JATI (Jurnal Jaringan dan Teknologi Informasi)
Vol. 3, No. 1, Juni 2023, hlm. 87-95 p-ISSN: xxxx-xxxx
DOI: 00.0000/jati e-ISSN: xxxx-xxxx

KLASIFIKASI KELAYAKAN AIR MINUM BAGI TUBUH MANUSIA MENGGUNAKAN METODE SUPPORT VEKTOR MACHINE DENGAN BACKWARD ELIMINATION

Aman Sudin[1], Munazat Salmin[2], Muhammad Fhadli[3], Arifandy Mario Mamonto[4]

[1,2,3,4] Program Studi Teknik Informatika, Fakultas Teknik, Universitas Khairun Jl. Jati Metro, Kota Ternate Selatan
Email: [1]amansudin99@gmail.com, [2]munazat@unkhair.ac.id, [3]mfhadli@unkhair.ac.id, [4]arifandymariomamonto@gmail.com

(Naskah masuk: 26-05-2023, diterima untuk diterbitkan: 01-06-2023)

Abstrak

Kualitas air dapat di deteksi berdasarkan keterkaitan parameter mineral yang terkandung di dalamnya, hal ini dapat di klasifikasikan menggunakan *machine learning*, salah satu metode yang digunakan adalah metode *Support Vektor Machine* (SVM). Kurang optimalnya metode SVM dalam pemilihan parameter kelayakan air minum sehingga apakah dengan menggunakan metode *Backward Elimination* dapat meningkatkan akurasi. Penelitian ini dilakukan dengan beberapa skenario implementasi metode SVM dan metode SVM dengan *Backward Elimination*, di dalamnya dilakukan *scaling* dan tanpa *scaling*, rasio perbandingan 80:20, selanjutnya mengeliminasi setiap parameter sehingga tersisa parameter yang paling berpengaruh. Nilai akurasi tertinggi jika hanya menggunakan metode *Support Vektor Machine* (SVM) terdapat pada jumlah data 1000 dengan tidak menggunakan *scaling* akurasinya 56%, untuk jumlah data 2000 dengan tidak menggunakan *scaling* akurasinya 47%, dan jumlah data 3276 dengan menggunakan *scaling* data akurasinya 45%. Jika diterapkan *Backward Elimination* nilai akurasi meningkat pada jumlah data 1000 dengan menggunakan *scaling* akurasinya 59%, untuk jumlah data 2000 dengan menggunakan *scaling* akurasinya 58%, tetapi pada jumlah data 3276 akurasinya menurun 1% sehingga menjadi 44%. indikator yang mempengaruhi suatu air layak di komsumsi adalah *water potability* dengan nilai 0 tidak dapat di komsumsi dan 1 dapat dikomsumsi, SVM dengan *Backward Elimination* berhasil mengklasifikasikan air minum layak dan tidak layak di komsumsi, jika menggunakan 1000 data hasil klasifikasi kelas 0 yaitu 136 dan kelas 1 adalah 64, jika menggunakan 2000 data hasil klasifikasi kelas 0 yaitu 269 dan kelas 1 adalah 131, sedangkan menggunakan 3276 data hasil klasifikasi kelas 0 yaitu 399 dan kelas 1 adalah 257.

Kata kunci: Klasifikasi, Kelayakan Air, Support Vektor Machine, Backward Elimination

CLASSIFICATION WORTHINESS OF DRINKING WATER FOR THE HUMAN BODY USING SUPPORT VECTOR MACHINE METHOD WITH BACKWARD ELIMINATION

Abstract

Water quality can be detected based on the related mineral parameters contained therein, this can be classified using machine learning,, one of the methods used is the Support Vector Machine (SVM) method. The SVM method is not optimal in selecting the feasibility parameters for drinking water so whether using the Backward Elimination method can improve accuracy. This research was conducted with several implementation scenarios of the SVM method and the SVM method with Backward Elimination, in which scaling and without scaling is carried out, the ratio ratio is 80:2, then eliminate each parameter so remaining that the most influential parameters. The highest accuracy value if only using the Support Vector Machine (SVM) method is found in the amount of data 1000 without using scaling, the accuracy is 56%, for the amount of data 2000 without using scaling the accuracy is 47%, and for the amount of data 3276 using scaling data the accuracy is 45%. If Backward Elimination is applied, the accuracy value increases at amount of data 1000 by using scaling, the accuracy is 59%, for amount of data 2000 using scaling the accuracy is 58%, but for amount of data 3276 the accuracy decreases by 1% to 44%. An indicator that affects a suitable water for consumption is the water quality of water with a value of 0 can not be consumed and 1 can be consumed, SVM with Backward Elimination has succeeded in classifying drinking water as suitable and unfit for consumption, if using 1000 data the results of class 0 classification are 136 and class 1 is 64, if using 2000 data class 0 classification results are 269 and class 1 is 131, while using 3276 data class 0 classification results are 399 and class 1 is 257.

