Tugas Kecil 1

Eksplorasi Library Decision Tree Learning pada Jupyter Notebook

Oleh:

Akeyla Pradia Naufal – 13519178

Muhammad Alfandavi Aryo Utomo – 13519211

**Pembagian Dataset**

Kedua dataset ini dibagi menjadi 80% training set dan 20% test set. Untuk memastikan data tidak teracak lagi setiap kali di-*compile*, digunakan random state yang konstan.

Text

Description automatically generated

**Label Encoding**

Untuk berurusan dengan data kategorikal di dataset tenis, digunakan pustaka LabelEncoder.

Text, letter

Description automatically generated

**Algoritma Decision Tree Classifier**

**Algoritma Id3 Estimator**

**Algoritma K-Means**

**Algoritma Logistic Regression**

Berdasarkan dokumentasi di sklearn, banyak iterasi maksimal secara default dari algoritma ini adalah 100. Ketika algoritma ini dijalankan secara default, akan ada peringatan bahwa *solver* dari algoritma ini gagal konvergen. Sehingga, diberi tahu bahwa iterasi maksimal dari algoritma ini adalah 2000.

Secara default juga, algoritma yang digunakan untuk permasalahan optimisasi adalah lbfgs. Dikarenakan dataset yang ada berukuran kecil, akan dicoba algoritma liblinear, sesuai dengan yang disarankan di dokumentasi. Dan benar saja, terjadi sedikit peningkatan akurasi dan skor F1.

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

**Algoritma Neural Network**

Dengan alasan yang sama seperti pada algoritma sebelumnya, diperlukan pengaturan atribut max\_iter menjadi sebesar 200000 untuk membuat algoritma dapat konvergen. Selain itu, dikarenakan dataset berukuran kecil, digunakan solver lbfgs yang lebih cocok untuk dataset berukuran kecil alih-alih menggunakan solver default yaitu adam. Terjadi peningkatan

Graphical user interface, text, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Lebih lanjut, juga terdapat beberapa activation function yang dapat dipakai. Secara default, yang digunakan adalah RELU. Setelah dicoba, yang menghasilkan akurasi terbaik adalah activation function logistic. Akan tetapi, kenaikannya tidak terlalu signifikan dan diperberat dengan fakta bahwa fungsi berjalan cukup lambat.

Graphical user interface, text, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text

Description automatically generated

Graphical user interface, text

Description automatically generated

**Algoritma Support Vector Machine (SVM)**

Di algoritma ini, parameter yang di-*tuning* adalah kernel yang dipakai. Secara default, kernel yang dipakai adalah rbf. Dari beberapa pilihan kernel yang ada, yang memberikan akurasi paling tinggi adalah linear.

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated