ISSN: 2339-2541
JURNAL GAUSSIAN, Volume 15, Nomor 1, Tahun 2026, Halaman 89 - 97
Online di: https://ejournal3.undip.ac.id/index.php/gaussian/

PERBANDINGAN PERFORMA *DISTANCE MEASURES* PADA ALGORITMA *NEAREST CENTROID NEIGHBOR* DAN *K-NEAREST NEIGHBOR* DALAM PROSES KLASIFIKASI BINTANG

Nurul Azizah[1], Dewi Sri Susanti[2*], Selvi Annisa[3]
[1,2,3] Program Studi Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lambung Mangkurat
***e-mail**: ds_susanti@ulm.ac.id

DOI: 10.14710/j.gauss.15.1.89-97

Article Info:
Received: 2025-11-04
Accepted: 2026-05-16
Available Online: 2025-05-20

Keywords:
Stellar Classification; Distance Measures; Nearest Centroid Neighbor; k-Nearest Neighbor

Abstract: Stars are celestial bodies that can be classified based on several characteristics, including temperature, luminosity, radius, magnitude, stellar color, and spectral class. Stars are generally grouped into six categories: brown dwarfs, red dwarfs, white dwarfs, main sequence stars, supergiants, and hypergiants. Stellar classification is important to astronomers because it can help identify new types of stars and improve our understanding of their composition, temperature, and evolutionary stages. This classification process can be done using the Nearest Centroid Neighbor (NCN) and k-Nearest Neighbor (k-NN) algorithms, by applying various distance measures such as Euclidean, Manhattan, Minkowski, Chebyshev, Cosine, Jaccard, and Hamming. This study aims to compare the performance between NCN and k-NN using these seven distance measures. The results show that Euclidean, Manhattan, and Minkowski distances produce a perfect performance of 100% in both algorithms. Chebyshev distance yielded perfect performance in k-NN but slightly lower in NCN with a performance of 92%. Thus, the k-NN algorithm provides superior performance compared to the NCN algorithm in the stellar classification process.

1. PENDAHULUAN

Bintang merupakan salah satu benda langit yang dapat dilihat dari bumi saat malam hari. Bintang dapat dibagi menjadi beberapa kategori, seperti *brown dwarf, red dwarf, white dwarf, main sequence, supergiants,* dan *hypergiants*. Klasifikasi bintang didasarkan pada temperatur, perbandingan tingkat kecerahan bintang dibandingkan tingkat kecerahan Matahari, perbandingan ukuran bintang dengan Matahari, magnitudo atau kecerahan bintang, warna bintang, serta kategori *spectral* dari bintang tersebut. Klasifikasi bintang atau *stellar classification* penting bagi astronom untuk menyediakan sistem apabila ditemukan bintang jenis baru. Proses klasifikasi dengan cara manual merupakan cara yang tidak mudah, sehingga dengan berkembangnya berbagai metode, penerapan *machine learning* dapat menjadi solusi untuk memudahkan proses pengklasifikasian.

Algoritma yang dapat diterapkan untuk mengklasifikasikan bintang adalah *k-Nearest Neighbor* (k-NN), sebagaimana penelitian oleh Nayak, Bhat, Reddy dan Rao (2022) yang membandingkan *distance measures* pada algoritma k-NN untuk pengklasifikasian bintang. Penelitian ini mengklasifikasikan bintang dengan tujuh variasi *distance measures* yang diterapkan dalam algoritma k-NN. Pemilihan *distance measures* menjadi komponen penting dalam algoritma klasifikasi yang bertujuan untuk mencocokkan data *training* dengan data *testing* (Wahyono, Trisna, Sariwening, Fajar dan Wijayanto 2020). *Distance measures* yang diterapkan dalam pengklasifikasian bintang oleh Nayak dkk. (2022) meliputi jarak Euclidean, Manhattan, Minkowski, Chebyshev, Cosine, Jaccard, dan Hamming. Hasil

klasifikasi bintang dengan algoritma k-NN menunjukkan akurasi tertinggi mencapai 84.72% ketika menggunakan jarak Cosine.

