Metode Pembobotan Jarak dengan Koefisien Variasi untuk Mengatasi Kelemahan *Euclidean Distance* pada Algoritma *k-Nearest Neighbor*

Agustiyar[1*)], Romi Satria Wahono [2)], Catur Supriyanto[3)]
[1)] LLDIKTI Wilayah VI Semarang
[2, 3)] Teknik Informatika, Universitas Dian Nuswantoro Semarang
[1)] *agustiyar001@gmail.com*, [2)] *romi@romisatriawahono.net*, [3)]*caturs@research.dinus.ac.id*

ABSTRACT

k-Nearest Neighbor (k-NN) is one of the classification algorithms which becomes top 10 in data mining. k-NN is simple and easy to apply. However, the classification results are greatly influenced by the scale of the data input. All of its attributes are considered equally important by Euclidean distance, but inappropriate with the relevance of each attribute. Thus, it makes classification results decreased. Some of the attributes are more or less relevance or, in fact, irrelevant in determining the classification results. To overcome the disadvantage of k-NN, Zolghadri, Parvinnia, and John proposed Weighted Distance Nearest Neighbor (WDNN) having the performance better than k-NN. However, when the result is k >1, the performance decrease. Gou proposed Dual Distance Weighted Voting k-Nearest Neighbor (DWKNN) having the performance better than k-NN. However, DWKNN focused in determining label of classification result by weighted voting. It applied Euclidean distance without attribute weighting. This might cause all attribute considered equally important by Euclidean distance, but inappropriate with the relevance of each attribute, which make classification results decreased. This research proposed Coefficient of Variation Weighting k-Nearest Neighbor (CVWKNN) integrating with MinMax normalization and weighted Euclidean distance. Seven public datasets from UCI Machine Learning Repository were used in this research. The results of Friedman test and Nemenyi post hoc test for accuracy showed CVWKNN had better performance and significantly different compared to k-NN algorithm.

Keywords*: k-NN, attribute weighting, weighted Euclidean distance, MinMax normalization*

I. PENDAHULUAN

k-NN pertama kali diperkenalkan pada awal tahun 1950an (Jiawei et al. 2012). *k*-NN adalah sebuah algoritma klasifikasi yang mengklasifikasikan data baru berdasarkan kedekatannya (Han and Kamber 2011). Cara kerja *k*-NN dengan mencari jarak terdekat data uji terhadap *k* tetangga terdekat data latih (Christian Gratia Nugroho, Didik Nugroho 2015) (Raymundus, YS, and Siswanti 2013). Kedekatan antar data diukur dengan fungsi jarak, fungsi jarak yang paling sering digunakan adalah *Euclidean distance* (Schenatto et al. 2017) (Akhil, Deekshatulu, and Chandra 2013). *k*-NN memiliki kelebihan, yaitu sederhana dan hasilnya mudah dijelaskan (Hocke and Martinetz 2015), mudah dimengerti untuk klasifikasi dan regresi (Song et al. 2017). *k*-NN menjadi salah satu metode terpopuler dan sederhana (Ke 2018), paling sukses pada teknik klasifikasi pola (Jiao, Pan, and Feng 2015). Menurut Neo dan Venture (Neo and Ventura 2012) *k*-NN efektif dalam pengenalan pola, memiliki konsep sederhana sehingga mudah diaplikasikan.

k-NN merupakan metode tertua dan paling sederhana dalam klasifikasi pola (Zolghadri, Parvinnia, and John 2009), akan tetapi *k*-NN memiliki kelemahan, yaitu hasil klasifikasi sangat dipengaruhi skala *input* data dan fungsi pengukuran jarak *Euclidean* memperlakukan atribut data secara sama, tidak sesuai dengan relevansi masing-masing atribut data yang berakibat pada menurunnya hasil klasifikasi (Zolghadri, Parvinnia, and John 2009). Kelemahan utama *k*-NN adalah hasilnya sangat dipengaruhi skala *input* data dan pengukuran atribut (Hocke and Martinetz 2015). *k*-NN memiliki dua kekurangan, 1) bergantung pada skala awal atribut dari data, 2) fungsi jarak *Euclidean* memperlakukan atribut data secara sama, kondisi ini tidak sesuai dengan kondisi data sebenarnya. Fungsi jarak *Euclidean k*-NN

mengukur jarak tidak sesuai dengan relevansi masing-masing atribut data (Hocke and Martinetz 2013). Menurut Schenatto *et al.* (Schenatto et al. 2017) pengukuran kesamaan menggunakan fungsi jarak *Euclidean* sensitif terhadap rentang variabel masukan, umumnya variabel seperti ini dilakukan normalisasi.

Normalisasi merupakan tahap *pre-processing* yang sangat berguna untuk algoritma klasifikasi dengan cara mencegah perbedaan rentang yang besar antar atribut (Jiawei et al. 2012). Ada beberapa metode normalisasi yang dapat digunakan untuk normalisasi data (Jain, Nandakumar, and Ross 2005), salah satunya adalah metode *MinMax*. Metode *MinMax* merupakan metode normalisasi yang paling sederhana, di mana data baru diperoleh dengan cara data lama dikurangi data terkecil dan dibagi dengan data terbesar dikurangi data terkecil. Penerapan normalisasi data *MinMax* dapat meningkatkan kinerja algoritma Neural Network (NN) (Nawi, Atomi, and Rehman 2013), meningkatkan kinerja algoritma Artificial Neural Network (ANN) (Shabani et al. 2017) pada prediksi area daun.

