LAPORAN TUGAS BESAR II
IF3170 Inteligensi Buatan

Implementasi Algoritma

Disusun oleh:

Kevin John Wesley Hutabarat	13521042
Muhammad Equillibrie Fajria	13521047
M Farrel Danendra Rachim	13521048
Jericho Russel Sebastian	13521107

PROGRAM STUDI TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
JALAN GANESHA 10

NOVEMBER 2023

DAFTAR ISI

DAFTAR GAMBAR

BAB I
PENDAHULUAN

1.1. Deskripsi Tugas

Setelah kalian melakukan EDA dan pemrosesan, pada tugas kecil 2, kalian akan diminta untuk mengimplementasikan algoritma pembelajaran mesin yang telah kalian pelajari di kuliah, yaitu KNN dan Naive-Bayes. Data yang digunakan sama seperti tugas kecil 2. Latihlah model dengan menggunakan data latih, kemudian validasi hasil dengan menggunakan data validasi untuk mendapatkan insight seberapa baik model melakukan generalisasi.

BAB II
IMPLEMENTASI

2.1. Implementasi Algoritma K-Nearest Neighbor

Algoritma k-Nearest Neighbor (kNN) adalah salah satu jenis algoritma *supervised learning* yang digunakan dalam melatih suatu dataset untuk dilakukan klasifikasi. KNN bersifat non-parametrik, yaitu tidak membuat asumsi atau hipotesis terhadap data. Selain itu, KNN juga bersifat *lazy learner*, karena algoritma ini tidak mempelajari data *training* secara langsung, dan hanya melakukan tindakan terhadap data tersebut saat proses klasifikasi dimulai.

Pertama-tama, akan dicari terlebih dahulu banyak nilai "k" yang akan digunakan, yakni banyaknya "neighbor" (k *instance* paling mirip). Nilai "k" dibebaskan, namun akan dijelaskan cara mencari nilai "k" yang ideal untuk dataset ini di subbab 2.3. Untuk implementasi manual, akan dimisalkan k = 15.

Kemudian, untuk setiap baris data *validation*, akan dihitung jarak antara item-item data di antara kedua dataset: *train* dan *validation*. Untuk kasus ini, kami menggunakan Euclidean *distance*. Nilai-nilai jarak dan data yang berhubungan tersebut akan dimasukkan ke dalam *list* "distances". List *distance* diurutkan secara menaik berdasarkan nilai jaraknya.

Setelah itu, dari *list* "distances", akan dipilih k *item list* pertama (k *item* dengan jarak terkecil) untuk dimasukkan ke dalam *list* "neighbors". *List* ini akan dijadikan *neighbor* untuk setiap baris di data tes (*validation*). Dari tetangga yang dipilih, akan dicari kelas mayoritas dari kolom target, yaitu "price_range", apakah termasuk kelas 0, 1, 2, atau 3. Akhirnya, hasil prediksi akan dimasukkan ke dalam list "predictions". Program akan kembali menghitung jarak Euclidean untuk baris data *validation* berikutnya.

Dari algoritma ini, akan dihasilkan prediksi "price_range" untuk setiap baris data *validation*.

2.2. Implementasi Algoritma Naive Bayes

Algoritma Naive Bayes merupakan salah satu jenis algoritma *supervised learning* yang digunakan untuk melakukan klasifikasi berdasarkan teorema bayes dengan asumsi kolom dataset saling independen. Naive Bayes bersifat *probabilistic classifier*. Hal ini dikarenakan Naive Bayes menggunakan probabilitas (Teorema Bayes) untuk melakukan klasifikasi. Berbeda dengan K-Nearest Neighbor, Naive Bayes memiliki hipotesis.

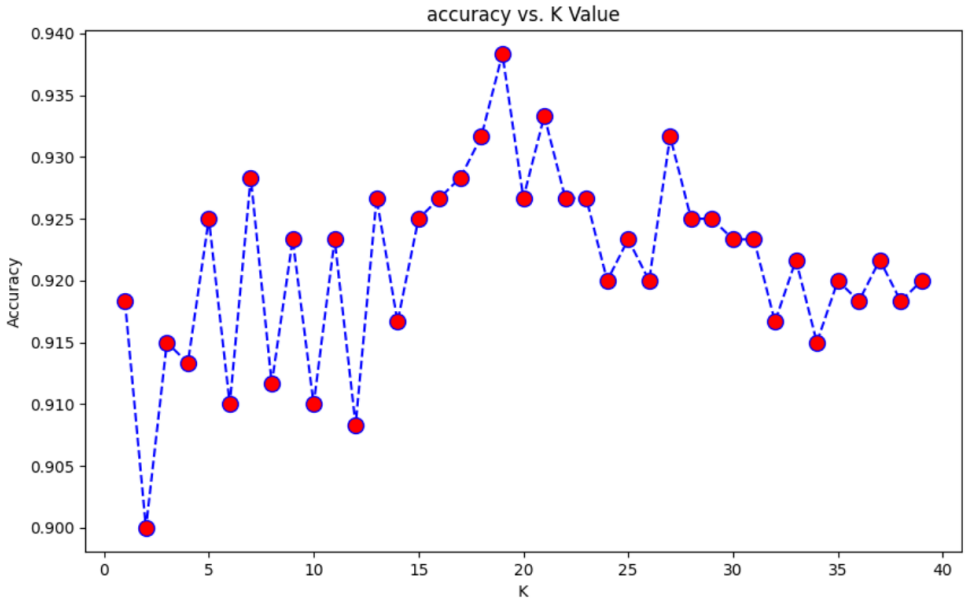
Untuk implementasikan algoritma Naive-Bayes, pertama-tama data akan dipisahkan berdasarkan kelasnya sesuai dengan nilai yang dimiliki pada kolom target. Nilai pada kolom target bervariasi antara 0, 1, 2, atau 3. Kemudian, dihitung nilai mean dan standar deviasi dari data secara keseluruhan. Setelah itu, akan dihitung statistik dasar dari masing-masing kelompok data tersebut, yaitu mean dan standar deviasi nya. Nilai dari peluang pada masing-masing kolom akan dihitung. Setelah itu, hasil dari nilai peluang pada setiap kolom dikalikan semuanya untuk setiap kelas. Kelas dengan nilai hasil kali peluang terbesar kemudian akan dipilih sebagai prediksi kelas dari data tersebut.