Selain algoritma k-NN yang dikemukakan oleh Fix dan Hodges (1951), metode klasifikasi berbasis jarak dan konsep ketetanggan adalah *Nearest Centroid Neighbor* (NCN) yang diperkenalkan oleh Chaudhuri (1996). Algoritma NCN ini mempertimbangkan jarak dan tata letak data dalam konsep ketetanggaannya (Chaudhuri 1996). Berdasarkan pemaran sebelumnya, dilakukan perbandingan performa algoritma NCN dan k-NN dengan berbagai *distance measures* dalam proses klasifikasi bintang.

2. TINJAUAN PUSTAKA

Bintang merupakan benda langit yang mampu memancarkan cahaya sendiri. Bintang dilahirkan di dalam awan debu dan gas. Gumpalan gas terbentuk dan menarik lebih banyak massa seiring waktu. Gumpalan-gumpalan tersebut akan mulai berputar dan memanas hingga menjadi cukup berat dan panas, maka fusi nuklir dimulai di inti mereka. Proses ini terjadi ketika proton bergabung untuk membentuk inti helium, sehingga melepaskan banyak energi yang memanaskan bintang dan melawan gaya gravitasi. Berdasarkan proses-proses tersebut sebuah bintang terbentuk. Bintang-bintang tersebut diklasifikasikan menjadi beberapa tipe bintang, yaitu *brown dwarf, red dwarf, white dwarf, main sequence, supergiants,* dan *hypergiants* (NASA Universe Web Team, 2024).

Bintang-bintang tersebut dapat diklasifikasikan menggunakan algoritma NCN dan k-NN dengan berbagai *distance measures*. Sebelum tahap klasifikasi, data dibersihkan dari *missing value* dengan mengisi nilai kosong tersebut menggunakan median untuk variabel bertipe numerik dan modus untuk variabel bertipe kategorik (Saputra & Kristiyanti 2022). Selain itu, juga dilakukan proses transformasi data dengan teknik normalisasi dan *one-hot encoding*. Proses normalisasi bertujuan untuk menyeragamkan data dengan skala yang berbeda. Proses normalisasi dilakukan menggunakan normalisasi *z-score* menggunakan Persamaan berikut.

$$z_i = \frac{x - \mu}{\sigma} \tag{1}$$

dengan z_i merupakan nilai *z-score*, x merupakan nilai dari atribut X, μ merupakan nilai rata-rata atribut X, dan σ merupakan standar deviasi atribut X (Saputra dan Kristiyanti 2022). Selain itu, *one-hot encoding* digunakan untuk proses data transformasi menjadi data numerik. Pada proses *one-hot encoding* ini, semua elemen diatur menjadi nol (0), kecuali elemen yang sesuai dengan kategori akan diatur menjadi satu (1). *One-hot encoding* banyak digunakan dalam *machine learning* seperti proses klasifikasi (Samuels 2024).

Selanjutnya, data diklasifikasikan menggunakan algoritma k-NN dan NCN. Algoritma k-NN merupakan metode klasifikasi berbasis jarak, di mana kategori untuk data *testing* diberikan berdasarkan mayoritas kategori pada tertangga terdekat atau nilai k. Berdasarkan Cholil dkk. (2021) tahapan untuk metode k-NN adalah:

1. Menentukan nilai k yang disarankan menggunakan nilai ganjil.
2. Menghitung jarak antara data *testing* terhadap data *training* berdasarkan masing-masing *distance measures* yang akan disajikan melalui persamaan (2), (3), (4), (5), (6), (7), dan (8) pada bagian selanjutnya.
3. Mengurutkan hasil pada langkah 2 dari nilai terendah hingga tertinggi.
4. Mengumpulkan data dari langkah 3 berdasarkan k yang telah ditentukan.
5. Memberikan label pada data testing berdasarkan kategori yang paling banyak muncul pada langkah 4.