Data hasil normalisasi kemudian diklasifikasi menggunakan metode k-NN dengan *Euclidean distance* sebagai fungsi pengukuran jaraknya. Menurut (Yunlong 2016) salah satu kelemahan *Euclidean distance* adalah memperlakukan atribut data secara sama, kondisi ini tidak sesuai dengan kondisi data sebenarnya. Menurut Siminski (Siminski 2017) setiap atribut belum tentu memiliki kontribusi yang sama terhadap kinerja algoritma, ada yang berkontribusi atau bahkan tidak memiliki kontribusi, dengan pembobotan atribut kinerja algoritma dapat ditingkatkan. Menurut Jiang *et al.* (Jiang et al. 2007) salah satu solusi untuk mengatasi permasalahan pada *Euclidean distance* ketika menghitung jarak antar data adalah dengan pembobotan atribut.

Pembobotan atribut dapat meningkatkan kinerja algoritma (Dialameh and Jahromi 2016) (Wu et al. 2015). Menurut Witten, Frank dan Hall (Witten, Frank, and Hall 2011) salah satu untuk meningkatkan kinerja *k*-NN adalah dengan pembobotan atribut. Ada beberapa metode pembobotan yang dapat digunakan untuk pembobotan atribut diantaranya adalah *Information Gain*, dan *Gain Ratio* (Da et al. 2012), Principal Component Analysis (PCA) (Cao and Liu 2010). Metode pembobotan atribut yang lain diantaranya *coefficient of variation* (CV) (Bechar and Vitner 2009), di mana nilai CV diperoleh dengan cara membagi nilai standard deviasi dengan nilai rata-rata atribut. Ren dan Fan (Ren and Fan 2011) berhasil meningkatkan kinerja algoritma *k*-means dengan metode pembobotan *coefficient of variation.*

Permasalahan pengukuran jarak pada algoritma *k*-NN telah menarik peneliti untuk melakukan perbaikan, diantaranya dengan melakukan pembobotan atribut, Large Margin Nearest Neighbor (LMNN) (Weinberger and Saul 2009), Weighted Distance Nearest Neighbor (WDNN) (Zolghadri, Parvinnia, and John 2009). Pada tahun 2011 Gou, Luo, dan Xiong mengusulkan Dual Distance Weighted Voting for *k*-Nearest Neighbor (DWKNN) (Gou, Luo, and Xiong 2011), hasil uji coba pada 10 dataset UCI *Machine Learning Repository* menunjukkan kinerja DWKNN lebih baik dari *k*-NN dan Distance Weighted *k*-Nearest Neighbor Rule (WKNN) (Dudani 1976). Akan tetapi menurut Mateos García, García Gutiérrez, dan Riquelme Santos (Mateos-García, García-Gutiérrez, and Riquelme-Santos 2017) DWKNN hanya fokus pada penentuan label dengan *weight voting*, tidak menggabungkan *weight voting* dengan seleksi atribut.

Dari uraian di atas dapat diperoleh informasi bahwa *k*-NN memiliki kelemahan sensitif terhadap skala *input* data, hal tersebut dapat diatasi dengan normalisasi data. Kelemahan lain *k*-NN terletak pada fungsi pengukuran jarak *Euclidean distance* yang memperlakukan atribut data secara sama tidak sesuai dengan relevansi masing-masing atribut data yang berakibat pada menurunnya hasil klasifikasi, kelemahan ini dapat diatasi dengan menambahkan bobot pada *Euclidean distance* yang dikenal dengan *weighted Euclidean distance.*

 DOI : https://doi.org/10.30646/sinus.v20i1.565

Penelitian ini mengusulkan *Coefficient of Variation Weighting k-Nearest Neighbor* (CVWKNN) atau Algoritma *k*-Nearest Neighbor dengan Metode Pembobotan Koefisien Variasi, yaitu mengintegrasikan normalisasi *MinMax* untuk mengatasi kelemahan *k*-NN yang sensitif terhadap skala *input* data dengan *weighted Euclidean distance* untuk mengatasi *Euclidean distance* yang memperlakukan atribut data secara sama, di mana bobot atribut diperoleh dengan *coefficient of variation* (CV). Normalisasi *MinMax* dipilih karena sederhana, dapat mencegah rentang yang besar antar atribut, dan terbukti mampu meningkatkan kinerja algoritma Neural Network (NN) (Nawi, Atomi, and Rehman 2013), Artificial Neural Network (ANN) (Shabani et al. 2017). *Weighted Euclidean distance* dipilih karena penambahan bobot pada *Euclidean distance* mampu mengatasi *Euclidean distance* yang memperlakukan atribut data secara sama, dan sesuai dengan Jiang *et al.* (Jiang et al. 2007) bahwa pembobotan mampu mengatasi permasalahan pada *Euclidean distance*.

Paper ini disusun sebagai berikut, bagian 2 menjelaskan tinjauan pustaka. Pada bagian 3, menjelaskan metode penelitian. Hasil dan pembahasan disajikan pada bagian 4. Kesimpulan dan saran disajikan pada bagian akhir.

