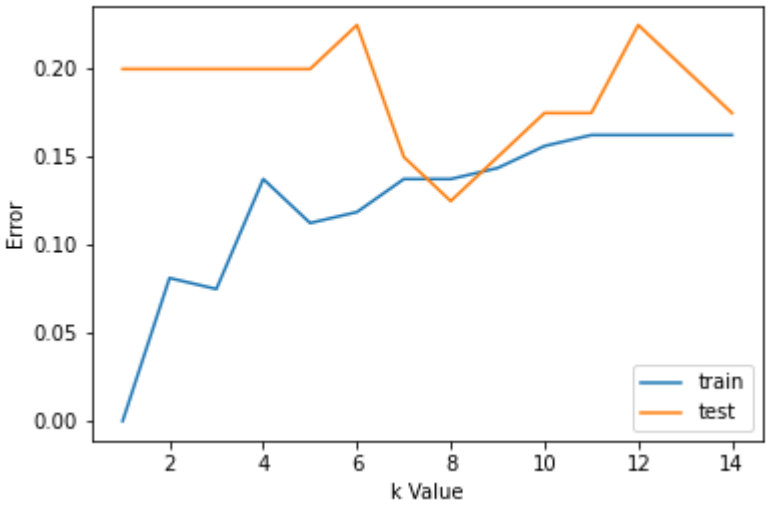
Nama : Ferza Reyaldi
NIM : 09021281924060
Mata Kuliah : Data Mining

Tugas Pertemuan 2 (Pertemuan 12)
Pelajari & tuliskan penjelasan tentang metode atau algoritma K-NN!

Jawab:
Algoritma K-Nearest-Neighbour (KNN) memperkirakan seberapa besar kemungkinan titik data menjadi anggota dari satu kelompok atau lainnya. Ini pada dasarnya melihat titik data di sekitar satu titik data untuk menentukan grup apa yang sebenarnya berada. Misalnya, jika satu titik ada di kisi dan algoritma mencoba menentukan grup mana titik data itu berada (Grup A atau Grup B, misalnya) ia akan melihat titik data di dekatnya untuk melihat kelompok mana yang menjadi mayoritas titik tersebut.

Pembelajaran algoritma KNN:
- *Instance-based learning*: Di sini kita tidak mempelajari bobot dari data pelatihan untuk memprediksi keluaran (seperti dalam algoritma berbasis model) tetapi menggunakan seluruh instance pelatihan untuk memprediksi keluaran untuk data yang tidak terlihat.
- *Lazy Learning*: Model tidak dipelajari menggunakan data pelatihan sebelumnya dan proses pembelajaran ditunda ke waktu ketika prediksi diminta pada instance baru (**tidak cocok untuk stream data)**.
- *Non-Parametric*: Dalam KNN, tidak ada bentuk fungsi pemetaan yang ditentukan sebelumnya.

Cara memilih nilai K:
1) Menggunakan kurva kesalahan: Gambar di bawah menunjukkan kurva kesalahan untuk nilai K yang berbeda untuk data training dan testing.



Pada nilai K rendah, terjadi overfitting data/varians tinggi. Oleh karena itu kesalahan testing tinggi dan kesalahan training rendah. Pada K=1 dalam data training, kesalahan selalu nol, karena tetangga terdekat ke titik itu adalah titik itu sendiri. Oleh

karena itu, meskipun kesalahan training rendah, kesalahan testing tinggi pada nilai K yang lebih rendah. Ini disebut overfitting.

Saat kita meningkatkan nilai K, kesalahan testing berkurang. Tetapi setelah nilai K tertentu, terjadi bias/underfitting dan training error menjadi tinggi. Jadi dapat disimpulkan bahwa awalnya kesalahan data testing tinggi (karena varians) kemudian menjadi rendah dan stabil dan dengan peningkatan lebih lanjut dalam nilai K, dan kemudian meningkat lagi (karena bias). Nilai K ketika kesalahan testing stabil dan rendah dianggap sebagai nilai optimal untuk K. Dari kurva kesalahan di atas dapat dipilih K=8 untuk implementasi algoritma KNN.

2) *Domain knowledge* sangat berguna dalam memilih nilai K.

3) Nilai K harus ganjil saat mempertimbangkan klasifikasi biner (dua kelas).

Pros:

- K-NN cukup intuitif dan sederhana.
- K-NN tidak memiliki asumsi yang harus dipenuhi.
- Tidak Ada Langkah Pelatihan.
- Terus berkembang.
- Sangat mudah diimplementasikan untuk masalah multi-kelas.
- Dapat digunakan baik untuk Klasifikasi maupun Regresi.
- One Hyper Parameter, KNN mungkin memerlukan beberapa waktu saat memilih parameter hyper pertama tetapi setelah itu parameter lainnya disejajarkan dengannya.
- Fleksibilitas dalam kriteria jarak yang dapat dipilih.

Cons:

- Algoritma yang lambat, K-NN sangat mudah diimplementasikan tetapi seiring dengan pertumbuhan dataset, efisiensi atau kecepatan algoritma menurun dengan sangat cepat.
- Curse of Dimensionality, KNN bekerja dengan baik dengan sejumlah kecil variabel input tetapi seiring dengan bertambahnya jumlah variabel, algoritma K-NN berjuang untuk memprediksi output dari titik data baru.
- K-NN membutuhkan fitur yang homogen.
- Jumlah tetangga yang optimal, salah satu masalah terbesar dengan K-NN adalah memilih jumlah tetangga yang optimal untuk dipertimbangkan saat mengklasifikasikan entri data baru.
- Masalah timbul ketika data tidak seimbang.
- Sensitivitas outlier yang sangat tinggi.
- K-NN secara inheren tidak memiliki kemampuan untuk menangani masalah missing values.