Metode klasifikasi lainnya yaitu metode NCN yang didasarkan pada ketetanggaan berbasis *centroid*. Metode ini menetapkan label kategori pada data *training*, di mana *centroid* dari setiap kategori memiliki jarak terdekat dengan data *testing*. Kemudian *centroid* pada kategori yang memiliki jarak terdekat dengan data *testing* akan dipilih sebagai label kategori untuk data tersebut (Tamatjita dan Mahastama 2016). Nilai *centroid* diperoleh dari rata-rata variabel X data *training* bertipe numerik, sedangkan *centroid* untuk variabel X data *training* bertipe kategorik menggunakan modus (Reihanah, I Maruddani dan Widiharih 2024). Tahapan klasifikasi pada metode NCN sebagai berikut:

1. Menghitung *centroid* (C) menggunakan rata-rata untuk variabel numerik dan menggunakan modus untuk variabel kategorik.
2. Menghitung jarak antara data *testing* terhadap *centroid* berdasarkan masing-masing *distance measures* pada Persamaan (2), (3), (4), (5), (6), (7), dan (8).
3. Mengurutkan hasil pada langkah 2 dari nilai terendah hingga tertinggi.
4. Memberikan label kategori berdasarkan kategori yang memiliki nilai jarak terkecil (Chaudhuri, 1996).

Kedua algoritma tersebut akan diterapkan pada beberapa *distance measures*. *Distance measures* umumnya disimbolkan dengan $d(i,j)$ yang menunjukkan jarak antara objek i dan objek j. Di mana $i = (x_{i1}, x_{i2}, \dots, x_{ip})$ dan $j = (x_{j1}, x_{j2}, \dots, x_{jp})$ merupakan dua objek yang dijelaskan oleh p atribut.

a. Jarak Euclidean

Jarak Euclidean merupakan *distance measures* yang paling populer untuk mengukur kemiripan atribut numerik yang dihitung dengan Persamaan

$$d(i,j) = \sqrt{\left(x_{i1} - x_{j1}\right)^2 + \left(x_{i2} - x_{j2}\right)^2 + \cdots + \left(x_{ip} - x_{jp}\right)^2} \tag{2}$$

Jarak Euclidean merupakan yang paling umum dan mudah diterapkan dengan menghitung akar kuadrat dari jumlah kuadrat selisih antara dua titik data.

b. Jarak Manhattan

Jarak Manhattan atau juga dikenal dengan *City Block Distance* merupakan *distance measures* untuk atribut numerik yang dihitung dengan Persamaan

$$d(i,j) = \left|x_{i1} - x_{j1}\right| + \left|x_{i2} - x_{j2}\right| + \cdots + \left|x_{ip} - x_{jp}\right| \tag{3}$$

Jarak Manhattan menghitung akumulasi selisih absolut, sehingga memiliki sifat yang lebih *robust* terhadap *outlier*.

c. Jarak Minkowski

Jarak Minkowski merupakan generalisasi dari jarak Euclidean dan jarak Manhattan yang juga digunakan untuk atribut numerik yang dihitung dengan Persamaan

$$d(i,j) = \sqrt[h]{\left|x_{i1} - x_{j1}\right|^h + \left|x_{i2} - x_{j2}\right|^h + \cdots + \left|x_{ip} - x_{jp}\right|^h} \tag{4}$$

di mana h merupakan bilangan *real* dengan syarat $h \geq 1$. Jarak Minkowski memberikan fleksibilitas yang lebih besar untuk memperoleh jarak yang paling sesuai, khususnya pada penelitian pemodelan spasial.

d. Jarak Chebyshev

Jarak Chebyshev merupakan jarak Minkowski ketika $h \to \infty$. Jarak Chebyshev dapat dihitung menggunakan Persamaan

$$d(i,j) = \max_f \left|x_{if} - x_{jf}\right| \tag{5}$$

Jarak Chebyshev mengekstrak selisih maksimum di antara seluruh dimensi atribut, sehingga klasifikasi didasarkan pada variabel yang memiliki perbedaan paling dominan.