2.3. Perbandingan Hasil Implementasi dengan Hasil dari Pustaka Scikit-learn

Sebenarnya, klasifikasi data baik dengan KNN maupun dengan Naive-Bayes dapat dilakukan dengan pustaka Scikit-learn. Scikit-learn memiliki berbagai fitur yang dapat langsung dimanfaatkan untuk meningkatkan akurasi klasifikasi.

2.3.1. Perbandingan Hasil KNN

Dengan pustaka scikit-learn, kami mampu menemukan nilai k yang optimal untuk algoritma KNN yang akan dijalankan. Kami menggunakan KNeighborsClassifier dan fungsi predict(), fit(), dan accuracy_score() untuk mencari nilai akurasi dari data yang diuji. Setelah itu, dengan menggunakan matplotlib, akan digambarkan *dashed plot* terhadap nilai akurasi untuk setiap nilai k dari 1 sampai 40. Dari analisis plot, diperoleh konklusi bahwa nilai k yang menghasilkan nilai akurasi optimal adalah 19, dengan nilai akurasi maksimal 0.9383334.



Gambar 2.3.1.1 Perbandingan akurasi dengan nilai k

Berdasarkan grafik di atas, dibandingkan dengan nilai akurasi implementasi KNN manual (dengan k=15), yakni 0.925, pustaka scikit-learn mampu menghasilkan nilai akurasi yang lebih tinggi.

2.3.2. Perbandingan Hasil Naive-Bayes

Pustaka Scikit Learn menyediakan fungsi yang dapat mengklasifikasi suatu data dengan langsung menggunakan model Naive-Bayes. Disini kami menggunakan GaussianNB dan fungsi fit(), predict(), dan dan accuracy_score() untuk mencari nilai akurasi dari data yang diuji. Dengan metode ini, kami memperoleh nilai akurasi sebesar 0.7866. Nilai ini sedikit lebih besar daripada implementasi *from scratch*, yaitu 0.7833, yang artinya pustaka scikit-learn mampu menghasilkan nilai akurasi yang lebih tinggi.

2.4. Pemrosesan Kaggle

Untuk mengevaluasi lebih lanjut performa model KNN dan Naive-Bayes, kami menguji sebuah dataset bernama “test.csv” yang mengandung 2000 baris data, serta memiliki seluruh atribut data latih dan data validasi, kecuali price_range. Tujuan pengujian ini adalah menentukan kategori untuk setiap instansi data yang paling tepat, mengunggah hasil pengujian berupa CSV ke Kaggle, dan melihat berapa skor berdasarkan *accuracy* pada *multi-class*.

BAB III
SIMPULAN

Berdasarkan hasil implementasi dan pengujian algoritma k-Nearest Neighbor dan Naive Bayes, didapat bahwa k-Nearest Neighbor dan Naive Bayes sama-sama bisa memberikan hasil prediksi yang cukup akurat dengan nilai akurasi maksimal k-Nearest Neighbor sebesar 0.9383334 dan Naive Bayes sebesar 0.7866. Dapat dilihat juga bahwa hasil klasifikasi dari algoritma k-Nearest Neighbor memiliki akurasi yang lebih tinggi dibandingkan dengan algoritma Naive Bayes. Hal ini dikarenakan ada beberapa data yang tidak saling independen (seperti three_g dan four_g) serta sifat data yang tidak sederhana (*simplistic*).

REFERENSI

- https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning
- https://machinelearningmastery.com/naive-bayes-classifier-scratch-python/

LAMPIRAN

Pembagian Tugas

NIM	Nama	Tugas
13521042	Kevin John Wesley Hutabarat	Naive-Bayes
13521047	Muhammad Equilibrie Fajria	kNN
13521048	M Farrel Danendra Rachim	kNN
13521107	Jericho Russel Sebastian	Naive-Bayes

Link Repository GitHub:
https://github.com/Breezy-DR/Tubes2_dont-mine-at-night.git