

Nama : Sulthan Esa Maulana

NIM : 3332190039

UTS Mata Kuliah Kecerdasan Buatan

ANALISIS LISTING PROGRAM

1. Analisa algoritma untuk `logistic_regression.py`. Dan analisa algoritmanya dan jalankan di komputer anda. (Untuk Chapter 2)
2. Analisa algoritma untuk `decision_trees.py`. Dan analisa algoritmanya dan jalankan di komputer anda. (Untuk Chapter 3)
3. Analisa algoritma untuk `mean_shift.py`. Dan analisa algoritmanya dan jalankan di komputer anda. (untuk Chapter 4)
4. Analisa algoritma untuk `nearest_neighbors_classifier.py`. Dan analisa algoritmanya dan jalankan di komputer anda (untuk Chapter 5)
5. Analisa algoritma untuk `states.py`. Dan analisa algoritmanya dan jalankan di komputer anda (untuk Chapter 6)

JAWAB:

1. Logistic Regression

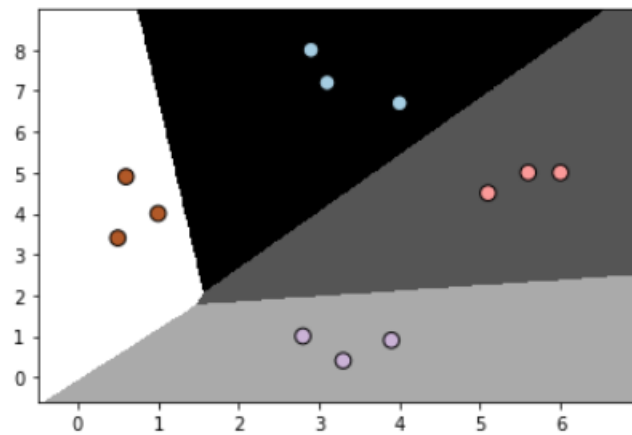
Data sampel yang di input atau digunakan adalah sebagai berikut

```
plt.show()
# Define sample input data
X = np.array([[3.1, 7.2], [4, 6.7], [2.9, 8], [5.1, 4.5], [6, 5], [5.6, 5], [3.3, 0.4], [3.9, 0.9], [2.8, 1], [0.5, 3.4], [1, 4], [0.6, 4.9]])
y = np.array([0, 0, 0, 1, 1, 1, 2, 2, 2, 3, 3, 3])
```

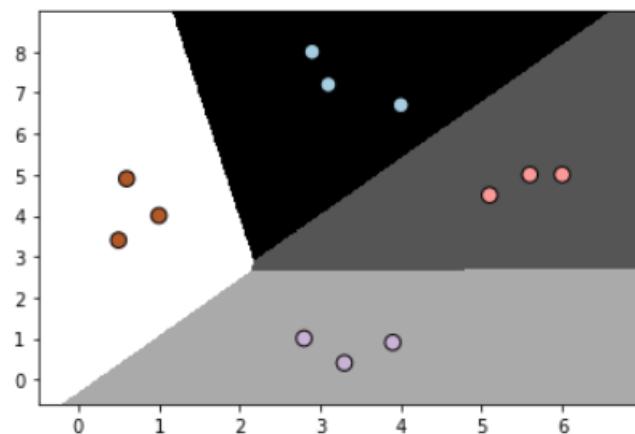
Dari data sampel tersebut dibuat *logistic regression classifier* menggunakan listing berikut

```
# Create the logistic regression classifier
classifier = linear_model.LogisticRegression(solver='liblinear', C=1)
```

Dan hasil dari *logistic regression classifier* yang memiliki nilai $C = 1$ adalah sebagai berikut

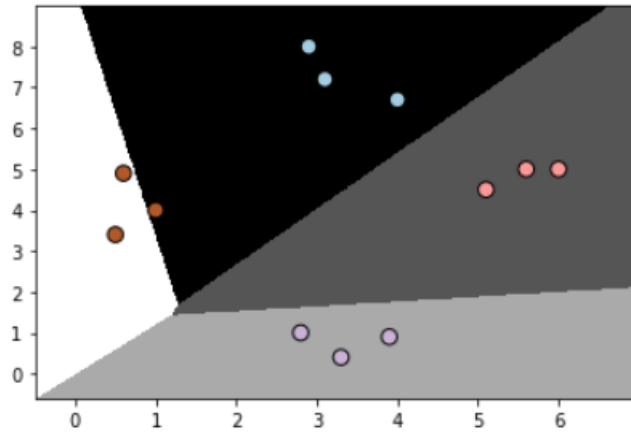


Untuk melihat perbedaannya kita ubah nilai C nya menjadi lebih besar, contohnya $C = 10$