e. Jarak Cosine

$$d(i,j) = 1 - \frac{x_{i1}x_{j1} + x_{i2}x_{j2} + \cdots + x_{ip}x_{jp}}{\sqrt{x_{i1}^2 + x_{i2}^2 + \cdots + x_{ip}^2}\sqrt{x_{j1}^2 + x_{j2}^2 + \cdots + x_{jp}^2}} \quad (6)$$

di mana nilai f untuk objek ke-i dan x_{jf} merupakan nilai f untuk objek ke-j (Han, Kamber dan Pei 2011). Jarak Cosine berfokus pada orientasi sudut antar vektor dan invarian terhadap skala, sehingga tetap efektif meskipun terdapat perbedaan intensitas pada atribut.

f. Jarak Jaccard

$$d(i,j) = 1 - \frac{|i \cap j|}{|i \cup j|} \quad (7)$$

di mana $|i \cap j|$ merupakan jumlah elemen yang sama dalam himpunan i dan j, serta $|i \cup j|$merupakan jumlah elemen gabungan dalam himpunan i dan j (Utomo, Much Ibnu Subroto dan Riansyah 2022). Jarak Jaccard mengukur rasio antara irisan dan gabungan yang efektif untuk menangani atribut kategorik pada data.

g. Jarak Hamming

$$d(i,j) = d(x_{i1}, x_{j1}) + d(x_{i2}, x_{j2}) + \cdots + d(x_{ip}, x_{jp}) \quad (8)$$

di mana jika $x_{if} = x_{jf}$ maka $d(x_{if}, x_{jf}) = 0$ dan jika $x_{if} \neq x_{jf}$ maka $d(x_{if}, x_{jf}) = 1$ (Wulandari, Oktaviani, Lestari dan Raya 2022). Jarak Hamming merupakan metrik sederhana untuk mengukur perbedaan dua atribut kategorik.

Performa dari kinerja algoritma diukur dengan *confusion matrix* dan akurasi. *Confusion matrix* untuk menganalisis performa algoritma dalam mengenali kategori.

Tabel 1. *Confusion Matrix* untuk Klasifikasi *Multiclass*

		Nilai Prediksi			
		1	**2**	**3**	**4**
Nilai Aktual	**1**	B_1	B_2	B_3	B_4
	2	B_5	B_6	B_7	B_8
	3	B_9	B_{10}	B_{11}	B_{12}
	4	B_{13}	B_{14}	B_{15}	B_{16}

Akurasi juga merupakan pengukuran performa kinerja suatu algoritma dengan persamaan

$$akurasi = \frac{jumlah\ prediksi\ benar}{jumlah\ seluruh\ prediksi} \times 100\% \quad (9)$$

Jika akurasi bernilai 100%, maka menunjukkan bahwa algoritma benar melakukan seluruh prediksi (Widodo 2022).

3. METODE PENELITIAN

Penelitian ini menggunakan data astronomi yang diperoleh dari *website* Kaggle. Data pada *website* Kaggle tersebut bersumber dari basis data NASA mengenai sifat-sifat bintang. Data yang digunakan merupakan data astronomi berupa analisis bintang pada tahun 2024 yang terdiri dari 240 baris dan 7 kolom. Dari ketujuh kolom tersebut digunakan sebagai variabel penelitian dengan 1 variabel target dan 6 variabel prediktor, yaitu:

1. *Star type* merupakan kategori tipe bintang, 0-*Brown dwarf*, 1-*Red dwarf*, 2-*White dwarf*, 3-*Main sequence*, 4-*Supergiants*, 5-*Hypergiants.*
2. *Temperature* atau suhu bintang.