II. TINJAUAN PUSTAKA

Permasalahan pengukuran jarak pada algoritma *k*-NN telah menarik peneliti untuk mengusulkan metode perbaikan. Pada tahun 2009 Zolghadri *et al*. mengusulkan Weighted Distance Nearest Neighbor (WDNN) (Zolghadri, Parvinnia, and John 2009). WDNN melakukan pembobotan pada data pelatihan, bobot diperoleh dengan menggunakan metode *leave-one-out* (LV1). *Euclidean distance* dimodifikasi mejadi *weighted Euclidean distance* untuk menghitung jarak data uji dengan data latih. Hasil uji coba pada 14 dataset publik menunjukkan kinerja rata-rata WDNN mengungguli Adaptive Nearest Neighbor (ANN) (Wang, Neskovic, and Cooper 2007) dan *k*-NN.

Pada tahun 2010 Yang, Cao, dan Zhang mengusulkan Weighted Distance *k*-Nearest Neighbor (WDKNN). Ini adalah pengembangan dari WDNN (Zolghadri, Parvinnia, and John 2009), jika WDNN memiliki kinerja bagus dengan nilai k=1 akan tetapi jika nilai $k > 1$ kinerjanya menurun, maka pada WDKNN memiliki kinerja yang bagus meskipun nilai $k \geq 1$. Hasil uji coba pada 20 dataset publik menunjukkan kinerja rata-rata WDKNN mengungguli WDNN dan *k*-NN.

Pada tahun 2011 Hu *et al.* (Hu et al. 2011) mengusulkan Sample Weight Learning via Minimizing Loss of Margin (SWL-MLM). Optimalisasi bobot data dilakukan dengan algoritma Gradient Decent. Untuk menghitung jarak data uji dengan data latih digunakan *weighted Euclidean distance*. Hasil uji coba pada 10 dataset publik menunjukkan kinerja rata-rata SWL-MLM lebih unggul dari Relief dan *k*-NN.

Pada tahun 2011 Gou, Luo, dan Xiong mengusulkan Dual Distance Weighted Voting for *k*-Nearest Neighbor (DWKNN). Metode ini merupakan pengembangan dari metode Distance Weighted *k*-NN (WKNN) (Dudani 1976). Meskipun hasil uji coba pada 10 dataset publik menunjukkan kinerja DWKNN lebih baik dari WKNN (Dudani 1976) dan *k*-NN akan tetapi DWKNN hanya fokus pada penentuan label dengan *weight voting*, tidak menggabungkan *weight voting* dengan seleksi atribut.

Penelitian ini mengusulkan integrasi normalisasi *MinMax* untuk mengatasi kelemahan *k*-NN yang sensitif terhadap skala *input* data dengan *weighted Euclidean distance* untuk mengatasi *Euclidean distance* yang memperlakukan atribut data secara sama, di mana bobot atribut diperoleh dengan *coefficient of variation* (CV).

III. METODE PENELITIAN

Penelitian ini mengusulkan *Coefficient of Variation Weighting k-Nearest Neighbor* (CVWKNN), sebuah metode untuk mengatasi permasalahan pengukuran jarak pada

algoritma *k*-NN dengan mengintegrasikan normalisasi *MinMax* dan *weighted Euclidean distance*. Kelemahan *k*-NN yang sensitif terhadap skala *input* data diatasi dengan normalisasi *MinMax*, sedangkan kelemahan *Euclidean distance k*-NN yang memperlakukan atribut data secara sama diatasi dengan *weighted Euclidean distance*.

Pada Gambar 1 ditampilkan alur kerja metode CVWKNN, sebagai berikut:

1. Memasukkan dataset atau himpunan data (x_{ia}, y_i) di mana *i* adalah urutan data (1, 2, …, *m*), *a* adalah urutan atribut (1, 2, …, *n*), dan *y* adalah label;
2. Menentukan nilai *k* tetangga terdekat;
3. Menghitung bobot atribut dengan *coefficient of variation* (CV);
4. Melakukan proses pengolahan data awal menggunakan normalisasi *MinMax*;
5. Menghitung jarak data uji dengan data latih dengan *weighted Euclidean distance*;
6. Mengurutkan hasil perhitungan jarak data uji dengan data latih dari urutan kecil ke besar;
7. Mengambil hasil perhitungan jarak yang sudah urut sesuai nilai *k*;
8. Menentukan hasil klasifikasi berdasarkan label mayoritas;
9. Menghitung akurasi hasil klasifikasi.

Secara umum metode CVWKNN sama dengan metode *k*-NN klasik, yang membedakan antara metode keduanya terletak pada tahap ke-3, ke-4, dan ke-5. Pada metode CVWKNN, tahap ke-3 terdapat proses penghitungan bobot atribut dengan *coefficient of variation* (CV). Untuk menghitung bobot atribut ke-*a* (w_a) digunakan Fungsi (1), di mana *a*: atribut, *n*: jumlah atribut, cv_a: *coefficient of variation* atribut ke-*a*.

$$w_a = \frac{cv_a}{\sum_{a=1}^{n} cv_a} \quad \text{.....(1)}$$

Untuk menghitung *coefficient of variation* atribut ke-*a* (cv_a) digunakan Fungsi (2), di mana δ_a: standar deviasi atribut ke-*a*, μ_a: nilai rata-rata atribut ke-*a*.

$$cv_a = \frac{\delta_a}{\mu_a} \quad \text{.....(2)}$$

Untuk menghitung standar deviasi atribut ke-*a* (δ_a) digunakan Fungsi (3) di mana *a*: atribut, *m*: jumlah data, x_{ia}: data ke-*i* atribut ke-*a*, μ_a: nilai rata-rata atribut ke-*a*.

