Jurnal Sains dan Edukasi Sains
Vol.5, No.1, Februari 2022: 28-37
https://doi.org/10.24246/juses.v5i1p28-37

Perbandingan Penggunaan Jarak Manhattan, Jarak Euclid, dan Jarak Minkowski dalam Klasifikasi Menggunakan Metode KNN pada Data Iris

Adi Setiawan[1,*]

[1]Departemen Matematika dan Sains Data, Fakultas Sains dan Matematika, Universitas Kristen Satya Wacana, Salatiga, Jawa Tengah 50711

*email korespondensi: adi.setiawan@uksw.edu

Received : 31 Januari 2022; ***Revised*** : 10 Februari 2022; ***Accepted*** : 11 Mei 2022; ***Published*** : 22 Mei 2022

ABSTRAK

Dalam klasifikasi, ukuran kebaikan suatu metode ditentukan oleh akurasi dan skor F1. Penelitian ini bertujuan untuk menentukan rata-rata akurasi dan skor F1 pada klasifikasi menggunakan metode KNN pada data Iris dengan bantuan program Python. Rata-rata akurasi maupun skor F1 ditentukan dengan membuat variasi parameter yang digunakan yaitu random state dari 1 sampai dengan 10000, k yang digunakan dalam metode KNN dari 1 sampai dengan 31 (tergantung prosentase data uji yang digunakan), q (definisi jarak yang digunakan yaitu jarak Manhattan, jarak Euclid, dan jarak Minkowski) serta persentase data uji yang digunakan. Dalam penelitian ini, digunakan persentase data uji 10%, 20 %, 30 %, dan 40 %. Jika digunakan k = 5 dalam metode KNN, variasi random state 1 sampai dengan 10000 dan digunakan data uji 20 % maka diperoleh akurasi rata-rata 96,6667 %. Demikian juga dapat diperoleh histogram dari akurasi tersebut. Histogram tersebut cenderung menceng ke kiri. Dalam penelitian ini, juga diperoleh hasil bahwa dengan melakukan variasi k dalam metode KNN juga dapat diperoleh akurasi maupun skor F1 yaitu bahwa makin besar data uji yang digunakan maka makin kecil nilai k untuk mencapai akurasi ataupun skor F1 maksimum. Lebih lanjut, dalam perubahan nilai p pada jarak Minkowski, apabila p membesar maka nilai maksimum rata-rata akurasi akan cenderung membesar namun nilai k cenderung mengecil.

Kata-kata kunci: *data Iris; jarak Euclid; jarak Manhattan; jarak Minkowski; metode klasifikasi KNN*

PENDAHULUAN

Berbagai metode klasifikasi yang dapat digunakan antara lain metode *Naive Bayes*, metode KNN (K-*Nearest Neighbors*), metode *Random Forest*, metode SVM (*Support Vector Machine*), dan metode *Neural Network*. Metode KNN banyak digunakan untuk klasifikasi karena relatif lebih sederhana dibandingkan metode lainnya (Baharuddin et al., 2019; Dewi & Dwidasmara, 2020; Handayani, 2020; Hidayati & Hermawan, 2021; Khamdani et al., 2021; Mara et al., 2021; Pamungkas & Kharisudin, 2021; Prakisya et al., 2021; Purwaningsih & Nurelasari, 2021; Retnoningsih & Pramudita, 2020). Pada penelitian (Purwaningsih & Nurelasari, 2021) digunakan metode KNN dengan membuat variasi nilai *k* dari 1 sampai dengan 10 namun tidak dilakukan variasi besaran data uji.

Data iris sering digunakan sebagai contoh untuk menganalisis metode, termasuk artikel (Djakaria et al., 2010), (Febrianti et al., 2016) dan (Hermanto, 2017). Namun, studi klasifikasi menggunakan metode KNN belum dilakukan pada data iris, yaitu dengan melakukan variasi *k* dalam metode KNN, variasi persentase data uji, dan berbagai jenis jarak yang digunakan. Dalam penelitian ini, kami menentukan akurasi dan F1 dalam metode KNN yang dapat digunakan untuk mengklasifikasikan data iris dengan melakukan variasi nilai *k*, persentase data uji, dan jarak yang digunakan. Studi simulasi juga dilakukan dengan mengubah klasifikasi peubah respons.

EKSPERIMEN

Dalam penelitian ini hanya mencakup metode KNN. Metode KNN merupakan salah satu metode yang digunakan untuk mengklasifikasikan data berdasarkan data latih (*training data*). Metode ini merupakan bagian dari pembelajaran terawasi (*supervised learning*), data terbaru dikategorikan berdasarkan kedekatan jumlah kategori yang ada. Hal itu berarti bahwa, algoritma KNN mengklasifikasikan data baru berdasarkan

jarak *datum* dan kedekatannya dengan data lain. Dalam hal ini jarak yang digunakan adalah jarak *Euclidean*, jarak *Manhattan* (*City block*), dan jarak *Minkowski*. Jarak *Euclid* didefinisikan sebagai

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}, \qquad (1)$$

jarak *Manhattan* atau *City block* :

$$d(x,y) = \sum_{i=1}^{n} |x_i - y_i|, \qquad (2)$$

dan jarak *Minkowski* :

$$d(x,y) = \left(\sum_{i=1}^{n} |x_i - y_i|^q \right)^{1/q}, \qquad (3)$$

dengan q bilangan bulat positif, $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ dan $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$ dalam $\mathbf{R}^n$ (Haviluddin et al., 2020).

Konsep dasar dari metode KNN adalah mencari jarak terpendek antara data yang direklasifikasi dalam lingkungan *k* (jarak minimum) yang paling dekat dengan data latih. Jumlah kelas/kategori terbesar dengan jarak terkecil menjadi data baru, yaitu kelas/kategori data uji (*testing data*). Keuntungan dari metode KNN adalah sederhana dan efektif bila datanya cukup besar dan bekerja dengan cukup baik, dan metode ini tahan terhadap penyimpanan data *noise* (AlZoman & Alenazi, 2021). Namun, metode ini juga memiliki kelemahan, yaitu semakin banyak data pelatihan yang digunakan, semakin lama waktu perhitungan yang akan digunakan. Metode ini sensitif terhadap sifat *redundant* atau tidak relevan karena semua data menghitung jarak dari semua data uji.