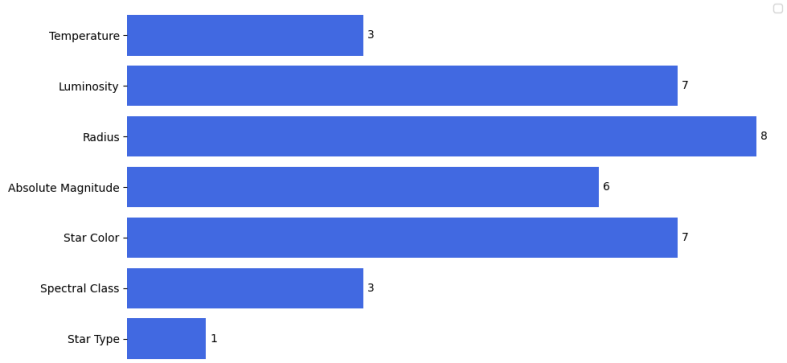
3. *Luminosity* merupakan tingkat kecerahan bintang dibandingkan tingkat kecerahan matahari
4. *Radius* merupakan perbandingan ukuran bintang dengan Matahari
5. *Absolute magnitude* merupakan kecerahan asli bintang
6. *Star color* merupakan warna bintang yang terdiri dari *blue, blue-white, red, white,* dan *yellow-white.*
7. *Spectral class* merupakan kelas spektrum bintang yang terdiri dari kategori A, B, F, G, K, M, dan O.

Penelitian ini akan mengaplikasikan metode klasifikasi NCN dan k-NN dengan tahapan analisis sebagai berikut.

1. Menangani *missing value* dengan cara mengisikan median pada variabel bertipe numerik dan modus pada variabel bertipe kategorik.
2. Membagi data dengan proporsi 90% untuk data *training* dan 10% untuk data *testing.*
3. Melakukan transformasi data
4. Melakukan proses klasifikasi menggunakan algoritma NCN dengan langkah sebagai berikut:
   a. Menghitung *centroid* data *training* pada setiap kategori *star type* menggunakan rata-rata untuk variabel bertipe numerik dan modus untuk variabel bertipe kategorik
   b. Menentukan tetangga terdekat dari *centroid* data *training* terhadap data *testing* menggunakan metode pengukuran, seperti jarak Euclidean pada Persamaan (2), jarak Manhattan pada Persamaan (3), jarak Minkowski pada Persamaan (4), dan jarak Chebyshev pada Persamaan (5), jarak Cosine pada Persamaan (6), serta jarak Jaccard pada Persamaan (7) dan jarak Hamming pada Persamaan (8).
   c. Memberikan label berupa *star type* berdasarkan jarak *centroid* dan data *testing* terdekat
   d. Melakukan evaluasi performa model klasifikasi pada algoritma NCN menggunakan pengukuran akurasi.
5. Melakukan proses klasifikasi menggunakan algoritma k-NN dengan langkah sebagai berikut:
   a. Menentukan nilai *k*
   b. Menghitung jarak antara data *testing* terhadap data *training* dengan menggunakan metode pengukuran, seperti jarak Euclidean pada Persamaan (2), jarak Manhattan pada Persamaan (3), jarak Minkowski pada Persamaan (4), dan jarak Chebyshev pada Persamaan (5), jarak Cosine pada Persamaan (6), serta jarak Jaccard pada Persamaan (7) dan jarak Hamming pada Persamaan (8).
   c. Mengurutkan hasil perhitungan jarak dari nilai terendah hingga nilai tertinggi
   d. Mengumpulkan data yang telah diurutkan berdasarkan nilai *k* yang ditentukan
   e. Memberikan label pada data *testing* berdasarkan kategori *star type* yang paling banyak muncul
   f. Melakukan evaluasi performa model klasifikasi pada algoritma NCN menggunakan pengukuran akurasi.
6. Membandingkan performa antara algoritma NCN dan k-NN yang memiliki hasil akurasi tertinggi.
7. Proses diakhiri dengan penarikan kesimpulan berdasarkan hasil penelitian yang diperoleh.

4. HASIL DAN PEMBAHASAN

Data yang digunakan terdiri dari 240 baris dan 7 kolom, di mana 25 baris di antaranya mengandung *missing value* dengan jumlah kolom kosong yang bervariasi di setiap barisnya.

