah titik C berada di atas atau bawah garis AB.
- `getDistance(A,B,C) -> float:`
fungsi `getDistance` digunakan untuk mendapat jarak dari titik C ke garis AB secara tegak lurus. Fungsi ini digunakan pada `convexhull` untuk menentukan titik maks yang jaraknya terjauh.

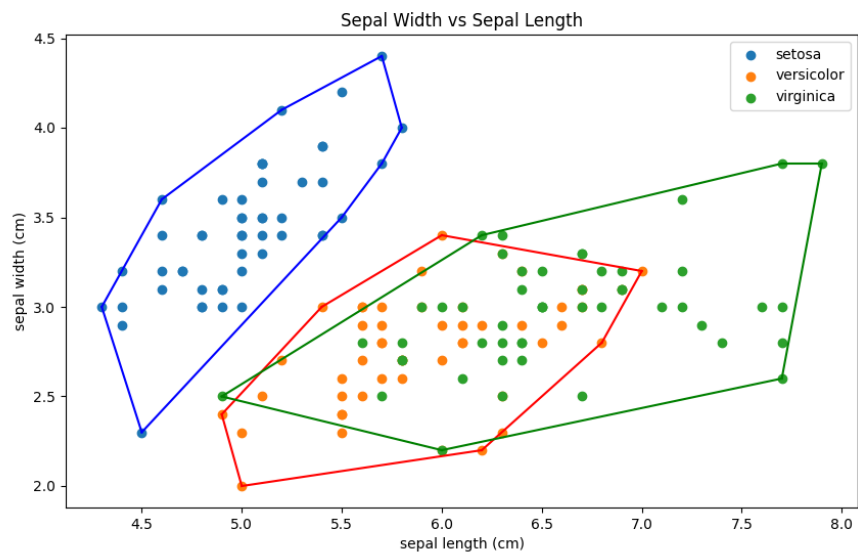
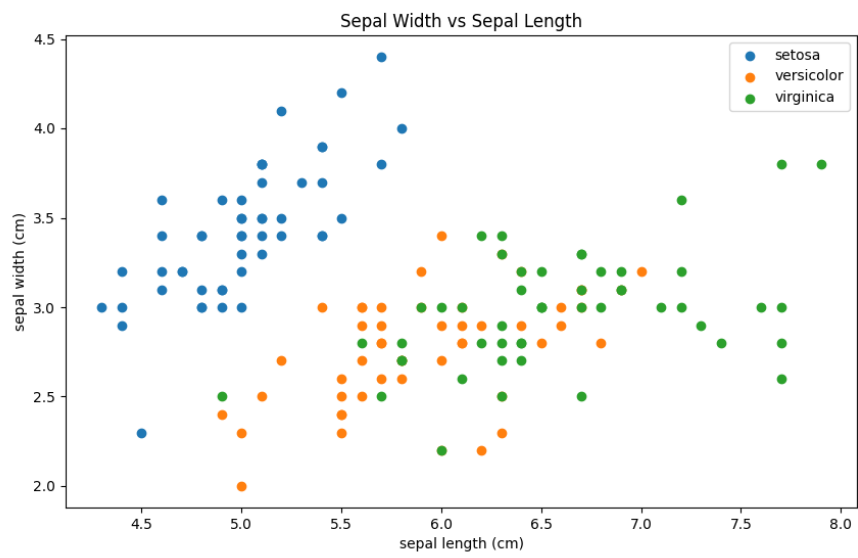
3. Hasil Uji

Berikut akan ditampilkan screenshot hasil uji pada tiap dataset, sebelum dan sesudah dilakukan convex-hull:

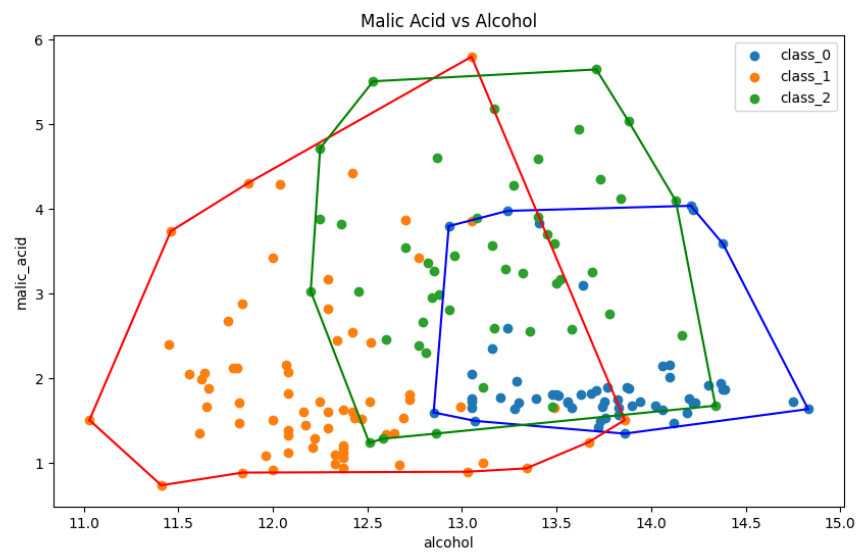
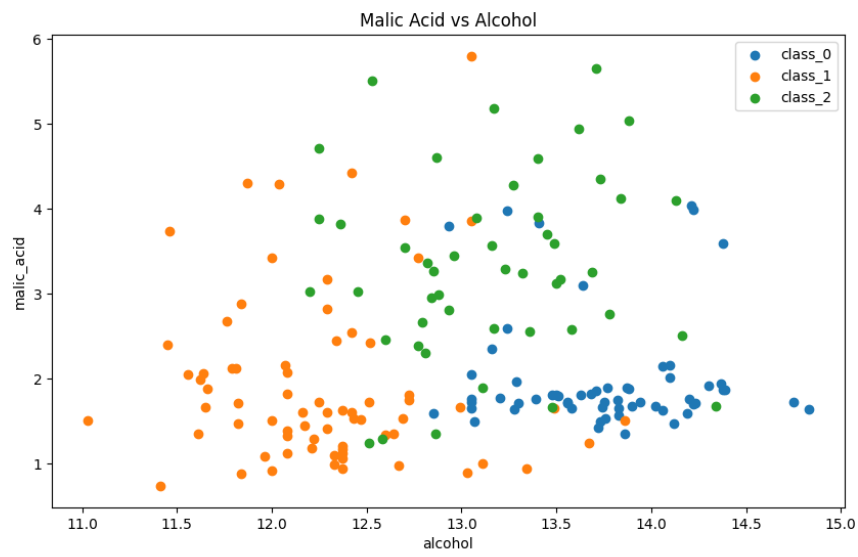
- Dataset Iris 1 (Petal Width vs Petal Length)



- Dataset Iris 2 (Sepal Width vs Sepal Length)



- Dataset Wine (Malic Acid vs Alcoholic)



- Dataset BreastCancer (Mean Texture vs Mean Radius)