$$\delta_a = \sqrt{\frac{1}{m-1}\sum_{i=1}^{m}(x_{ia} - \mu_a)^2} \quad \text{.....(3)}$$

Untuk menghitung nilai rata-rata atribut ke-a (μ_a) digunakan Fungsi (4) di mana a: atribut, *m*: jumlah data, x_{ia}: data ke-*i* atribut ke-*a*.

$$\mu_a = \frac{1}{m}\sum_{i=1}^{m} x_{ia} \quad \text{.....(4)}$$

Pada metode CVWKNN, tahap ke-4 ditambahkan proses pengolahan data awal berupa normalisasi *MinMax* yang bertujuan menstandarkan nilai atribut pada skala tertentu sebelum dilakukan proses klasifikasi. Normalisasi *MinMax* telah digunakan oleh beberapa peneliti (Jain, Nandakumar, and Ross 2005) (Nawi, Atomi, and Rehman 2013) (Tzortzis and Likas 2014) (Venugopal and Sundaram 2017) untuk mengatasi permasalahan skala nilai atribut. Fungsi Normalisasi *MinMax* (5), di mana *z*: nilai atribut hasil normalisasi, *x*: nilai atribut, min (*x)*: nilai minimal atribut, max (*x)*: nilai maksimal atribut.

$$z = \frac{x - min(x)}{max\ (x) - min(x)} \quad \text{.....(5)}$$

Pada metode CVWKNN, tahap ke-5 hitung jarak data uji dengan data latih menggunakan *weighted Euclidean distance*. Jika pada *k*-NN klasik semua atribut dianggap sama, jarak antara data uji dengan data pelatihan dihitung dengan *Euclidean distance* (6)

tanpa adanya pembobotan, di mana *d*: distance, x': data uji, *x*: data latih, *a*: atribut, *n*: jumlah atribut.

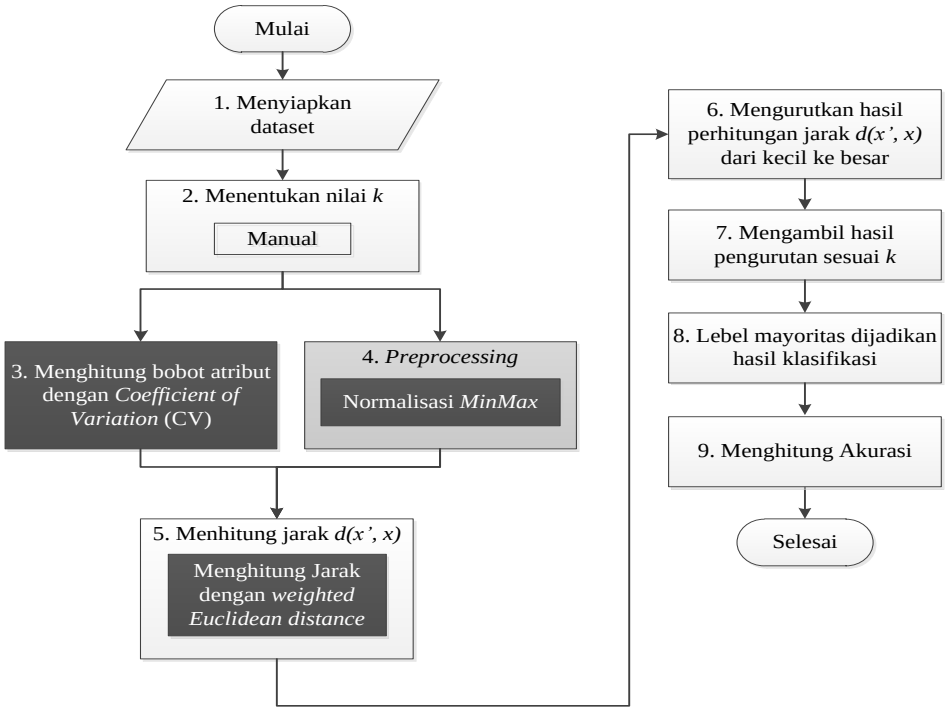
$$d(x', x) = \sqrt{\sum_{a=1}^{n} (x'_a - x_a)^2} \quad \text{.....(6)}$$

Maka pada metode CVWKNN, untuk mengukur jarak antara data uji dengan data pelatihan dihitung dengan *weighted Euclidean distance* (7), di mana *w* adalah bobot yang diperoleh dengan fungsi (1).

$$d(x', x) = \sqrt{\sum_{a=1}^{n} w^2 (x'_a - x_a)^2} \quad \text{.....(7)}$$

Label hasil klasifikasi diperoleh dari label mayoritas tetangga terdekatnya, sebagaimana Fungsi (8), di mana y': label hasil klasifikasi, *y*: label, y_i: label ke-*i* dari *k* tetangga terdekatnya, *k*: jumlah tetangga terdekat.

$$y' = arg\, max_y \sum_{i=1}^{k} y = y_i \quad \text{.....(8)}$$



Gambar 1 Alur Kerja Metode CVWKNN

IV. HASIL DAN PEMBAHASAN

Eksperimen dilakukan menggunakan komputer degan spesifikasi Intel Core i7-7500U 2,70 GHz, 8 GB RAM, sistem operasi Windows 10 Home 64-bit, serta aplikasi Rapid Miner Versi 7.5.