Untuk itu karakteristik yang penting saja yang sebaiknya digunakan dan tidak mudah ditentukan jarak mana yang lebih sesuai untuk sebagian besar data. Algoritma KNN dapat dijelaskan berikut ini:

1. Pilih parameter *k* yaitu banyaknya tetangga terdekat (*nearest neighbors*).
2. Dihitung jarak antara *datum* yang akan ditentukan klasifikasinya dengan data latih.
3. Mengurutkan jarak yang diperoleh pada langkah ke-2 dalam urutan naik dan menentukan jarak terkecil sampai pada urutan ke-*k*.
4. Tentukan kelas/kategori yang bersesuaian dengan langkah ke-3.
5. Kelas/kategori yang terbanyak pada langkah ke-4 menjadi kelas *datum* baru tersebut.

Untuk memberikan gambaran bagaimana metode KNN itu diimplementasikan, digunakan contoh berikut ini. Misalkan dimiliki data peubah penjelas (*explanatory variable*) X_1 dan peubah penjelas X_2 sedangkan peubah respons *Y* yang menyatakan kelas dari pasangan (X_1, X_2). Secara lengkap data tersebut dinyatakan pada **Tabel 1**.

Tabel 1. Tabel Contoh Data untuk Penggambaran Metode KNN

X_1	X_2	*Y*
9	5	A
5	6	B
5	7	B
8	8	A
6	7	B
7	6	A

Apabila dimiliki pasangan berurutan (X_1, X_2) = (7, 4) dan dengan menggunakan *k* = 3 akan ditentukan kelas atau peubah respons *Y* yang dipandang sesuai berdasarkan algoritma KNN. Untuk menentukan peubah respons yang sesuai digunakan langkah-langkah berikut ini:

1. Memilih *k* dan dalam contoh ini dipilih *k* = 3.
2. Menghitung jarak antara *datum* yang akan diklasifikasikan dengan data latih.

Tabel 2. Tabel Perhitungan Jarak *Euclid* untuk Contoh Data pada **Tabel 1**

X_1	X_2	Kuadrat jarak ke *test datum* (X_1, X_2) = (7, 4)
9	5	$(9-7)^2 + (5-4)^2 = 5$
5	6	$(5-7)^2 + (6-4)^2 = 8$
5	7	$(5-7)^2 + (7-4)^2 = 13$
8	8	$(8-7)^2 + (8-4)^2 = 17$
6	7	$(6-7)^2 + (7-4)^2 = 10$
7	6	$(7-7)^2 + (6-4)^2 = 4$

3. Mengurutkan jarak dalam urutan naik dan menentukan jarak terdekat sampai pada urutan ke-*k* dengan *k* = 3.

Tabel 3. Tabel Penggambaran Penentuan Jarak Terkecil ke Terbesar dan Penentuan Masuk dalam KNN atau tidak

X_1	X_2	Kuadrat jarak ke *test datum* (X_1, X_2) = (7, 4)	Mengurutkan dari jarak terkecil ke terbesar	Menentukan apakah termasuk dalam KNN atau tidak
9	5	$(9-7)^2 + (5-4)^2 = 5$	2	Ya
5	6	$(5-7)^2 + (6-4)^2 = 8$	3	Ya
5	7	$(5-7)^2 + (7-4)^2 = 13$	5	Tidak
8	8	$(8-7)^2 + (8-4)^2 = 17$	6	Tidak
6	7	$(6-7)^2 + (7-4)^2 = 10$	4	Tidak
7	6	$(7-7)^2 + (6-4)^2 = 4$	1	Ya

4. Pada langkah ini, menentukan kelas yang bersesuaian dengan langkah ke-3. Hal itu berarti *datum* nomor 1, 2, 5 dan 6 dengan kelas/kategori berurut-turut adalah A, B, B, dan A.

Tabel 4. Tabel Penggambaran Penentuan Kelas/Kategori dalam Metode KNN

X_1	X_2	Kuadrat jarak ke test *datum* (X_1, X_2) = (7, 4)	Mengurutkan dari jarak terkecil ke terbesar	Menentukan apakah termasuk dalam KNN atau tidak	Y = kelas/kategori nearest neighbors
9	5	$(9-7)^2 + (5-4)^2 = 5$	2	Ya	A
5	6	$(5-7)^2 + (6-4)^2 = 8$	3	Ya	B
5	7	$(5-7)^2 + (7-4)^2 = 13$	5	Tidak	-
8	8	$(8-7)^2 + (8-4)^2 = 17$	6	Tidak	-
6	7	$(6-7)^2 + (7-4)^2 = 10$	4	Tidak	-
7	6	$(7-7)^2 + (6-4)^2 = 4$	1	Ya	A

5. Berdasarkan langkah ke-4 maka diperoleh banyaknya kelas/kategori A adalah 2 sedangkan banyaknya kelas/kategori B adalah 1 sehingga dengan menggunakan algoritma KNN maka (X_1, X_2) = (7, 4) lebih dekat ke kelas A dibandingkan dengan ke kelas B sehingga diputuskan bahwa pasangan yang dievaluasi masuk dalam kelas A.

Apabila digunakan pemilihan *k* yang lain maka berturut-turut untuk *k* = 1, 2, 4, 5 dan 6 akan diperoleh hasil klasifikasi (X_1, X_2) = (7, 4) sebagai A, A, tidak dapat ditentukan (karena sama banyaknya antara kategori A dan B), B dan tidak dapat ditentukan.

Tabel 5. Tabel Klasifikasi

Kelas	Terklasifikasi Positif	Terklasifikasi Negatif
Positif	TP	FN
Negatif	FP	TN

Dalam penggunaan, tentu saja metode KNN tidak hanya digunakan untuk mengklasifikasikan 1 *datum* tetapi sekelompok *datum* yang dikenal dengan data uji. Untuk mengukur kinerja algoritma klasifikasi digunakan *confusion matrix*. Pada kasus klasifikasi 2 kelas, dalam *confusion matrix* terdapat informasi yang dapat digunakan untuk membandingkan hasil klasifikasi yang dilakukan oleh metode KNN dengan klasifikasi yang sebenarnya. Dalam hal ini, terdapat 4 istilah yang digunakan yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN). Masing-masing dapat diartikan sebagai berikut: TP yaitu banyaknya *datum* positif yang terklasifikasi dengan benar oleh metode KNN, TN yaitu banyaknya *datum* negatif yang terklasifikasi dengan benar oleh metode KNN, FN yaitu banyaknya *datum* negatif namun terklasifikasi salah oleh metode KNN dan FP yaitu banyaknya *datum* positif namun terklasifikasi salah oleh metode KNN. Dalam bentuk tabel, hal tersebut dapat dinyatakan pada **Tabel 5** dan akurasi (*accuracy*) metode KNN dirumuskan sebagai:

$$A = \frac{TP + TN}{TP + TN + FP + FN}, \quad (4)$$

yaitu merupakan rasio antara prediksi benar dan keseluruhan data, presisi (*precision*) dirumuskan sebagai:

$$P = \frac{TP}{TP + FP}, \quad (5)$$

yaitu merupakan rasio antara prediksi benar positif dan keseluruhan hasil yang diprediksi positif. Sensitivitas (*recall*) dirumuskan sebagai:

$$R = \frac{TP}{TP + FN}, \quad (6)$$

yaitu merupakan rasio antara prediksi benar positif dan keseluruhan data yang positif dan yang terakhir F1 *score* dirumuskan sebagai:

$$F1 = 2\frac{P * R}{P + R}, \quad (7)$$

yaitu merupakan perbandingan rata-rata presisi dan *recall* (Nugroho, 2020). Akurasi mempunyai nilai yang hampir sama dengan F1. Apabila klasifikasi lebih dari 2 kelas maka akurasinya dapat diperoleh dari perbandingan antara jumlah elemen diagonal dalam *confusion matrix* dibagi dengan jumlah elemen keseluruhan dalam *confusion matrix*, namun nilai F1 tidak dapat ditentukan untuk kasus ini.

Dataset Iris merupakan data yang diambil dari UCI *repository* yaitu: https://archive.ics.uci.edu/ml/datasets/iris. Data bunga Iris mempunyai 5 atribut yaitu *sepal length*, *sepal width*, *petal length*, *petal width* dan *variety*. Data 4 atribut yang pertama merupakan atribut numerik sedangkan atribut variety merupakan data kategorikal yang terdiri dari varietas bunga iris yaitu *setosa*, *virginica*, dan *versicolor*. **Tabel 6** merupakan sebagian data dari *dataset* Iris. Penelitian ini dilakukan dengan melakukan klasifikasi dengan metode KNN menggunakan data iris dan melakukan variasi *k*, bilangan *random* awal yang digunakan, persentase data uji dan definisi jarak yang digunakan.

Tabel 6. Sebagian data Iris yang mempunyai 4 atribut

No.	*Sepal.length*	*Sepal.width*	*Petal.length*	*Petal.width*	*Variety*
1	5,1	3,5	1,4	0,2	*Setosa*
2	4,9	3	1,4	0,2	*Setosa*
3	7	3,2	4,7	1,4	*Versicolor*
4	6,4	3,2	4,5	1,5	*Versicolor*
5	6,3	3,3	6	2,5	*Virginica*
6	5,8	2,7	5,1	1,9	*Virginica*