Dapat dilihat perbedaan antara grafik yang menggunakan nilai $C = 1$ dan grafik yang menggunakan $C = 10$, grafik yang menggunakan nilai $C = 10$ mempunyai output klasifikasi atau area segmentasi yang lebih rapih dibandingkan dengan grafik $C = 1$. *Logistic regression* mempunyai fungsi untuk menampilkan 4 buah data yang kemudian algoritmanya akan mengklasifikasikan data tersebut menggunakan 4 area pada masing masing data

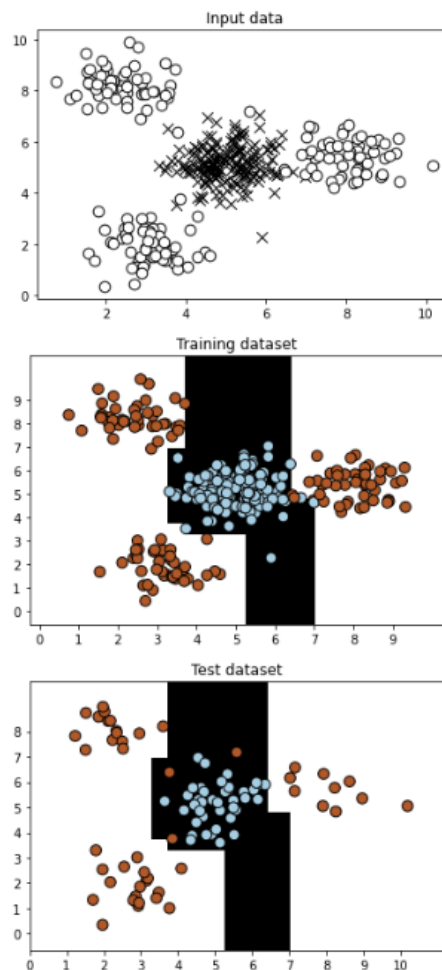
Jika menggunakan nilai $C = 0.5$ maka akan tercipta grafik seperti berikut ini



Dapat dilihat terdapat data yang berada di dua tempat sekaligus, hal ini terjadi ketika algoritma tidak dapat membedakan data yang mengakibatkan data tersebut muncul di dua tempat hal ini dinamakan sebagai *underfit*

2. Decision Trees

Setelah menginputkan *script decision trees* dari chapter 3 dan menjalankan *script* tersebut muncul 3 *output* grafik seperti berikut ini



```
#####
```

Classifier performance on training dataset				
	precision	recall	f1-score	support
Class-0	0.99	1.00	1.00	137
Class-1	1.00	0.99	1.00	133
accuracy			1.00	270
macro avg	1.00	1.00	1.00	270
weighted avg	1.00	1.00	1.00	270

```
#####
```

```
#####
```

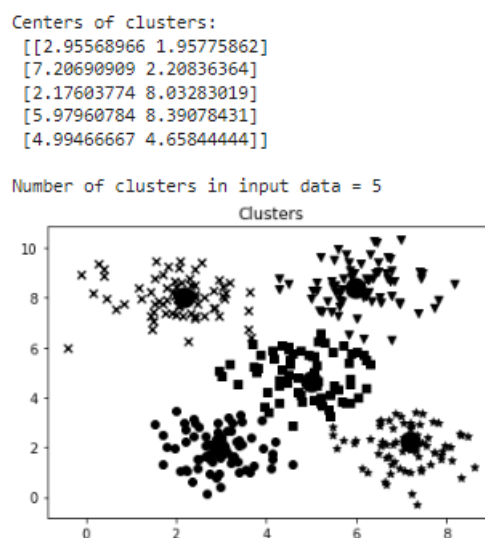
Classifier performance on test dataset				
	precision	recall	f1-score	support
Class-0	0.93	1.00	0.97	43
Class-1	1.00	0.94	0.97	47
accuracy			0.97	90
macro avg	0.97	0.97	0.97	90
weighted avg	0.97	0.97	0.97	90

```
#####
```

Didapatkan data – data berikut, yang dimana data tersebut mencantumkan *precision* atau ketepatan data menempati plot, *recall* atau banyaknya data yang terpanggil, *f1-score* atau nilai harmonis rata – rata, dan *support* atau banyaknya data yang dimasukan