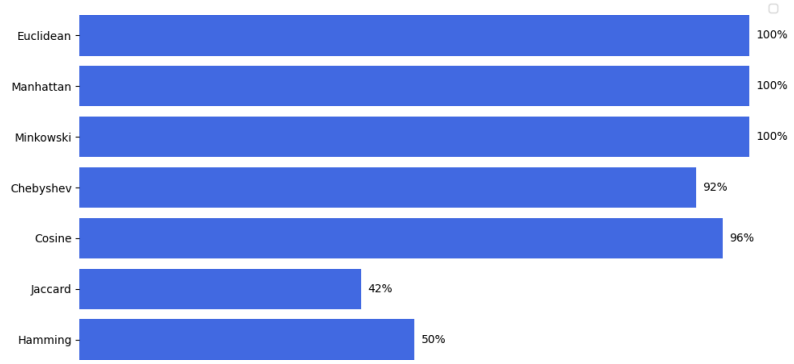


Gambar 1. Sebaran *Missing Value*

Berdasarkan Gambar 1, diketahui bahwa terdapat *missing value* dengan jumlah yang bervariasi pada setiap variabel. *Missing value* paling banyak ditemukan pada variabel *radius* sebanyak 8 atau 3.33%, sedangkan yang paling sedikit terdapat pada variabel *star type* yaitu hanya 1 atau 0.42% dari seluruh data. Penanganan terhadap *missing value* ini dilakukan dengan teknik imputasi atau pengisian data.

Teknik imputasi yang diterapkan pada variabel numerik dilakukan dengan mengisi *missing value* dengan nilai median yang sifatnya lebih fleksibel, terutama jika ditemukan data yang tidak simetris. Sedangkan teknik imputasi pada variabel kategorik menggunakan nilai modus, dimana modus adalah salah satu ukuran pemusatan data yang dapat diterapkan untuk atribut kualitatif yaitu pengamatan yang memiliki frekuensi tertinggi. Melalui prosedur tersebut, variabel numerik diisi dengan nilai 5800 untuk variabel *temperature*, 0.153 untuk variabel *luminosity*, 0.83 untuk variabel *radius*, dan 10.15 untuk variabel *absolute magnitude*. Sedangkan pada variabel kategorik, *missing value* diisi dengan modus, yaitu *red* pada *star color*, M pada *spectral class,* dan *red dwarf* pada *star type*. Setelah data lengkap, dilanjutkan dengan tahapan analisis lainnya.

Sebelum memasuki tahapan klasifikasi, data dibagi menjadi 90% data *training* dan 10% data *testing*. Variabel bertipe numerik dilakukan proses normalisasi menggunakan *z-score* pada Persamaan (1) dan variabel bertipe kategorik dikonversi menggunakan *one-hot encoding* untuk penerapan jarak Cosine, serta dikonversi menjadi *float* untuk jarak Jaccard dan Hamming. Berdasarkan tahapan-tahapan tersebut diperoleh performa pada setiap *distance measures* untuk algoritma NCN seperti disajikan pada Gambar 2 berikut.

