Tugas Model Klasifikasi dengan KNN

Tugas ini dikerjakan menggunakan Python dan beberapa library yang ada pada Python.

Tahapan yang dilakukan:

1. Mengimport library yang diperlukan

Text

Description automatically generated

=> library yang digunakan ini sangat membantu dalam pengerjaan karena kita tidak perlu membuat fungsi-fungsi baru yang digunakan dalam pengerjaan tugas ini, dan tinggal memanggil fungsi tersebut. Library yang digunakan dalam tugas ini, antara lain: pandas, numpy, sklearn, dan matplotlib

1. Mengimport data yang akan digunakan dan menampilkan 5 data teratas

A screenshot of a computer

Description automatically generated with medium confidence

=> Kita mengimport data dan meng-assign-nya ke dalam variable df, sehingga memudahkan kita untuk memanggil data yang ada. Menampilkan 5 data teratas merupakan hal yang banyak dilakukan untuk mengecek apakah data yang diambil sudah sesuai ekspektasi, dan memberikan gambaran singkat terkait data yang digunakan.

1. Membagi data menjadi 2 bagian, untuk cross validasi sebesar 80% : 20%

Text

Description automatically generated

=> Melakukan cross validasi dengan membagi data yang dipunyai menjadi 80% untuk training dan 20% untuk testing sangatlah berguna untuk memberi gambaran apakah klasifikasi yang kita buat sudah baik atau belum dan menghindari adanya underfitting dan overfitting.

1. Menampilkan confusion matrix dan tabel akurasi

Text

Description automatically generated

=> Confusion matrix dan table akurasi sangat berguna bagi kita dalam mengevaluasi kinerja dari klasifikasi yang kita buat. Dalam kasus ini, diperoleh hasil yang sangat baik, terlihat pada table akurasi, kita mendapatkan presisi dan akurasi dari kinerja klasifikasi yang kita buat sebesar 1.00, hal ini sangatlah bagus, hal ini mungkin dikarenakan data yang kita gunakan sudah sangat clean dan mudah untuk diklasifikasikan.

1. Menghitung error untuk nilai K dari 1 sampai 40

Graphical user interface

Description automatically generated

=> Menghitung error untuk setiap nilai K sangat baik untuk menentukan berapa nilai K yang lebih cocok untuk digunakan. Terlihat pada hasil visualisasi, bahwa nilai K dari 1 sampai 34 memberikan error rata-rata sebesar 0.00, hal ini tentu sangatlah bagus dalam kinerja klasifikasi kita untuk prediksi data nantinya

1. Melakukan prediksi

Text

Description automatically generated

=> Terakhir, dilakukan prediksi data dengan data yang diambil dari data yang pertama (dapat dilihat pada tahap ke-2), dengan sedikit perubahan pada body mass-nya, dari 3750 diubah menjadi 3500 (tidak terlalu jauh nilainya, sehingga kita memiliki ekspektasi hasil yang akan keluar apa). Dan hasil yang dikeluarkan sesuai ekspektasi kita yaitu spesies Adelie, sesuai dengan data pertama.