Tabel 1 Dataset yang Digunakan

No	Nama Dataset	Jumlah Data	Jumlah	Jumlah Label
1	*Wine*	178	13	3
2	*Iris*	150	4	3
3	*Balance Scale*	625	4	3
4	*Heart*	270	13	2
5	*Breast Cancer*	699	10	2
6	*Ecoli*	336	8	7
7	*Dermatology*	366	34	6

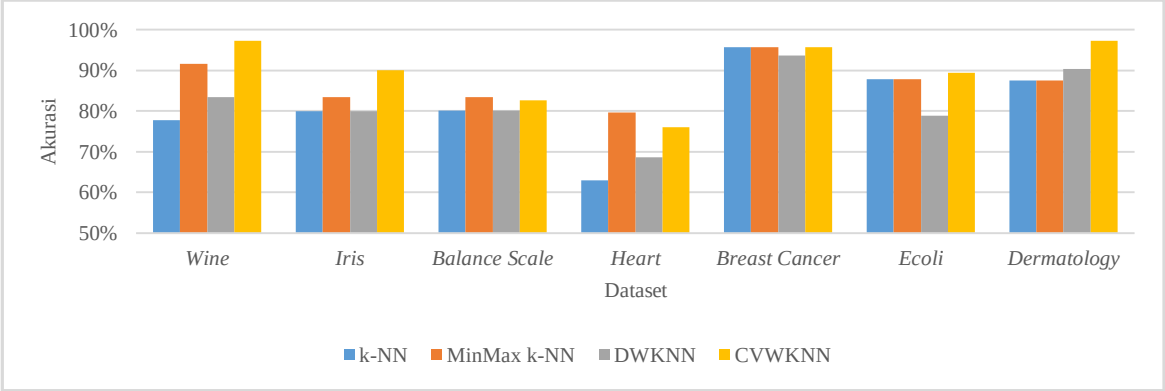
Pada Tabel 1 ditampilkan 7 dataset dari UCI Machine Learning Repository yang digunakan dalam eksperimen ini. Semua dataset bertipe numerik dengan jumlah data mulai dari 150 sampai dengan 699, jumlah atribut mulai dari 4 sampai dengan 34, serta label bertipe *binominal* dan *polynominal*.

Split data digunakan untuk membagi dataset menjadi data uji dan data pelatihan dengan perbandingan 20:80. Evaluasi hasil eksperimen menggunakan akurasi. Akurasi menggambarkan persentase data uji yang diklasifikasikan benar oleh metode klasifikasi (Jiawei et al. 2012). *Friedman test* dan *Nemenyi post hoc test* digunakan untuk mengetahui perbedaan yang signifikan antar metode yang diperbandingkan. Menurut Demsar (Demsar 2006), *Friedman test* sesuai digunakan untuk perbandingan pada lebih dari dua metode pada beberapa dataset.

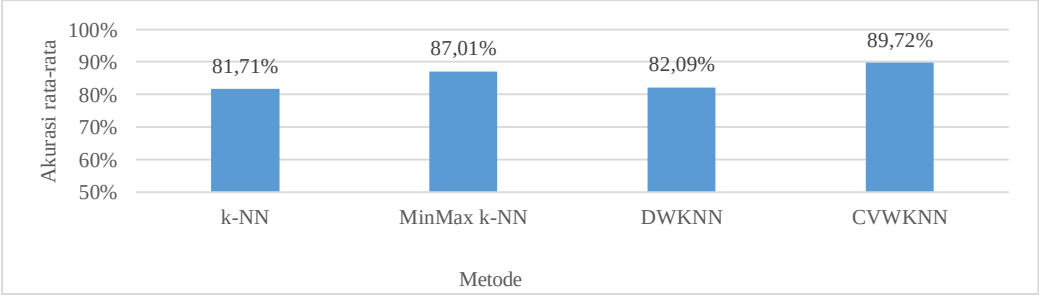
Untuk mengetahui kinerja metode usulan, CVWKNN dibandingkan dengan *k*-NN, *MinMax k*-NN, dan DWKNN. Seluruh metode yang diperbandingkan diuji dengan nilai *k*=5, kemudian dapat dilihat metode mana yang memiliki kinerja terbaik. Pada Tabel 2, Gambar 2, dan Gambar 3 ditampilkan perbandingan akurasi metode yang diperbandingkan, dan dapat dilihat bahwa CVWKNN memiliki akurasi rata-rata tertinggi.

Tabel 2 Perbandingan Akurasi pada Tujuh Dataset dari Metode yang Diperbandingkan

Dataset	Akurasi *k*-NN	Akurasi *MinMax k*-NN	Akurasi DWKNN	Akurasi CVWKNN
Wine	77.78%	91.67%	83.33%	**97.22%**
Iris	80.00%	83.33%	80.00%	**90.00%**
Balance Scale	80.16%	**83.33%**	80.16%	82.54%
Heart	62.96%	**79.63%**	68.52%	75.93%
Breast Cancer	**95.71%**	**95.71%**	93.57%	**95.71%**
Ecoli	87.88%	87.88%	78.79%	**89.39%**
Dermatology	87.50%	87.50%	90.28%	**97.22%**
Akurasi rata-rata	81.71%	87.01%	82.09%	**89.72%**



Gambar 2 Grafik Perbandingan Akurasi pada Tujuh Dataset dari Metode yang Diperbandingkan



Gambar 3 Grafik Perbandingan Akurasi Rata-Rata pada Tujuh Dataset dari Metode yang Diperbandingkan