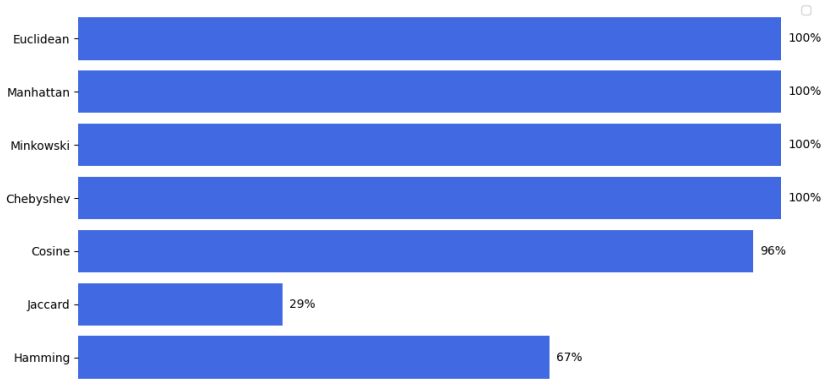


Gambar 2. Performa Algoritma NCN

Berdasarkan Gambar 2, akurasi tertinggi mencapai 100% diperoleh pada algoritma NCN dengan jarak Euclidean, Manhattan, dan Minkowski. Hal ini menunjukkan bahwa algoritma NCN dengan ketiga *distance measures* tersebut mampu mengklasifikasikan semua *star type* pada data *testing* dengan benar tanpa kesalahan. Dengan kata lain, setiap bintang dalam data *testing* berhasil diklasifikasikan sesuai dengan *star type* sebenarnya. Performa tersebut menunjukkan bahwa jarak Euclidean, Manhattan, dan Minkowski merupakan *distance measures* yang paling optimal untuk diterapkan dalam klasifikasi bintang menggunakan algoritma NCN.

Sebaliknya, akurasi terendah diperoleh pada algoritma NCN dengan jarak Jaccard, yaitu sebesar 42%. Nilai ini menunjukkan bahwa algoritma NCN dengan jarak Jaccard hanya mampu mengklasifikasikan 42% bintang dalam data *testing* dengan benar, sedangkan 58% bintang lainnya diklasifikasikan tidak sesuai dengan *star type* sebenarnya.

Klasifikasi bintang juga dilakukan menggunakan algoritma k-NN dengan berbagai *distance measures* dan nilai *k* sebesar 3, sehingga diperoleh performa algoritma k-NN dengan berbagai *distance measures* seperti disajikan pada Gambar 3 berikut.



Gambar 3. Performa Algoritma k-NN

Berdasarkan Gambar 3, akurasi tertinggi mencapai 100% diperoleh pada algoritma k-NN dengan jarak Euclidean, Manhattan, Minkowski, dan Chebyshev. Hal ini menunjukkan bahwa algoritma k-NN dengan keempat *distance measures* tersebut mampu mengklasifikasikan semua *star type* pada data *testing* dengan benar tanpa kesalahan. Dengan kata lain, setiap bintang dalam data *testing* berhasil diklasifikasikan sesuai dengan *star type* sebenarnya. Performa tersebut menunjukkan bahwa jarak Euclidean, Manhattan, Minkowski, dan Chebyshev merupakan *distance measures* yang paling optimal untuk diterapkan dalam klasifikasi bintang menggunakan algoritma k-NN.

Sebaliknya, akurasi terendah diperoleh pada algoritma k-NN dengan jarak Jaccard, yaitu sebesar 29%. Nilai ini menunjukkan bahwa algoritma k-NN dengan jarak Jaccard hanya mampu mengklasifikasikan 29% bintang dalam data *testing* dengan benar, sedangkan 71% bintang lainnya diklasifikasikan tidak sesuai dengan *star type* sebenarnya. Keberhasilan pada jarak Euclidean, Manhattan, Minkowski, dan Chebyshev dapat berkaitan dengan proses normalisasi menggunakan *z-score* pada tahap *preprocessing*. Normalisasi yang bertujuan menyeragamkan skala data yang berbeda dan pengukuran jarak yang bergantung pada selisih nilai, mengindikasikan hasil perhitungan kemiripan yang akurat. Sedangkan pada jarak Jaccard dan Hamming memberikan informasi bahwa variabel kategorik dalam data ini belum berperan kuat secara tunggal dalam membedakan *star type*

Secara keseluruhan, algoritma k-NN menunjukkan performa yang lebih luas dibandingkan metode NCN karena kemampuannya mencapai akurasi sempurna pada lebih banyak tipe ukuran jarak yang digunakan. Hal ini menunjukkan bahwa pendekatan tetangga

terdekat pada k-NN memiliki fleksibilitas lebih tinggi dalam mengklasifikasikan *star type* dibandingkan pendekatan berbasis *centroid* pada NCN.