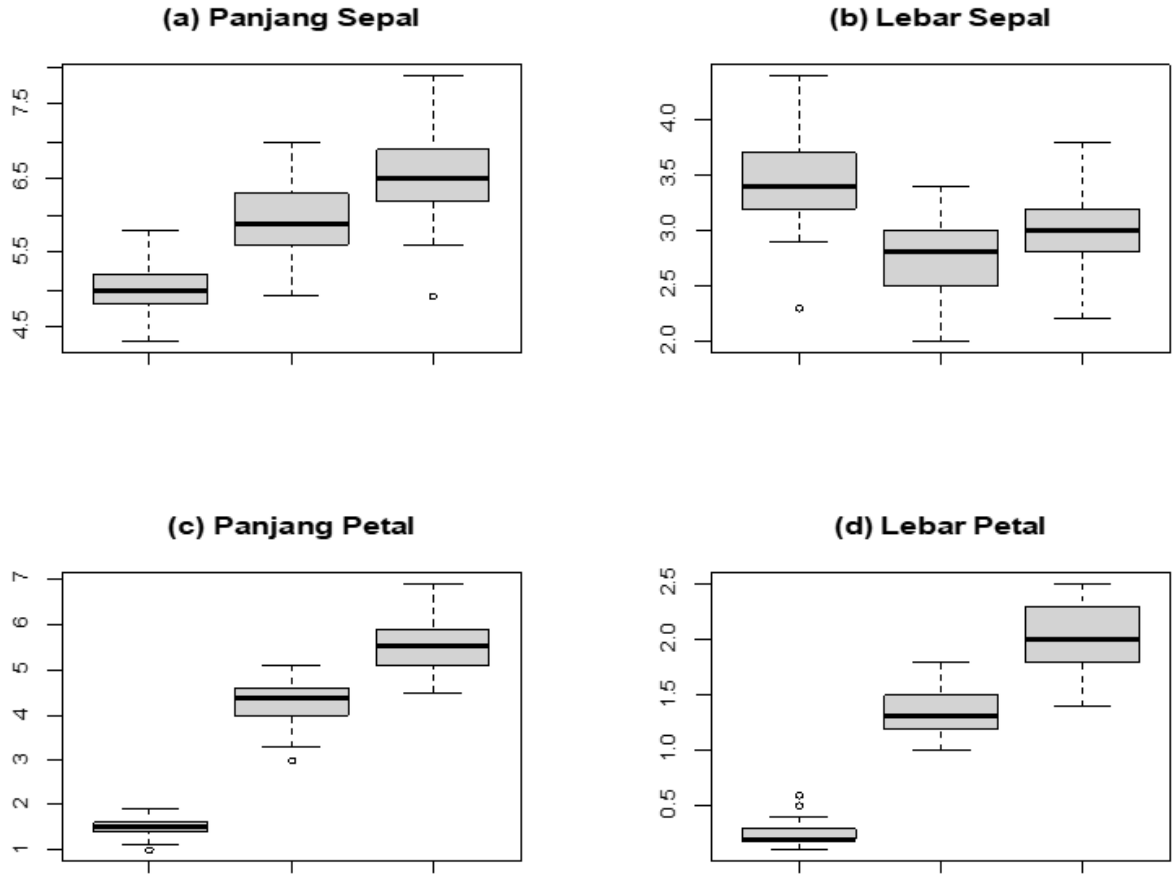
HASIL DAN DISKUSI

Untuk memberikan kesan pertama dari masing-masing atribut, biasanya ditentukan ringkasan statistiknya. **Tabel 7** mempresentasikan ringkasan statistik dari tiap-tiap atribut pada data Iris. Sebagai contoh, berturut-turut nilai minimum, rata-rata (*mean*), median, simpangan baku (*standard deviation*), dan nilai maksimum untuk panjang sepal (*sepal.length*) adalah 4,3; 5,84; 5,8; 0,84; dan 7,9 (dalam cm). Lebih lanjut, *skewness*, dan *excess kurtosis* dari panjang sepal berturut-turut adalah 0,3149 dan -0,5521. Hal itu berarti bahwa data panjang sepal condong ke kanan sehingga lebih banyak yang mempunyai panjang kecil dibandingkan dengan yang mempunyai panjang besar. Demikian juga, *excess kurtosis* bernilai negatif berarti bahwa distribusi data panjang sepal lebih tipis ekornya dibandingkan distribusi normal. Di samping itu, koefisien korelasi (*coefficient of correlation*) antara 4 atribut yaitu panjang sepal (sepal.length), lebar sepal, panjang petal (*petal.length*), dan lebar petal (*petal.width*) dengan atribut *variety* berturut-turut adalah 0,78; -0,43; 0,95; dan 0,96. Terlihat bahwa panjang petal dan lebar petal lebih berpengaruh terhadap pemisahan *variety* dibandingkan dengan atribut yang lain yaitu panjang sepal dan lebar sepal. **Gambar 1** menyatakan *boxplot* dari masing-masing kelas dan masing-masing atribut yaitu panjang sepal, lebar sepal, panjang petal dan lebar petal.

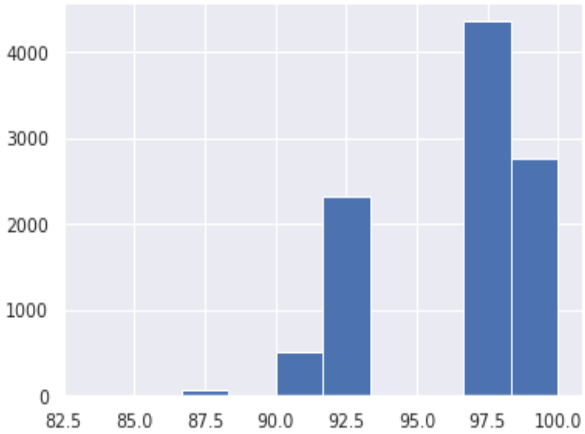
Tabel 7. Ringkasan statistik berdasarkan data Iris yang mempunyai 4 atribut

Statistik	*Sepal.length*	*Sepal.width*	*Petal.length*	*Petal.width*
Minimum	4,3	2	1	0,1
Rata-rata (*mean*)	5,84	3,06	3,76	1,20
Median	5,8	3	4,53	1,3
Simpangan Baku	0,83	0,44	1,77	0,76
Maksimum	7,9	4,4	6,9	2,5
Skewness	0,3149	0,3190	-0,2749	-0,1030
Excess Kurtosis	-0,5521	0,2282	-1,4021	-1,3406

Apabila metode KNN digunakan untuk mengklasifikasikan data Iris dengan menggunakan *k* = 5, jarak *Euclid*, data uji (*testing data*) 20 %, dan menggunakan variasi *random state* 1 sampai 10000 maka akan diperoleh nilai minimum, rata-rata, median, simpangan baku dan nilai maksimum dari akurasi berturut-turut adalah 83,3333; 96,4070; 96,6667; 2,9276; dan 100 (dalam %). Histogram dari hasil variasi akurasi dinyatakan pada **Gambar 2**. Terlihat bahwa histogram dari nilai-nilai akurasi yang diperoleh menceng ke kiri sehingga ukuran pemusatan median yaitu 96,6667 lebih beralasan dibandingkan dengan rata-rata 96,4070. Selanjutnya, apabila dibuat variasi *k* dari *k* = 1 sampai dengan *k* = 30 dan variasi *random state* 1 sampai 10000 (mengingat waktu yang diperlukan untuk eksekusi program cukup lama) dan variasi persentase data uji mulai dari 10 %, 20 %, 30 %, dan 40 % sehingga berturut-turut diperoleh grafik yang dinyatakan pada **Gambar 3(a)**, (b), (c) dan (d). Apabila digunakan data uji 10 %, terlihat bahwa *k* akan turun dan mencapai minimum pada saat *k* = 2 (yaitu mempunyai rata-rata akurasi 95,3267 %) dan naik sampai mencapai maksimum pada saat *k* = 20 yaitu mempunyai rata-rata akurasi 96,4343 %. Hasil yang mirip juga dapat dilihat pada **Gambar 3(b), (c) dan (d)** untuk kasus data uji 20 %, 30 %, dan 40 %, namun nilai *k* yang makin mengecil untuk mencapai puncak.