 DOI : https://doi.org/10.30646/sinus.v20i1.565

Pada tahap akhir, untuk mengetahui perbedaan yang signifikan antar metode yang diperbandingkan dilakukan *Friedman test* dan *Nemenyi post hoc test*. *Friedman test* membandingkan peringkat rata-rata kinerja atau *average ranked* (R) metode pada setiap dataset (Wahono, Herman, and Ahmad 2014). Pada *Friedman test* taraf pengujian (α) ditentukan sebesar 5% atau 0.05. Hipotesis, H0: tidak terdapat perbedaan nilai rata-rata antar metode yang diperbandingkan, Ha: terdapat perbedaan nilai rata-rata antar metode yang diperbandingkan. Daerah penolakan, tolak H0 jika *P-value* lebih kecil daripada α (alpha) dan Ha diterima. Terima H0 jika *P-value* lebih besar daripada α (alpha) dan Ha ditolak. Setelah *Friedman test* dilakukan jika hasilnya H0 ditolak dan Ha diterima, selanjutnya dilakukan *Nemenyi post hoc test*.

Nemenyi post hoc test digunakan untuk membandingkan satu sama lain diantara semua metode klasifikasi yang diperbandingkan. Kinerja dua metode klasifikasi dikatakan berbeda signifikan jika hasil perbandingan peringkat rata-rata kinerja lebih besar dari nilai *Critical Different* (CD). Fungsi yang digunakan ditampilkan pada Fungsi (9), di mana CD: *Critical Different*, q_a: *student sized range statisti*c, *K:* jumlah metode yang diperbandingkan, *D*: jumlah dataset. Pada Tabel 3 ditampilkan nilai q_a pada α (*alpha*) 0.05 (Demsar 2006). Jadi nilai q_a sesuai jumlah metode yang diperbandingkan.

$$CD = q_a \sqrt{\frac{K(K+1)}{6D}} \quad \text{.....(9)}$$

Tabel 3 Nilai q_a Pada α= 0.05

Metode	2	3	4	5	6	7	8	9	10
q_a.05	1.960	2.343	2.569	2.728	2.850	2.949	3.031	3.102	3.164

Pada

Tabel 4 ditampilkan hasil *Friedman test* akurasi metode yang diperbandingkan, di mana nilai ***p*= 0.01** lebih kecil dari *alpha* (α=0.05) maka H0 ditolak dan Ha diterima, sehingga dapat disimpulkan terdapat perbedaan yang signifikan antar metode yang diperbandingkan. Pada

Tabel 5 ditampilkan perbandingan berpasangan *Nemenyi post hoc test* akurasi metode yang diperbandingkan dibandingkan dengan nilai CD. Metode CVWKNN dibandingkan dengan *k*-NN memperoleh nilai **1.86** dan dibandingkan dengan DWKNN memperoleh nilai **1.86** lebih besar dari nilai *Critical Different* (CD) = **1.77**, sedangkan dibandingkan dengan *MinMax k*-NN memperoleh nilai 0.57 lebih kecil dari nilai *Critical Different* (CD) = **1.77**. Pada Tabel 6 ditampilkan perbedaan signifikan pada *Nemenyi post hoc test* dari akurasi metode yang diperbandingkan. Dilihat dari akurasi, CVWKNN memiliki perbedaan kinerja yang signifikan jika dibandingkan *k*-NN dan DWKNN, sedangkan dibandingkan dengan *MinMax k*-NN, CVWKNN tidak berbeda secara signifikan.

Tabel 4 Hasil *Friedman Test* Akurasi Metode yang Diperbandingkan

Q (*Observed value*)	12.48
Q (*Critical value*)	7.81
DF	3
p-value (*Two-tailed*)	0.01
alpha	0.05

Tabel 5 Perbandingan Berpasangan *Nemenyi Post Hoc Test* dari Akurasi Dibandingkan dengan *Critical Different*

Metode	*k*-NN	*MinMax k*-NN	DWKNN	CVWKNN
k-NN	0.00	-1.29	0.00	**-1.86**
MinMax k-NN	1.29	0.00	1.29	-0.57
DWKNN	0.00	-1.29	0.00	**-1.86**
CVWKNN	**1.86**	0.57	**1.86**	0.00
Critical Different	**1.77**			

Tabel 6 Perbedaan Signifikan pada Nemenyi Post Hoc Test dari Akurasi Metode yang Diperbandingkan

Metode	*k*-NN	*MinMax k*-NN	DWKNN	CVWKNN
k-NN	No	No	No	**Yes**
MinMax k-NN	No	No	No	No
DWKNN	No	No	No	**Yes**
CVWKNN	**Yes**	No	**Yes**	No

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

CVWKNN, integrasi normalisasi *MinMax* dan *weighted Euclidean distance* diusulkan untuk meningkatkan kinerja algoritma *k*-NN. Normalisasi *MinMax* dimanfaatkan untuk mengatasi skala *input* data, sedangkan *weighted Euclidean distance* digunakan untuk mengatasi *Euclidean distance* yang memperlakukan atribut data secara sama, tidak sesuai dengan relevansi masing-masing atribut data. CVWKNN diujicobakan pada 7 dataset publik dari UCI *Machine Learning Repository*. Hasil uji coba menunjukkan kinerja metode usulan lebih baik dari *k*-NN. Kesimpulan penelitian ini adalah metode usulan mampu mengatasi kelemahan pada algoritma *k*-NN, sehingga kinerja algoritma *k*-NN meningkat.

5.2 Saran

Saran untuk penelitian selanjutnya Pada penelitian ini nilai *k* ditentukan secara manual, pada penelitian selanjutnya nilai *k* dapat ditentukan secara otomatis. Pada penelitian ini untuk menghitung bobot atribut mengagunkan coefficient *of variation* (CV), pada penelitian selanjutnya dapat digunakan metode pembobotan yang lain seperti *deep feature weightin*g untuk meningkatkan kinerja algoritma k-NN.

DAFTAR PUSTAKA

Akhil, M, B L Deekshatulu, and Priti Chandra. 2013. "Classification of Heart Disease Using K- Nearest Neighbor and Genetic Algorithm." In *Procedia Technology*, 10:85–94. Kalyani, Nadia, West Bengal, September 27-28: Elsevier B.V.

Bechar, Avital, and Gad Vitner. 2009. "A Weight Coefficient of Variation Based Mathematical Model to Support the Production of ' Packages Labelled by Count ' in

Agriculture." *Biosystems Engineering* 104 (3). IAgrE: 362–68. doi:10.1016/j.biosystemseng.2009.08.003.

Cao, Qinghua, and Yu Liu. 2010. "A KNN Classifier with PSO Feature Weight Learning Ensemble." In *International Conference on Intelligent Control and Information Processing*, 110–14. Dalian, Aug 13-15.

Christian Gratia Nugroho, Didik Nugroho, Sri Hariyati Fitriasih. 2015. "Sistem Pendukung Keputusan Untuk Pemilihan Metode Kontrasepsi PADA Pasangan Usia Subur Dengan Algoritna K-Nearset Neighbor (K-KN)." *Jurnal Ilmiah SINUS* 13 (1): 21–30. doi:10.2473/shigentosozai1953.83.947_421.

Da, H, K E Say, I Yenido, S Albayrak, and C Acar. 2012. "Comparison of Feature Selection Algorithms for Medical Data." In *2012 International Symposium on Innovations in Intelligent Systems and Applications*. Trabzon, July 2-4.

Demsar, Janez. 2006. "Statistical Comparisons of Classifiers over Multiple Data Sets." *Journal of Machine Learning Research 7* 7: 1–30.

Dialameh, Maryam, and Mansoor Zolghadri Jahromi. 2016. "Proposing a General Feature Weighting Function." *Expert Systems With Applications*. Elsevier Ltd. doi:10.1016/j.eswa.2016.12.016.

Dudani, Sahibsingh A. 1976. "The Distance-Weighted K-Nearest-Neighbor Rule." *IEEE Transactions on Systems, Man and Cybernetics* SMC-6 (4): 325–27. doi:10.1109/TSMC.1976.5408784.

Gou, Jianping, Mingying Luo, and Taisong Xiong. 2011. "Improving K-Nearest Neighbor Rule with Dual Weighted Voting for Pattern Classification." *Communications in Computer and Information Science* 159 CCIS (PART 2): 118–23. doi:10.1007/978-3-642-22691-5_21.

Han, Jiawei, and Micheline Kamber. 2011. *Data Mining: Concepts and Techniques*. *Elsevier*. Second Edi. Vol. 12. Elsevier Inc. doi:10.1007/978-3-642-19721-5.

Hocke, Jens, and Thomas Martinetz. 2013. "Feature Weighting by Maximum Distance Minimization." In *International Conference on Artificial Neural Networks*, 420–25. Bulgaria, September 10-13.

———. 2015. "Maximum Distance Minimization for Feature Weighting." *Pattern Recognition Letters* 52. Elsevier Ltd.: 48–52. doi:10.1016/j.patrec.2014.10.003.

Hu, Qinghua, Pengfei Zhu, Yongbin Yang, and Daren Yu. 2011. "Large-Margin Nearest Neighbor Classifiers via Sample Weight Learning." *Neurocomputing* 74 (4). Elsevier: 656–60. doi:10.1016/j.neucom.2010.09.006.

Jain, Anil, Karthik Nandakumar, and Arun Ross. 2005. "Score Normalization in Multimodal Biometric Systems." *Pattern Recognition* 38: 2270–85. doi:10.1016/j.patcog.2005.01.012.

Jiang, Liangxiao, Zhihua Cai, Dianhong Wang, and Siwei Jiang. 2007. "Survey of Improving K-Nearest-Neighbor for Classification." In *Proceedings - Fourth International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2007*, 1:679–83. Haikou, Aug 24-27. doi:10.1109/FSKD.2007.552.

Jiao, Lianmeng, Quan Pan, and Xiaoxue Feng. 2015. "Multi-Hypothesis Nearest-Neighbor Classifier Based on Class-Conditional Weighted Distance Metric." *Neurocomputing* 151 (P3). Elsevier: 1468–76. doi:10.1016/j.neucom.2014.10.039.

Jiawei, H, Micheline Kamber, Jiawei Han, Micheline Kamber, and Jian Pei. 2012. *Data Mining: Concepts and Techniques*. *San Francisco, CA, Itd: Morgan Kaufmann*. Third Edit. Elsevier Inc. doi:10.1016/B978-0-12-381479-1.00001-0.

Ke, N. 2018. "Region Based Segmentation of Social Images Usinng Soft KNN Algorithhm." In *Procedia Computer Science*, 125:93–98. Kurukshetra, December 7-8: Elsevier B.V. doi:10.1016/j.procs.2017.12.014.

Mateos-García, Daniel, Jorge García-Gutiérrez, and José C. Riquelme-Santos. 2017. "On the Evolutionary Weighting of Neighbours and Features in the K-Nearest Neighbour Rule." *Neurocomputing* 0: 1–7. doi:10.1016/j.neucom.2016.08.159.

Nawi, Nazri Mohd, Walid Hasen Atomi, and M Z Rehman. 2013. "The Effect of Data Pre-Processing on Optimized Training of Artificial Neural Networks." In *Procedia Technology*, 11:32–39. Selangor, Jun 24-25: Elsevier B.V. doi:10.1016/j.protcy.2013.12.159.

Neo, Toh Koon Charlie, and Dan Ventura. 2012. "A Direct Boosting Algorithm for the K-Nearest Neighbor Classifier via Local Warping of the Distance Metric." *Pattern Recognition Letters* 33 (1). Elsevier B.V.: 92–102. doi:10.1016/j.patrec.2011.09.028.

Raymundus, Nandy Irawan;, Wawan Laksito; YS, and Sri Siswanti. 2013. "ISSN : 1693-1173 Sistem Pendukung Keputusan Untuk Menentukan Status Prestasi Siswa Menggunakan Metode K- Nearest Neighbor Raymundus Nandy Irawan, Wawan Laksito YS., Sri Siswanti." *Jurnal Ilmiah SINUS* 11 (2): 53–66.

Ren, Shuhua, and Alin Fan. 2011. "K -Means Clustering Algorithm Based On Coefficient Of Variation." In *2011 4th International Congress on Image and Signal Processing*, 2076–79. Shanghai, October 15-17.

Schenatto, Kelyn, Eduardo Godoy De Souza, Claudio Leones Bazzi, Alan Gavioli, Nelson Miguel, and Humberto Martins. 2017. "Normalization of Data for Delineating Management Zones." *Computers and Electronics in Agriculture* 143 (November). Elsevier: 238–48. doi:10.1016/j.compag.2017.10.017.

Shabani, Ali, Keramat Allah, Ali Reza, and Ali Akbar Kamgar-haghighi. 2017. "Using the Artificial Neural Network to Estimate Leaf Area." *Scientia Horticulturae* 216. Elsevier B.V.: 103–10. doi:10.1016/j.scienta.2016.12.032.

Siminski, Krzysztof. 2017. "Fuzzy Weighted C-Ordered Means Clustering Algorithm." *Fuzzy Sets and Systems* 1. Elsevier B.V.: 1–33. doi:10.1016/j.fss.2017.01.001.

Song, Yunsheng, Jiye Liang, Jing Lu, and Xingwang Zhao. 2017. "An Efficient Instance Selection Algorithm for K Nearest Neighbor Regression." *Neurocomputing*. Elsevier B.V. doi:10.1016/j.neucom.2017.04.018.

Tzortzis, Grigorios, and Aristidis Likas. 2014. "The MinMax K -Means Clustering Algorithm." *Pattern Recognition* 47 (7). Elsevier: 2505–16. doi:10.1016/j.patcog.2014.01.015.

Venugopal, Vivek, and Suresh Sundaram. 2017. "An Online Writer Identification System Using Regression-Based Feature Normalization and Codebook Descriptors." *Expert Systems With Applications* 72. Elsevier Ltd: 196–206. doi:10.1016/j.eswa.2016.11.038.

Wahono, Romi Satria, Nanna Suryana Herman, and Sabrina Ahmad. 2014. "A Comparison Framework of Classification Models for Software Defect Prediction." *Advanced Science Letters* 20 (10–12): 1945–50. doi:10.1166/asl.2014.5640.

Wang, Jigang, Predrag Neskovic, and Leon N. Cooper. 2007. "Improving Nearest Neighbor Rule with a Simple Adaptive Distance Measure." *Pattern Recognition Letters* 28 (2): 207–13. doi:10.1016/j.patrec.2006.07.002.

Weinberger, Kilian Q, and Lawrence K Saul. 2009. "Distance Metric Learning for Large Margin Nearest Neighbor Classification." *The Journal of Machine Learning Research* 10: 207–44. doi:10.1126/science.277.5323.215.

Witten, Ian, Eibe Frank, and Mark Hall. 2011. *Data Mining Practical Machine Learning Tools and Techniques Third Edition. Data Mining*. Vol. 277. Elsevier Inc. doi:10.1002/1521-3773(20010316)40:6<9823::AID-ANIE9823>3.3.CO;2-C.

Wu, Jia, Shirui Pan, Xingquan Zhu, Zhihua Cai, Peng Zhang, and Chengqi Zhang. 2015. "Self-Adaptive Attribute Weighting for Naive Bayes Classification." *Expert Systems with Applications* 42 (3). Elsevier Ltd: 1487–1502. doi:10.1016/j.eswa.2014.09.019.

Yunlong, GAO; Yixiao LIU. 2016. "An Improved Feature-Weight Method Based on K-NN." In *Proceedings of the 35th Chinese Control Conference*, 6950–56. Chengdu, July 27-29.

Zolghadri, Mansoor, Elham Parvinnia, and Robert John. 2009. "A Method of Learning Weighted Similarity Function to Improve the Performance of Nearest Neighbor." *Information Sciences* 179 (17). Elsevier Inc.: 2964–73. doi:10.1016/j.ins.2009.04.012.