5. KESIMPULAN

Berdasarkan hasil analisis klasifikasi bintang dengan proporsi data *training* sebesar 90% dan data *testing* sebesar 10% dengan perbandingan *distance measures* pada algoritma NCN dan k-NN, diperoleh hasil bahwa algoritma k-NN lebih unggul dibandingkan algoritma NCN pada 4 *distances measures*. Keempat *distances measures* tersebut adalah jarak Euclidean, Manhattan, Minkowski, dan Chebyshev dengan akurasi sebesar 100%. Sedangkan, jarak Chebyshev pada algoritma NCN memiliki akurasi 92% dan jarak Cosine kedua algoritma memiliki akurasi sebesar 96%. Namun, pada jarak Jaccard dan Hamming, pada kedua algoritma, menunjukkan akurasi yang lebih rendah yaitu di bawah 70%.

DAFTAR PUSTAKA

Chaudhuri, B. B., 1996. A new definition of neighborhood of a point in multi-dimensional space. *Pattern Recognition Letters,* pp. 11-17.

Cholil, S. R., Handayani, T., Prathivi, R. & Ardianita, T., 2021. Implementasi algoritma klasifikasi K-Nearest Neighbor (KNN) untuk klasifikasi seleksi penerima beasiswa. *Indonesian Journal on Computer and Information Technology,* pp. 118-127.

Fix, E. & Hodges, J. L., 1951. *Discriminatory analysis. nonparamteric discrimination: consistency properties,* Texas: USAF School of Aviation.

Garrison, R. F. et al., 1953. Stellar classification (classification stellaire). *A Checklist of Regional Archaelogical Journals,* pp. 631-647.

Han, J., Kamber, M. & Pei, J., 2012. *Data mining concepts and techniques.* Waltham: Elsevier Inc..

NASA Universe Web Team, 2024. *The lives, times, and deaths of stars.* [Online] Available at: https://science.nasa.gov/universe/the-lives-times-and-deaths-of-stars/ [Diakses 22 September 2024].

Nayak, S., Bhat, M., Reddy, N. V. S. & Rao, B. A., 2022. Study of distance metrics on k-nearest neighbor algorithma for star categorization. *Journal of Physics: Conference Series,* pp. 1-8.

Noble, J., 2026. *What are distance metrics?.* [Online] Available at: https://www.ibm.com/think/topics/distance-metrics [Diakses April 2026].

Reihanah, K. N., Maruddani, D. A. I. & Widiharih, T., 2023. Clustering karakteristik industri kecil dan menengah di Kota Kendari menggunakan algoritma k-Prototypes. *JURNAL GAUSSIAN,* pp. 340-351.

Samuels, J. I., 2024. One-hot encoding and two-hot encoding: an introduction.

Saputra, I. & Kristiyanti, D. A., 2022. *Machine learnig untuk pemula.* Jakarta: Penerbit INFORMATIKA.

Tamatjita, E. N. & Mahastama, A. W., 2016. Comparison of music genre classification using nearest centroid classifier and k-nearest neighbours. *International Conference on Information Management and Technology (ICIMTech),* pp. 118-123.

Utomo, S., Subroto, I. M. I. & Riansyah, A., 2022. Deteksi plagiat tugas akhir dengan metode Jaccard Similarity. *Jurnal Transistor Elektro dan Informatika,* pp. 132-141.

Wahyono, et al., 2020. Perbandingan penghitungan jarak pada k-nearest neighbour dalam klasifikasi data tekstual. *Jurnal Teknologi dan Sistem Komputer,* pp. 54-58.

Widodo, R. B., 2022. *Machine learning metode k-Nearest Neigbors klasifikasi angka bahasa isyarat.* Malang: Media Nusa Creative.

Wulandari, T. K., Oktaviani, E. D. & Lestari, A., 2022. Penerapan metode binary search dan hamming distance pada e-library SMAN 2 Katingan Hilir. *KONSTELASI: Konvergensi Teknologi dan Sistem Informasi,* pp. 33-42.