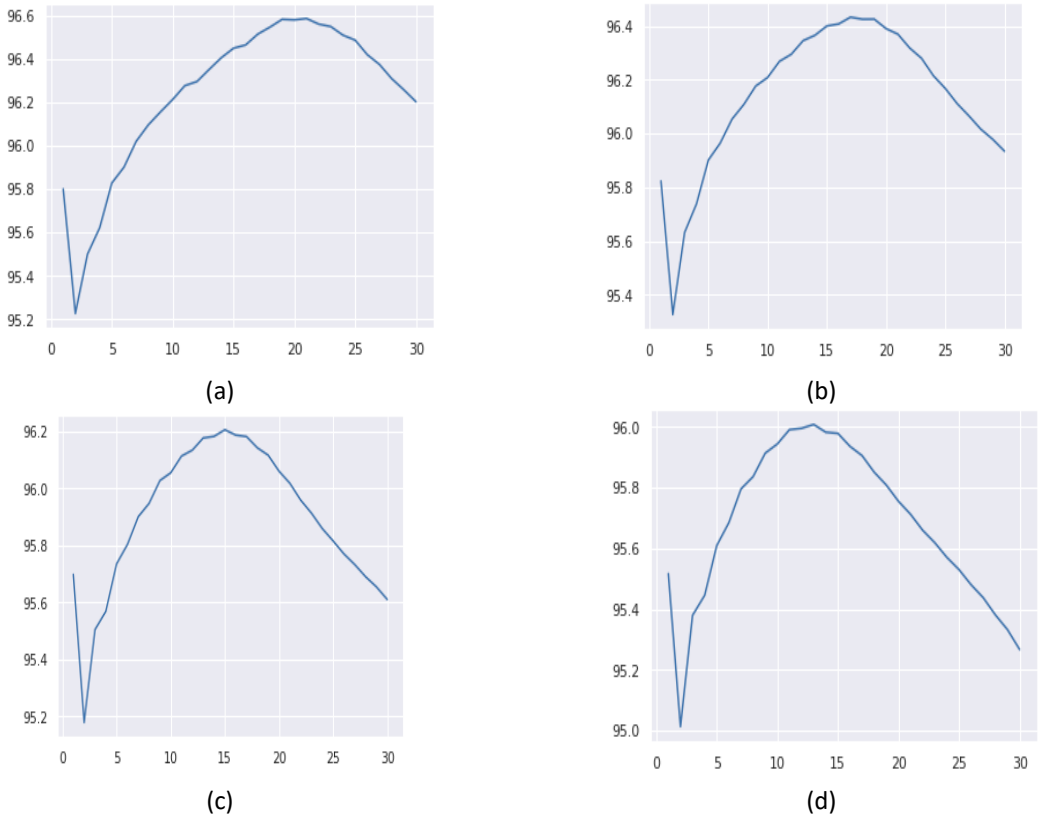


Gambar 1. *Boxplot* untuk tiap-tiap kelas (*setosa* (pertama)*, virginica* (kedua), dan *versicolor* (ketiga)) dan tiap-tiap atribut (panjang sepal, lebar sepal, panjang petal, dan lebar petal) dari data Iris

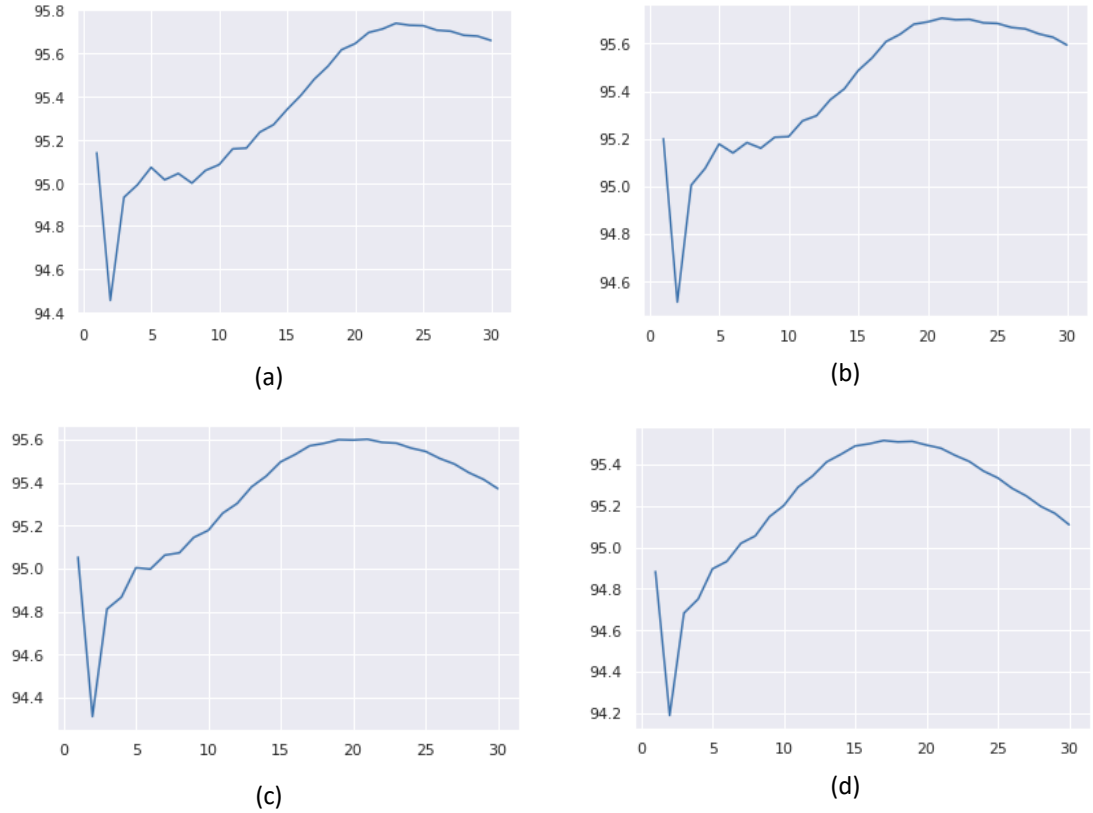


Gambar 2. Histogram dari akurasi dengan membuat variasi *random state* 1 sampai dengan 10000

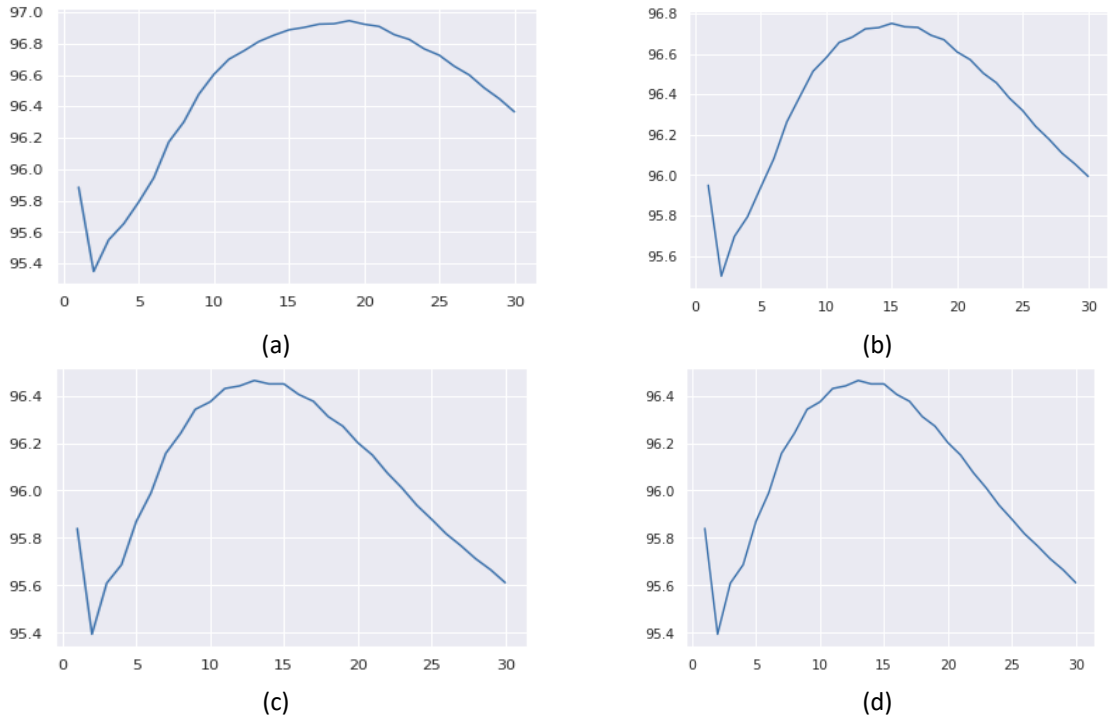
Apabila dilakukan perubahan definisi jarak yang digunakan yaitu dengan menggunakan jarak *Manhattan* dan variasi persentase data uji mulai dari 10 %, 20 %, 30 %, dan 40 % maka berturut-turut diperoleh grafik yang dinyatakan pada **Gambar 4 (a), (b), (c) dan (d)**. Terlihat bahwa *k* akan turun mencapai minimum pada saat *k* = 2 yaitu mempunyai rata-rata akurasi 94,5127 % dan naik akan mencapai maksimum pada saat *k* = 21 yaitu 95,7019 %. Demikian juga, apabila dilakukan perubahan definisi jarak menjadi jarak *Minkowski* untuk *q* = 3, variasi *k* dari *k* = 1 sampai dengan *k* = 30 dan variasi *random state* 1 sampai 10000 serta variasi persentase data uji dari 10 %, 20 %, 30 %, dan 40 % maka akan diperoleh grafik pada **Gambar 5(a), (b), (c), dan (d)**. Berdasarkan **Gambar 5(a)**, diperoleh bahwa rata-rata akurasi akan mencapai maksimum pada saat *k* = 19 yaitu 96,7511 %. Hasil yang analog juga bisa diperoleh untuk **Gambar 5(b), (c), dan (d)**, namun demikain puncak dicapai oleh nilai *k* yang makin mengecil. Dari **Gambar 3** sampai dengan **Gambar 5** dapat disimpulkan bahwa apabila *p* membesar maka nilai maksimum rata-rata akurasi akan cenderung membesar namun nilai *k* cenderung mengecil.



Gambar 3. Grafik rata-rata akurasi untuk *k* = 1, sampai dengan *k* = 30, variasi *random state* 1 sampai dengan 10000 dengan menggunakan definisi jarak *Euclid* dan variasi persentase data uji 10 %, 20 %, 30 %, dan 40 % berturut-turut untuk (a), (b), (c), dan (d)



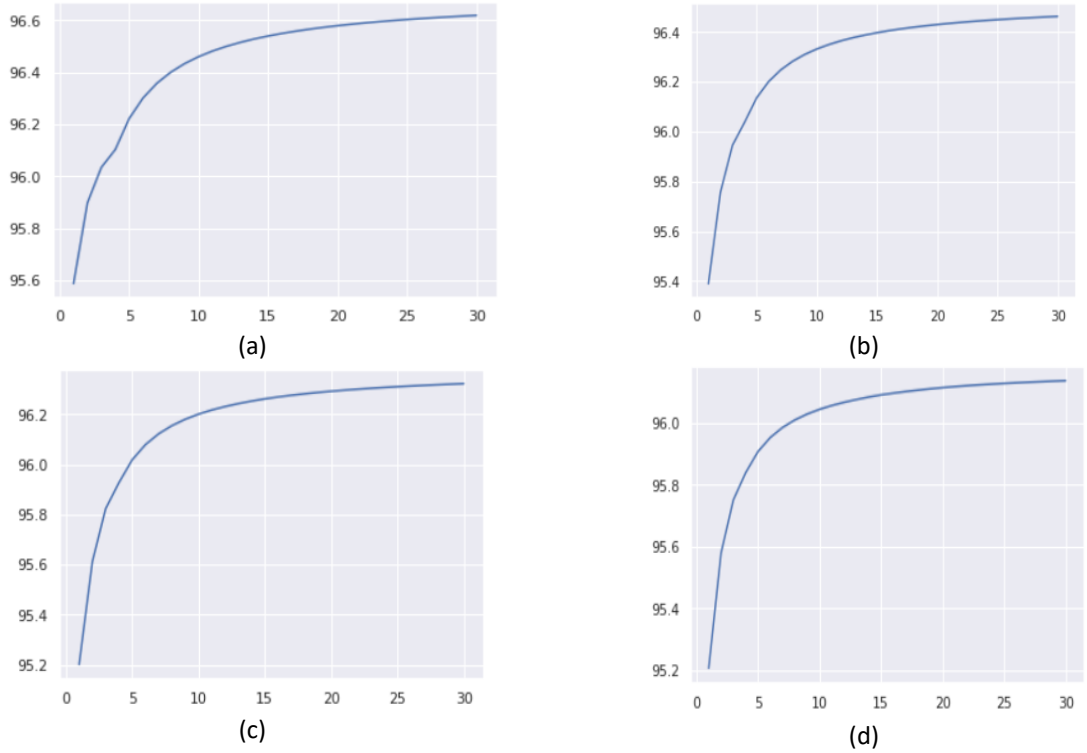
Gambar 4. Grafik rata-rata akurasi untuk *k* = 1, sampai dengan *k* = 30, variasi *random state* 1 sampai dengan 10000 dengan menggunakan definisi jarak *Manhattan* dan variasi persentase data uji 10 %, 20 %, 30 %, dan 40 % berturut-turut untuk (a), (b), (c), dan (d)



Gambar 5. Grafik rata-rata akurasi untuk *k* = 1, sampai dengan *k* = 30, variasi *random state* 1 sampai dengan 10000 dengan menggunakan definisi jarak *Minkowski* untuk *q* = 3 dan variasi persentase data uji 10 %, 20 %, 30 %, dan 40 % berturut-turut untuk (a), (b), (c), dan (d)

Apabila dilakukan studi simulasi dengan memberikan kode 1 untuk jenis bunga Iris *Setosa* dan *Versicolor* maka pada **Gambar 6** mempresentasikan grafik rata-rata F1 untuk *k* = 2, variasi *random state* dari 1 sampai dengan 10000 dengan menggunakan jarak *Minkowski* untuk *q* = 1 sampai dengan *q* = 31 dan data uji berturut-turut (a) 10 %, (b) 20 %, (c) 30 %, dan (d) 40 %. Terlihat bahwa rata-rata F1 mencapai naik untuk variasi *q* = 1 sampai dengan *q* = 31. Demikian juga, **Gambar 7** mempresentasikan grafik rata-rata F1 untuk *k* = 2, variasi *random state* dari 1 sampai dengan 10000 dengan menggunakan jarak *Minkowski* untuk *q* = 1

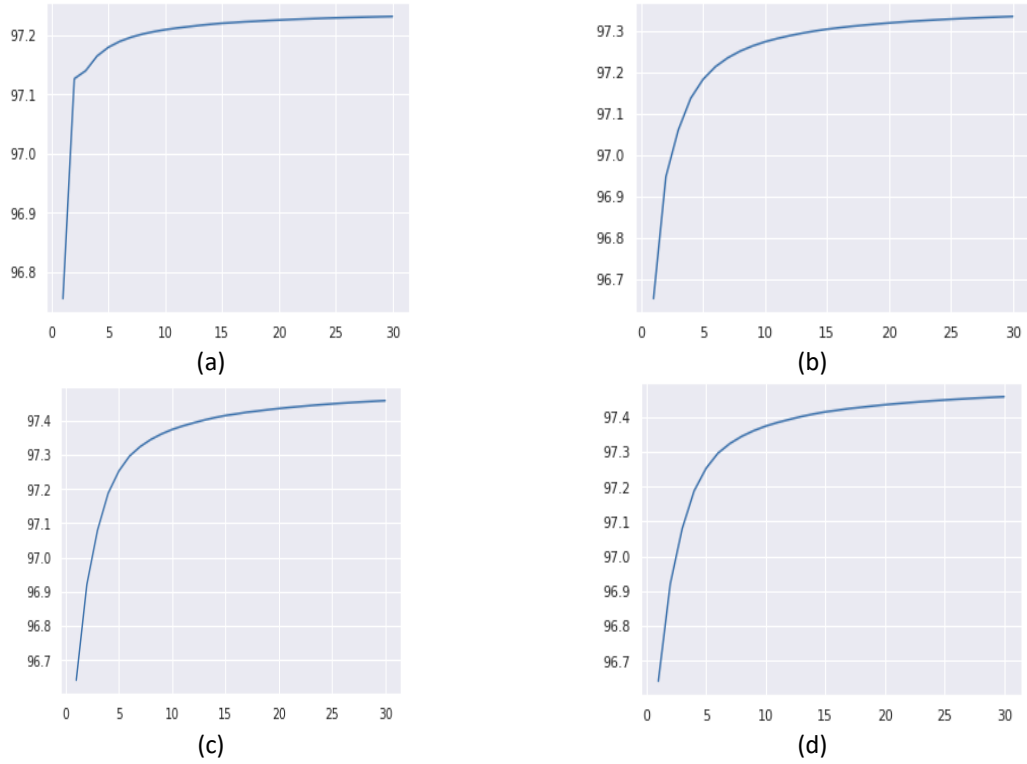
sampai dengan q = 31 dan data uji berturut-turut (a) 10 %, (b) 20 %, (c) 30 %, dan (d) 40 %. Seperti **Gambar 6**, pada **Gambar 7** juga terlihat bahwa rata-rata F1 mencapai naik untuk variasi q =1 sampai dengan q = 31.



Gambar 6. Grafik Rata-rata F1 dengan metode KNN dengan k = 2, variasi *random state* 1 sampai dengan 10000 dengan menggunakan definisi jarak *Minkowski* untuk q = 1 sampai dengan q = 31 dan variasi persentase data uji 10 %, 20 %, 30 %, dan 40 % berturut-turut untuk (a), (b), (c), dan (d)

Apabila dilakukan studi simulasi dengan memberikan kode 1 untuk jenis bunga Iris *Setosa* dan kode 2 untuk jenis bunga *Versicolor* dan *Virginica* maka menghasilkan rata-rata F1 bernilai 100 %. Hal itu berarti bahwa dengan klasifikasi seperti itu, sudah jelas membedakan antara kelas dengan kode 1 maupun kelas dengan kode 2. Demikian juga akurasi dalam studi simulasi pada kasus ini memberikan hasil 100 %.

Penelitian tentang klasifikasi dengan metode KNN juga digunakan dalam penelitian (Ali et al., 2020; Bah & Yu, 2021; Pamuji, 2021; Pratama et al., 2021; Rudiyan et al., 2022; Wahyono et al., 2020; Yunus et al., 2021). Dalam penelitian (Wahyono et al., 2020) digunakan data uji 10 %, p = 5 dan k bervariasi dari 3 sampai dengan 9, namun hanya menggunakan satu nilai *random state* yaitu 0. Dalam hal ini juga berbagai macam definisi jarak yaitu *Euclid*, *Manhattan,* dan *Minkowski* namun tidak dijelaskan alasan pemilihan data uji 10 %. Selanjutnya, penelitian tentang klasifikasi dengan metode KNN juga terdapat pada penelitian (Yunus et al., 2021), namun metode yang digunakan dibandingkan dengan metode *random forest*. Berbeda dengan penelitian ini yang tidak membandingkan hasilnya dengan metode lain seperti metode *random forest*. Demikian juga, penelitian lain tentang klasifikasi dengan menggunakan metode KNN dapat ditemukan pada penelitian (Pamuji, 2021). Dalam penelitian tersebut menggunakan variasi k yaitu untuk k = 1 sampai dengan k = 18 dan jarak yang digunakan adalah jarak *Euclid* yang biasa digunakan, namun tidak dilakukan variasi definisi jarak yang digunakan serta hanya menggunakan perhitungan nilai akurasi, presisi, dan *recall* tanpa menggunakan perhitungan F1.



Gambar 7. Grafik rata-rata F1 dengan metode KNN dengan *k* = 5, variasi *random state* 1 sampai dengan 10000 dengan menggunakan definisi jarak *Minkowski* untuk *q* = 1 sampai dengan *q* = 31 dan variasi persentase data uji 10 %, 20 %, 30 %, dan 40 % berturut-turut untuk (a), (b), (c), dan (d)

KESIMPULAN

Dalam penelitian ini telah dijelaskan bagaimana menggunakan metode KNN pada data Iris dan melakukan analisis dengan membuat variasi *k* dalam metode KNN, bilangan random awal yang digunakan, persentase data uji dan definisi jarak yang digunakan. Jika digunakan *k* = 5 dalam metode KNN, variasi random sample 1 sampai dengan 1000 dan digunakan data uji 20 % maka diperoleh akurasi rata-rata 96,6667 %. Selanjutnya juga dilakukan studi simulasi untuk melihat apa yang terjadi jika peubah respons atau klasifikasi diganti. Penelitian ini dapat dikembangkan dengan hanya menggunakan sebagian saja atribut atau peubah bebas yang digunakan dan juga dapat digunakan untuk contoh data klasik yang lain.