3. Data Clustering

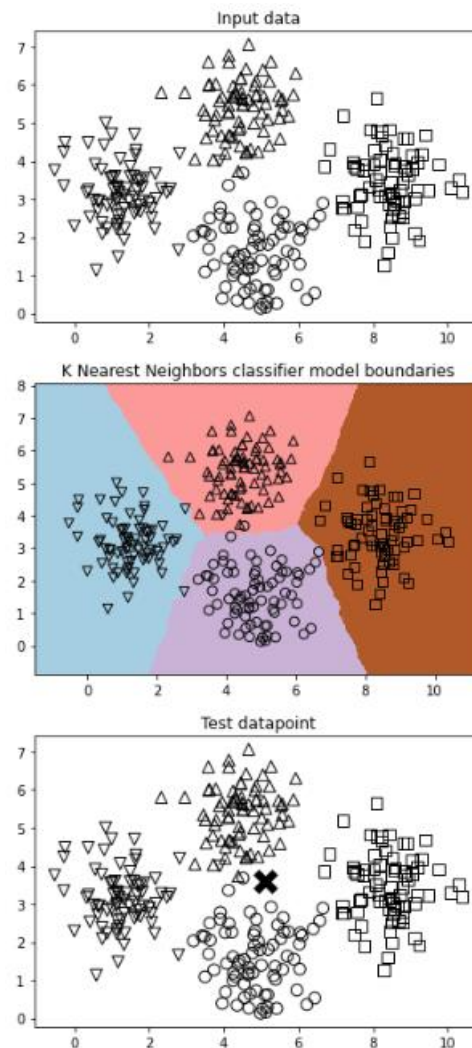
Setelah menginputkan *script meanshift* dari chapter 4 dan menjalankan *script* tersebut muncul *output* seperti berikut ini

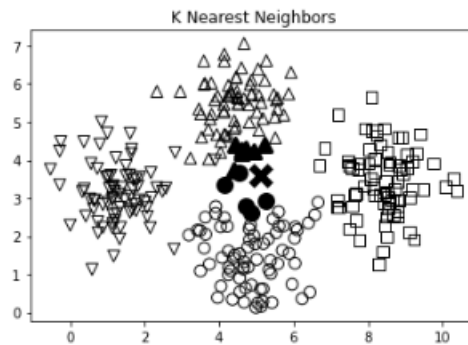


Data *clustering* merupakan script untuk mengelompokan data, dapat dilihat pada grafik diatas data yang dimaksud adalah bentuk – bentuk kotak, bintang, segitiga, silang dan lingkaran. Dan dapat dilihat juga dari tiap – tiap pengelompokan terdapat sebuah lingkaran di pusatnya, hal itu menandakan data tersebut sangat mirip

4. K-Nearest Neighbor

Setelah menginputkan *script nearest neighbors classifier* dari chapter 5 dan menjalankan *script* tersebut muncul *output* seperti berikut ini





K-nearest neighbors memiliki prinsip yang hampir sama dengan *clustering*, tetapi k-nearest neighbor mengelompokkan data berdasar dari kedekatan nilai tetangganya dari data yang diinput secara acak, dapat dilihat pada grafik diatas juga terdapat bentuk lingkaran di antara kelompok segitiga, hal ini menyatakan bahwa kedekatan nilai tetangga lingkaran tersebut masuk kedalam bentuk segitiga, tapi secara bentuk merupakan bentuk lingkaran

5. States

Setelah menginputkan *script states* dari chapter 6 dan menjalankan *script* tersebut muncul *output* seperti berikut ini

```
Is Nevada adjacent to Louisiana?:
No

List of states adjacent to Oregon:
Idaho
Washington
Nevada
California

List of coastal states adjacent to Mississippi:
Louisiana
Alabama

List of 7 states that border a coastal state:
New Hampshire
Mississippi
Georgia
Pennsylvania
Tennessee
New York
Oregon

List of states that are adjacent to Arkansas and Kentucky:
Missouri
Tennessee
```

Hasil tersebut menyatakan bahwa algoritma *states* termasuk ke dalam aturan *rule base*

NOTE:

- a. Dari kelima script diatas, *logistic regression*, *decision trees*, *k-nearest neighbor*, *data clustering (meanshift)* masuk kedalam *supervised learning*
- b. *States* masuk kedalam *Expert system*

LINK GITHUB:

<https://github.com/Sulthanesa/Kecerdasan-Buatan-AI->