DAFTAR PUSTAKA

Ali, A., Alrubei, M., Hassan, L. F. M., Al-Ja'afari, M., & Abdulwahed, S. (2020). Diabetes Classification based on KNN. *IIUM Engineering Journal*, *21*(1), 175–181.

AlZoman, R. M., & Alenazi, M. J. F. (2021). A Comparative Study of Traffic Classification Techniques for Smart City Networks. *Sensors*, *21*(14).

Bah, I., & Yu, X. (2021). KNN Algorithm Used for Heart Attack Detection. *FES Journal of Engineering Sciences*, 7–19.

Baharuddin, M. M., Azis, H., & Hasanuddin, T. (2019). Analisis Performa Metode K-Nearest Neighbor untuk Identifikasi Jenis Kaca. *ILKOM Jurnal Ilmiah*, *11*(3), 269–274.

Dewi, A. M. S. I., & Dwidasmara, I. B. G. (2020). Implementation of the K-Nearest Neighbor (KNN) Algorithm for Classification of Obesity Levels. *JELIKU (Jurnal Elektronik Ilmu Komputer Udayana)*, *9*(2), 277–284.

Djakaria, I., Guritno, S., & Kartiko, S. H. (2010). Visualisasi Data Iris menggunakan Analisis Komponen Utama dan Analisis Komponen Utama Kernel. *Jurnal ILMU DASAR*, *11*(1), 31–38.

Febrianti, F., Hafiyusholeh, M., & Asyhar, A. H. (2016). Perbandingan Pengklusteran Data Iris menggunakan Metode K-Means Dan Fuzzy C-Means. *Jurnal Matematika "MANTIK," 2*(1), 7–13.

Handayani, P. K. (2020). Penerapan Principal Component Analysis untuk Peningkatan Kinerja Algoritma Decision Tree pada Iris Dataset. *Indonesian Journal of Technology, Informatics and Science (IJTIS)*, *1*(2), 55–58.

Haviluddin, Iqbal, M., Putra, G. M., Puspitasari, N., Setyadi, H. J., Dwiyanto, F. A., Wibawa, A. P., & Alfred, R. (2020). A Performance Comparison of Euclidean, Manhattan and Minkowski Distances in K-Means Clustering. *2020 6th International Conference on Science in Information Technology (ICSITech)*, 184–188.

Hermanto, D. M. C. (2017). Analisis Algoritma Clustering dalam Kasus Penentuan Jenis Bunga Iris. *Jurnal Media Aplikom*, *9*(2), 72–84.

Hidayati, N., & Hermawan, A. (2021). K-Nearest Neighbor (K-NN) Algorithm with Euclidean and Manhattan in Classification of Student Graduation. *Journal of Engineering and Applied Technology*, *2*(2), 86–91.

Khamdani, M. K., Hidayat, N., & Dewi, R. K. (2021). Implementasi Metode K-Nearest Neighbor untuk Mendiagnosis Penyakit Tanaman Bawang Merah. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *5*(1), 11–16.

Mara, A. A. P. T., Sediyono, E., & Purnomo, H. (2021). Penerapan Algoritma K-Nearest Neighbors pada Analisis Sentimen Metode Pembelajaran Dalam Jaringan (DARING) di Universitas Kristen Wira Wacana Sumba. *JOINTER - Journal of Informatics Engineering*, *1*(2), 24–31.

Nugroho, W. (2020). Optimasi Metode K-Nearest Neighbours dengan Backward Elimination Menggunakan Dataset Software Effort Estimation. *Bianglala Informatika*, *8*(2), 129–133.

Pamuji, A. (2021). Performance of the K-Nearest Neighbors Method on Analysis of Social Media Sentiment. *JUISI*, *7*(1), 32–37.

Pamungkas, F. S., & Kharisudin, I. (2021). Analisis Sentimen dengan SVM, NAIVE BAYES dan KNN untuk Studi Tanggapan Masyarakat Indonesia Terhadap Pandemi Covid-19 pada Media Sosial Twitter. *PRISMA, Prosiding Seminar Nasional Matematika*, *4*, 628–634.

Prakisya, N. P. T., Liantoni, F., Hatta, P., Aristyagama, Y. H., & Setiawan, A. (2021). Utilization of K-Nearest Neighbor Algorithm for Classification of White Blood Cells in AML M4, M5, and M7. *Open Engineering*, *11*, 662–668.

Pratama, N. S. H., Afandi, D. T., Mulyawan, Iin, & Nuris, N. D. (2021). Menurunkan Presentase Kredit Macet Nasabah dengan menggunakan Algoritma K-Nearest Neighbor. *Information System for Educators and Professionals*, *5*(2), 131–140.

Purwaningsih, E., & Nurelasari, E. (2021). Penerapan K-Nearest Neighbor Untuk Klasifikasi Tingkat Kelulusan Pada Siswa. *Syntax: Jurnal Informatika*, *10*(1), 46–55.

Retnoningsih, E., & Pramudita, R. (2020). Mengenal Machine Learning dengan Teknik Supervised dan Unsupervised Learning Menggunakan Python. *Bina Insani Ict Journal*, *7*(2), 156–165.

Rudiyan, A., Dzulkifli, A. E., & Munazar, K. (2022). Klasifikasi Kebakaran Hutan Menggunakan Metode K-Nearest Neighbor: Studi Kasus Hutan Provinsi Kalimantan Barat. *JTIM: Jurnal Teknologi Informasi Dan Multimedia*, *3*(4), 195–202.

Wahyono, Trisna, I. N. P., Sariwening, S. L., Fajar, M., & Wijayanto, D. (2020). Perbandingan Penghitungan Jarak pada K-Nearest Neighbour dalam Klasifikasi Data Tekstual. *Jurnal Teknologi Dan Sistem Komputer*, *8*(1), 54–58.

Yunus, R., Ulfa, U., & Safitri, M. D. (2021). Application of the K-Nearest Neighbors (K-NN) Algorithm for Classification of Heart Failure. *Journal of Applied Intelligent System*, *6*(1), 1–9.