Keywords: *Classification, Water Eligibility, Support Vector Machine, Backward Elimination*

1. PENDAHULUAN

Teknik *machine learning* pada bidang kecerdasan buatan diperkenalkan agar membantu meningkatkan kemampuan pendeteksian otomatis. Metode *machine learning* merupakan salah satu teknik penambangan data yang dapat membantu mendeteksi suatu data. Proses penebangan data atau lebih di kenal dengan data *mining* merupakan proses untuk menganalisis data besar yang setiap tahun jumlah datanya meningkat, mengekstrak dan menentukan informasi atau pengetahuan yang ada di dalam data.

Support Vektor Machine (SVM) bekerja dengan baik pada set data berdimensi tinggi. Metode *Support Vektor Machine* awalnya hanya digunakan untuk klasifikasi data yang berbentuk *linier*, namun kini dikembangkan untuk data *nonlinier* dengan menerapkan kernel *trik.* Cara kerja pada mentode ini adalah mencari *hyperplane* dan margin untuk memaksimalkan antar kelas klasifikasi [1].

Metode SVM memiliki banyak kelebihan namun terdapat kelemahan pada pemilihan fitur yang sesuai dan optimal pada bobot atribut yang digunakan sehingga menyebabkan tingkat akurasi klasifikasi menjadi rendah. Maka diperlukan seleksi fitur variabel yang akurat menggunakan algoritma pemilihan fitur, salah satu fitur yang digunakan adalah *Backward Elimination* agar bisa memaksimalkan tingkat akurasi dalam klasifikasi air minum bagi tubuh manusia nantinya. *Backward Elimination* adalah algoritma yang bertujuan untuk mengoptimalkan kinerjal suatu model dengan cara kerja pemilihan mundur [1].

Bahan alami yang diperlukan oleh mahluk hidup adalah air, air digunakan sebagai media transportasi zat makanan. Air sebagai sumber energi dan berbagai kebutuhan lainnya [2]. Dari data WHO ada sebanyak 663 juta penduduk bahwa mereka susah untuk mengakses air bersih, di prediksi 2025 nanti 2/3 penduduk dunia akan bermukiman di wilayah-wilayah yang mengalami kekurangan air [3]. WHO juga menjelaskan bahwa 829 ribu orang mati setiap tahunnya akibat dari air minum, sanitasi, dan kebersihan tangan yang tidak aman. Menurut prediksi yang dilakukan oleh *World Water Assessment Programme* (WWAP) kondisi air pada beberapah tahun kedepan untuk kebutuhan sehari-hari tidak kurang 85% air bersih akan menjadi limbah [4]. Pada tahun 2016 Badan Pusat Statistik mencatat Indonesia mengalami peningkatan yang cukup signifikan. Terkait persentase sumber air minum bersih yang layak untuk tiap rumah tangga, yaitu yang awalnya 41,39% di tahun 2012 menjadi 72,55% pada tahun 2015.

Kualitas air biasanya digambarkan dalam bentuk beberapa variabel dan parameter. Ada bermancam-macam parameter yang digunakan sebagai dasar untuk menentukan model. Dalam mengklasifikasi data, maka digunakan beberapa metode klasifikasi yang akan ditentukan secara manual dan komputasional dengan menggunakan atau memanfaatkan *machine learning.* Pada penelitian ini penulis tertarik untuk menerapkan metode *Support Vektor Machine* (SVM) dengan *Backward Elimination* untuk mengklasifikasikan kelayakan air minum bagi tubuh manusia.

Berdasarkan latar belakang diatas dapat di jelaskan sulitnya pemilihan fitur yang sesuai dan optimal pada bobot atribut yang digunakan untuk melakukan klasifikasi menggunakan metode SVM, maka digunakan metode *Backward Elimination.* Kurang optimal metode SVM dalam pemilihan parameter kelayakan air minum sehingga apakah dengan menggunakan metode *Backward Elimination* dapat meningkatkan akurasi. Penelitian ini bermaksud untuk melihat hasil klasifikasi dan peningkatan akurasi dari kelayakan air minum bagi tubuh manusia menggunakan metode *Support Vektor Machine* dengan *Backward Elimination*.

2. METODE PENELITIAN

Metode dalam penelitian ini penulis menggunakan metode *Support Vektor Machine* (SVM) dan *Backward Elimination*, selangkapnya sebagai bertikut:

1. Algoritma *Support Vektor Machine* (SVM)

Menurut Weiskhy (2021), *Support Vektor Machine* (SVM) merupakan metode klasifikasi yang memaksimalakan batas *hyperplane (maximal margin hyperplane).* Dalam SVM hanya data terpilih yang akan berkontribusi membentuk model yang digunakan dalam klasifikasi yang akan dipelajari [5]. Metode Support Vector Machine (SVM) merupakan sistem pembelajaran yang menggunakan hipotesis ruang berupa fungsi linier pada fitur berdimensi tinggi dan penyempurnaan menggunakan algoritma pembelajaran berdasarkan teori optimasi [6].

Konsep SVM adalah usaha mencari *hyperplane* "terbaik" yang berperan penting sebagai garis batas dua buah kelas. SVM mencari *hyperplane* ini berdasarkan *support vektor* dan *margin. Support vektor* adalah seluruh *vektor* data yang berjarak paling dekat dengan *hyperplane*, sedangkan *margin* menyatakan lebar dan *separating hyperplane* [7].

Linearly separable data merupakan data yang di persahkan secara *linier.* Misalkan $\{x1, \ldots., xn\}$ adalah *datasets* dan $y\ I \in \{+1, -1\}$ adalah label kelas dan data xi, label +1 menandakan bahwa data tersebut diklasifikasi sebagai kelas +1 dan label -1.

Langka pertama pada algoritma SVM adalah pendefinisian persaman suatu *hyperplane* pemisah yang ditulis dengan persamaan 1.

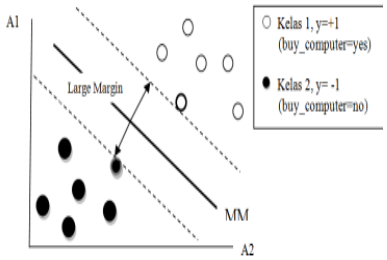
$$W.X + b = 0 \quad \text{.....(1)}$$

W merupakan bobot *vektor*, dimana $W = \{W1, W2, \ldots..W3\}$: n adalah jumlah atribut dan

b merupakan suatu scalar yang disebut dengan bias. Jika mengacuh pada atribut $A1, A2$ dengan permisalan tupel pelatihan $X = (x1, x2)$ $x1$ dan $x2$ merupakan nilai dari atribut $A1$ dan $A2$, dan jika b dianggap sebagai suatu bobot tambahan $w0$, maka persamaan suatu *hyperplane* pemisah dapat ditulis ulang dengan persamaan 2.

$$w0 + w1x1 + w2x2 = 0 \quad \text{.....(2)}$$

Setelah persamaan dapat didefinisikan, nilai $x1$ dan $x2$ dapat dimasukkan ke dalam persamaan untuk mencari bobot $w1, w2$ dan $w0$ atau b. dapat dilihat pada gambar 1 pemisahan dua kelas data dengan *margin* maksimum.



Gambar 1. Pemisahan dua kelas data dengan *margin* maksimum [8]

Pada gambar 1 dimana SVM menemukan *hyperplane* pemisah maksimum, yaitu *hyperplane* yang mempunyai jarak maksimum antara tupel pelatihan terdekat. *Support vektor* ditunjukan dengan batasan tebal pada titik tupel. Dengan demikian, setiap titik yang terletak di atas *hyperplane* pemisah memenuhi rumus:

$$w0 + w1x1 + w1x1 > 0 \quad \text{.....(3)}$$

Sedangkan, titik yang terletak dibawah *hyperplane* pemisah memenuhi persamaan dibawa ini:

$$w0 + w1x1 + w1x1 < 0 \quad \text{.....(4)}$$

Melihat dua kondisi di atas, maka didapatkan dua persamaan *hyperplane* yaitu:

$$w1x1 + w1x1 \geq 1 \text{untuk } yi = +1 \quad \text{.....(5)}$$
$$w1x1 + w1x1 \leq 1 \text{untuk } yi = -1 \quad \text{.....(6)}$$

keterangan :
Xi = data ke – I, W = nilai bobot *support vecktor* yang tegak lurus dengan *hyperplane*
b = nilai bobot, yi = kelas data ke – i
Dari persamaan diatas maka peneliti menentukan nilai parameter ambang batas untuk mendapatkan hasil prediksi, ambang batasnya yaitu 1. Jika hasil perjumlahan lebih dari atau sama dengan 1 maka dibaca “Ya”, Tetapi jika hasilnya kurang dari 1 maka hasilnya “ Tidak”.

Perumusan model SVM menggunakan trik matematika yaitu formula *lagrangian.* Berdasarkan *lagrangian formulation.* Maksimum Margin *Hyperplane* (MMH) dapat ditulis ulang sebagai suatu batas keputusan (*decision boundary)* yaitu:

$$d(X^T) = \sum_{i=1}^{l} y_i \alpha_i X_i X^T + b_0 \quad \text{.....(7)}$$

y_i adalah label kelas dari *support vektor* X_i. X^T merupakan suatu tupel test. α_i dan b_0 adalah parameter numeric yang ditentukan secara otomatis oleh optimalisasi algoritma SVM dan l adalah jumlah *vektor support* [9]

2. Backward Elimination

Backward Elimination adalah Algoritma yang dapat menghilangkan atribut yang tidak signifikan dari model [10]. Dalam prosedur *Backward Elimination*, model dimulai dengan semua atribut di dalamnya, dan atribut dengan statistik parsial paling sedikit dihilangkan. *Backward Elimination* menghilangkan atribut-atribut yang tidak relevan. Algoritma ini didasarkan pada model regrasi *linear* [11].

Langkah-langkah *Backward Elimination* sebagai berikut :
1. Membuat model dengan meregresikan variabel respon Y dengan semua variabel prediktor.
2. Hapus variabel prediktor satu per satu dengan menguji parameter menggunakan Ftes parsial. Nilai Fpartial terkecil dibandingkan dengan Ftabel.
a. Jika Fparsial < Ftabel, maka X dikeluarkan dari model dan dilanjutkan dengan pembuatan model baru tanpa variabel tersebut.
b. Jika Fparsial > Ftabel, maka proses dihentikan artinya tidak ada variabel yang perlu dikeluarkan dan persamaan terakhir tersebut yang digunakan/dipilih.

3. HASIL DAN PEMBAHASAN

1. Analisis Data

Bagaimana mengklasifikasikan kualitas air minum dengan sumber *datasets* dari web kaggel.com. berikut *datasets water_potability* yang digunakan pada penelitian ini.

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	NaN	204.890455	20791.318981	7.300212	368.516441	564.308654	10.379783	86.990970	2.963135	0
1	3.716080	129.422921	18630.057858	6.635246	NaN	592.885359	15.180013	56.329076	4.500656	0
2	8.099124	224.236259	19909.541732	9.275884	NaN	418.606213	16.868637	66.420093	3.055934	0
3	8.316766	214.373394	22018.417441	8.059332	356.886136	363.266516	18.436524	100.341674	4.628771	0
4	9.092223	181.101509	17978.986339	6.546600	310.135738	398.410813	11.558279	31.997993	4.075075	0
...	...	...	...	...	...	...	...	...	...	...
3271	4.668102	193.681735	47580.991603	7.166639	359.948574	526.424171	13.894419	66.687695	4.435821	1
3272	7.808856	193.553212	17329.802160	8.061362	NaN	392.449580	19.903225	NaN	2.798243	1
3273	9.419510	175.762646	33155.578218	7.350233	NaN	432.044783	11.039070	69.845400	3.298875	1
3274	5.126763	230.603758	11983.869376	6.303357	NaN	402.883113	11.168946	77.488213	4.708658	1
3275	7.874671	195.102299	17404.177861	7.509306	NaN	327.459760	16.140368	78.698446	2.309149	1

3276 rows × 10 columns

Gambar 2 Data Water_potability

2. Prapremosesan Data

Dari datasest yang peneliti gunakan masih

terdapat data yang kosong atau berbentuk NaN, data akan diisi dengan angka 0. Sebelum di isi peneliti mengeluarkan parameter Potability, sehingga tersisa 9 parameter. Data yang telah di isi seperti pada gambar 3 dibawa ini.

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity
0	0.000000	204.890455	20791.318981	7.300212	368.516441	564.308654	10.379783	86.990970	2.963135
1	3.716080	129.422921	18630.057858	6.635246	0.000000	592.885359	15.180013	56.329076	4.500656
2	8.099124	224.236259	19909.541732	9.275884	0.000000	418.606213	16.868637	66.420093	3.055934
3	8.316766	214.373394	22018.417441	8.059332	356.886136	363.266516	18.436524	100.341674	4.628771
4	9.092223	181.101509	17978.986339	6.546600	310.135738	398.410813	11.558279	31.997993	4.075075
...	...	...	...	...	...	...	...	...	...
3271	4.668102	193.681735	47580.991603	7.166639	359.948574	526.424171	13.894419	66.687695	4.435821
3272	7.808856	193.553212	17329.802160	8.061362	0.000000	392.449580	19.903225	0.000000	2.798243
3273	9.419510	175.762646	33155.578218	7.350233	0.000000	432.044783	11.039070	69.845400	3.298875
3274	5.126763	230.603758	11983.869376	6.303357	0.000000	402.883113	11.168946	77.488213	4.708658
3275	7.874671	195.102299	17404.177061	7.509306	0.000000	327.459760	16.140368	78.698446	2.309149

3276 rows × 9 columns

Gambar 3 data yang telah diisi dengan angka 0

3. Pembagian Data

Membagi *datasets water_potability* menjadi beberapa set data untuk digunakan nantinya untuk di imlementasi pada metode dimulai dari jumlah data 1000, 2000, dan 3276.

4. Pembagian Data *Training* dan *Testing*

datasets dibagi 2 yaitu data training dan data testing secara random menggunakan fungsi train_test_split dari library sklearn. Dari masing-masing pembagian data yang di mulai dari 1000, 2000, dan 3276 datasets. Rasio yang umum digunakan adalah 80:20 [12]. Lebih jelasnya dapat dilihat pada tabel 1.

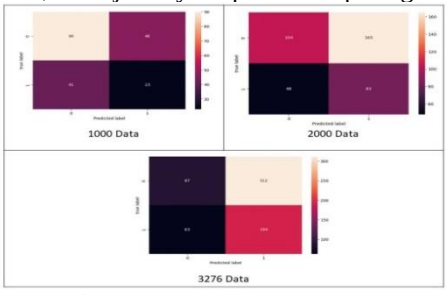
Tabel 1 Pembagian data Training dan data Testing

Jumlah data	Data *training*	Data *testing*
1000	800	200
2000	1600	400
3276	2620	656

5. Implementasi metode SVM

A. Tidak menggunakan Standar data atau *scaling*

Metode SVM diterapkan tanpa menggunakan *scaling* data, dengan nilai *C*=1.0, *cache_size*=100, *decision_funcion*_shape=’ovr’, dengan nilai *gamma*=1.0, menggunakan kernel *linear*, maxsimal itersi=1, dengan tol=0.00001. dilakukan pengujian *confusion matrix* dengan tidak menggunakan *scaling* pada jumlah data 1000, jumlah data 2000, dan jumlah data 3276, lebih jelasnya dapat dilihat pada gambar 4.



Gambar 4. *Confusion Matrix* Tidak Menggunakan *Scaling* 1000, 2000, Dan 3276 Data

Berdasarkan hasil pengujian *confusion matrix* pada gambar 4, metode SVM pada jumlah data 1000 dapat mengenali pola data lebih baik dari pada jumlah data 2000 dan jumlah data 3276. Lebih jelasnya dapat dilihat pada Tabel 2.

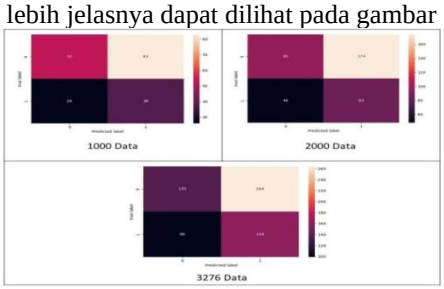
Tabel 2 Hasil Kinerja Evaluasi Pengujian Masing-Masing Data

Jumlah Data	**Label**	***Precision***	***Recall***	**Data**	**Akurasi**
1000	0	0,69	0,66	136	56%
	1	0,33	0,36	64	
2000	0	0,68	0,39	269	47%
	1	0,33	0,63	131	
3276	0	0,58	0,22	399	43%
	1	0,38	0,75	257	

Hasil pengujian kinerja pada masing-masing jumlah data tanpa *scaling* data, pada tabel 2 menunjukan hasil kinerja *evaluasi* dengan rata-rata *precision*, dan *recal.* Dengan akurasi sebesar 56% untuk jumlah data 1000, 47% untuk jumlah data 2000, dan 43% untuk jumlah data 3276. Label yang diklasifikasi sebagai 0 pada jumlah data 1000, 2000, dan 3276 adalah 136, 269, dan 399. Serta label yang diklasifikasi sebagai 1 pada jumlah data 1000, 2000, dan 3276 adalah 64, 131, 257.

B. Menggunakan Standar data atau *scaling*

Metode SVM diterapkan menggunakan *scaling* data, dengan nilai *C*=1.0, *cache*_size=100, *decision_funcion_shape*=’ovr’, dengan nilai *Gamma*=1.0, menggunakan kernel *linear*, maxsimal itersi=1, dengan tol=0.00001. dilakukan pengujian *confusion matrix* dengan menggunakan *scaling* pada jumlah data 1000, jumlah data 2000, dan jumlah data 3276, lebih jelasnya dapat dilihat pada gambar 5.



Gambar 5 *Confusion Matrix* Menggunakan *Scaling* 1000, 2000, 3276 Data

Hasil pengujian *confusion matrix* pada gambar 5. metode SVM pada jumlah data 1000 dapat menggenali pola, akan tetapi memiliki banyak kesalahan dari pada pengujian sebelumnya. Lebih jelasnya terdapat pata tabel 3.

Tebel 3. Hasil Kinerja Evaluasi Pengujian Masing-Masing Data Menggunakan *Scaling*

Jumlah Data	**Label**	***Precision***	***Recall***	**Data**	**Akurasi**
1000	0	0,67	0,39	136	46%
	1	0,31	0,59	64	
2000	0	0,66	0,35	269	45%
	1	0,32	0,63	131	
3276	0	0,58	0,34	399	45%

	1	0,37	0,61	257	

Hasil pengujian kinerja pada masing-masing jumlah data dengan *scaling*, pada tabel 3 menunjukan hasil kinerja *evaluasi* dengan rata-rata *precision*, dan *recall* dengan akurasi tertinggi 46% pada 1000 data. Label yang diklasifikasi sebagai 0 pada jumlah data 1000, 2000, dan 3276 adalah 136, 269, dan 399. Serta label yang diklasifikasi sebagai 1 pada jumlah data 1000, 2000, dan 3276 adalah 64, 131, 257.

6. Implementasi metode SVM dengan *backward elimination*

Sebelum dilakukan imlplementasi SVM dilakukan penerapa fitur *backward elimination* untuk mengeliminasi fitur atau parameter yang tidak terlalu berpengaruh pada tahapan selanjutnya.

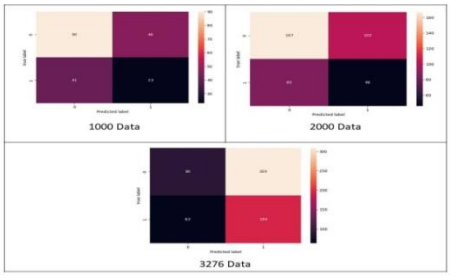
Tabel 4. Mengeliminasi *K_Feature* Satu Persatu

1. No	**2. *K_Feature***
3. 1	*[ph, Hardness, Solids, Chlomines, sulfate, Conductivity, Organic_carbon, Trihalomethanes, dan Turbidity.]*
4. 2	*['ph', 'Hardness', 'Solids', 'Sulfate', 'Conductivity', 'Organic_carbon', 'Trihalomethanes', 'Turbidity']*
5. 3	*['ph', 'Hardness', 'Solids', 'Conductivity', 'Organic_carbon', 'Trihalomethanes', 'Turbidity']*
6. 4	*['ph', 'Hardness', 'Solids', 'Conductivity', 'Organic_carbon', 'Trihalomethanes']*
7. 5	*['ph', 'Hardness', 'Solids', 'Organic_carbon', 'Trihalomethanes']*
8. 6	*['ph', 'Solids', 'Organic_carbon', 'Trihalomethanes']*
9. 7	*['ph', 'Solids', 'Trihalomethanes']*
10. 8	*['Solids', 'Trihalomethanes']*
11. 9	*['Solids']*

1. Implementasi metode SVM dengan tidak menggunakan *scaling*

A. menggunakan 9 *K_feature* dari masing-masing jumlah data

Metode SVM diterapkan tanpa menggunakan *scaling* data, dengan nilai *C*=1.0, *cache_size*=100, *decision_funcion*_shape=’ovr’, dengan nilai *gamma*=1.0, menggunakan kernel *linear*, maxsimal itersi=1, dengan tol=0.00001. dilakukan pengujian *confusion matrix* dengan tidak menggunakan *scaling* pada jumlah data 1000, jumlah data 2000, dan jumlah data 3276, lebih jelasnya dapat dilihat pada gambar 6.



Gambar 6 *Confusion Matrix* 9 *K_Feature* Dengan Tidak Menggunakan *Scaling* 1000, 2000, Dan 3276 Data

Berdasarkan hasil pengujian *confusion matrix* pada gambar 6, metode SVM pada jumlah data 1000 dapat mengenali pola data lebih baik dari pada jumlah data 2000 dan jumlah data 3276. Lebih jelasnya dapat dilihat pada tabel 5.

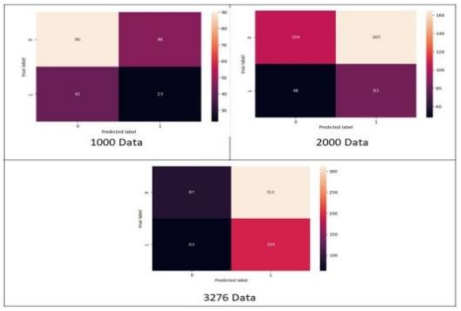
Tabel 5. Hasil Kinerja Evaluasi Pengujian 9 *K_Feature* Tidak Menggunakan *Scaling*

Jumlah Data	**Label**	***Precision***	***Recall***	**Data**	**Akurasi**
1000	0	0.69	0.66	136	56%
	1	0.33	0.36	64	
2000	0	0.68	0.39	269	47%
	1	0.33	0.63	131	
3276	0	0.58	0.22	399	43%
	1	0.38	0.75	257	

Hasil pengujian kinerja pada masing-masing jumlah data dengan tanpa *scaling*, pada tabel 5 menunjukan hasil kinerja *evaluasi* dengan rata-rata *precision*, dan *recall* dengan akurasi tertinggi 56% pada jumlah data 1000. Label yang diklasifikasi sebagai 0 pada jumlah data 1000, 2000, dan 3276 adalah 136, 269, dan 399. Serta label yang diklasifikasi sebagai 1 pada jumlah data 1000, 2000, dan 3276 adalah 64, 131, 257.

B. menggunakan 1 *K_feature* dari masing-masing jumlah data

Metode SVM diterapkan tanpa menggunakan *scaling* data, dengan nilai *C*=1.0, *cache_size*=100, *decision_funcion*_shape=’ovr’, dengan nilai *gamma*=1.0, menggunakan kernel *linear*, maxsimal itersi=1, dengan tol=0.00001. dilakukan pengujian *confusion matrix* dengan tidak menggunakan *scaling* pada jumlah data 1000, jumlah data 2000, dan jumlah data 3276, lebih jelasnya dapat dilihat pada gambar 7.



Gambar 7 *Confusion Matrix* 1 *K_Feature* Dengan Tidak Menggunakan *Scaling* 1000, 2000, Dan 3276 Data

Berdasarkan hasil pengujian *confusion matrix* pada gambar 5, metode SVM pada jumlah data 1000 dapat mengenali pola data lebih baik dari pada jumlah data 2000 dan jumlah data 3276. Lebih jelasnya dapat dilihat pada tabel 6.

Tabel 6 Hasil Kinerja Evaluasi Pengujian 1 *K_Feature* Tidak Menggunakan *Scaling*

Jumlah Data	**Label**	***Precision***	***Recall***	**Data**	**Akurasi**
1000	0	0.69	0.66	136	56%
	1	0.33	0.33	64	

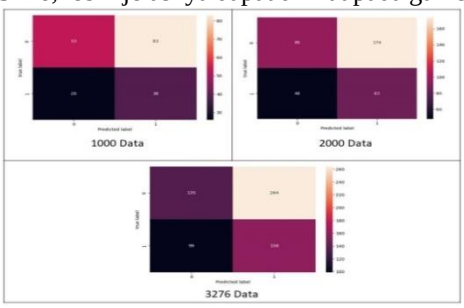
2000	0	0.66	0.62	269	53%
	1	0.31	0.35	131	
3276	0	0.59	0.23	399	43%
	1	0.39	0.75	257	

Hasil pengujian kinerja pada masing-masing jumlah data dengan tanpa *scaling*, pada tabel 6 menunjukan hasil kinerja *evaluasi* dengan rata-rata *precision*, dan *recall* dengan akurasi tertinggi 56% pada jumlah data 1000, untuk jumlah data 2000 menggalami peningkatan 6% sehingga menjadi 53%. Label yang diklasifikasi sebagai 0 pada jumlah data 1000, 2000, dan 3276 adalah 136, 269, dan 399. Serta label yang diklasifikasi sebagai 1 pada jumlah data 1000, 2000, dan 3276 adalah 64, 131, 257.

2. Implementasi metode SVM dengan menggunakan *scaling*

A. menggunakan 9 *K_feature* dari masing-masing jumlah data

Metode SVM diterapkan menggunakan *scaling* data, dengan nilai *C*=1.0, *cache_size*=100, *decision_funcion*_shape=’ovr’, dengan nilai *gamma*=1.0, menggunakan kernel *linear*, maxsimal itersi=1, dengan tol=0.00001. dilakukan pengujian *confusion matrix* dengan tidak menggunakan *scaling* pada jumlah data 1000, jumlah data 2000, dan jumlah data 3276, lebih jelasnya dapat dilihat pada gambar 8.



Gambar 8 *Confusion Matrix* 9 *K_Feature* Menggunakan *Scaling* 1000, 2000, Dan 3276 Data

Berdasarkan hasil pengujian *confusion matrix* pada gambar 8, metode SVM pada jumlah data 1000 dapat mengenali pola data lebih baik dari pada jumlah data 2000 dan jumlah data 3276. Lebih jelasnya dapat dilihat pada tabel 6.

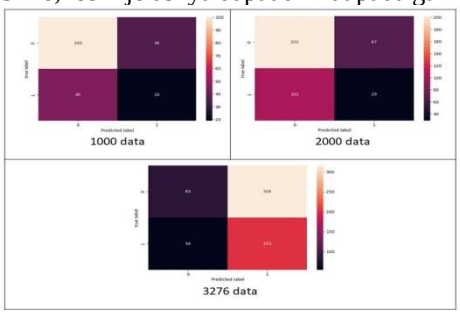
Tabel 7 Hasil Kinerja Evaluasi Pengujian 9 *K_Feature* Tidak Menggunakan *Scaling*

Jumlah Data	Label	*Precision*	*Recall*	Data	Akurasi
1000	0	0.67	0.39	136	46%
	1	0.31	0.59	64	
2000	0	0.66	0.35	269	45%
	1	0.32	0.63	131	
3276	0	0.58	0.34	399	45%
	1	0.37	0.61	257	

Hasil pengujian kinerja pada masing-masing jumlah data dengan *scaling*, pada tabel 6 menunjukan hasil kinerja *evaluasi* dengan rata-rata *precision*, dan *recall* dengan akurasi tertinggi 46% pada jumlah data 1000. Label yang diklasifikasi sebagai 0 pada jumlah data 1000, 2000, dan 3276 adalah 136, 269, dan 399. Serta label yang diklasifikasi sebagai 1 pada jumlah data 1000, 2000, dan 3276 adalah 64, 131, 257.

B. Menggunakan 1 *K_feature* dari masing-masing jumlah data

Metode SVM diterapkan menggunakan *scaling* data, dengan nilai *C*=1.0, *cache_size*=100, *decision_funcion*_shape=’ovr’, dengan nilai *gamma*=1.0, menggunakan kernel *linear*, maxsimal itersi=1, dengan tol=0.00001. dilakukan pengujian *confusion matrix* dengan tidak menggunakan *scaling* pada jumlah data 1000, jumlah data 2000, dan jumlah data 3276, lebih jelasnya dapat dilihat pada gambar 9.



Gambar 9. *Confusion Matrix* 1 *K_Feature* Menggunakan *Scaling* 1000, 2000, Dan 3276 Data

Berdasarkan hasil pengujian *confusion matrix* pada gambar 9, metode SVM pada jumlah data 1000 dapat mengenali pola data lebih baik dari pada jumlah data 2000 dan jumlah data 3276. Lebih jelasnya dapat dilihat pada tabel 8.

Tabel 8 Hasil Kinerja Evaluasi Pengujian 1 *K_Feature* Menggunakan *Scaling*

Jumlah Data	Label	*Precision*	*Recall*	Data	Akurasi
1000	0	0.69	0.74	136	59%
	1	0.35	0.30	64	
2000	0	0.66	0.75	269	58%
	1	0.30	0.22	131	
3276	0	0.61	0.21	399	44%
	1	0.39	0.79	257	

Hasil pengujian kinerja pada masing-masing jumlah data dengan *scaling*, pada tabel 7 menunjukan hasil kinerja *evaluasi* dengan rata-rata *precision*, dan *recall* dengan akurasi tertinggi 59% pada jumlah data 1000 mengalami peningkatan 13%, pada jumlah data 2000 menggalami peningkatan 13%, sedangkan 3276 mengalami penurunan 1%. Label yang diklasifikasi sebagai 0 pada jumlah data 1000, 2000, dan 3276 adalah 136, 269, dan 399. Serta label yang diklasifikasi sebagai 1 pada jumlah data 1000, 2000, dan 3276 adalah 64, 131, 257.

7. Analisis

Setelah dilakukan berbagai macam scenario pengujian terhadap Metode SVM serta Metode SVM dengan *Backward Elimination*, maka dapat dianalisi hasil perbandigan pengujian berdasarkan tingkat akurasi yang di hasilkan.

Tabel 9. Perbandingan Kinerja Evaluasi SVM Dan SVM Dengan *Backward Elimination*

Jumlah Data	***Scaling***	**Akurasi SVM**	**Akurasi SVM + *Backward Elimination***									***Datasets***	**nilai C**	***gamma***
			9	8	7	6	5	4	3	2	1			
1000	ya	46%	46%	46%	46%	58%	43%	47%	52%	49%	59%	80:20	1.0	1.0
	tidak	56%	56%	56%	56%	56%	56%	56%	56%	56%	56%	80:20	1.0	1.0
2000	ya	45%	45%	43%	45%	48%	45%	50%	39%	44%	58%	80:20	1.0	1.0
	tidak	47%	47%	47%	54%	54%	53%	53%	53%	53%	53%	80:20	1.0	1.0
3276	ya	45%	45%	51%	50%	46%	52%	48%	51%	52%	44%	80:20	1.0	1.0
	tidak	43%	43%	43%	43%	43%	43%	43%	43%	43%	43%	1.0	1.0	

Berdasarkan tabel 9 diatas didapat dua metode akurasi yaitu Metode SVM dan metode SVM dengan *Backward Elimination* dari setiap jumlah data yang di uji, jika menggunakan Metode SVM hasil akurasi tertinggi terdapat pada jumlah data 1000 dengan tidak menggunakan *scaling* data, nilai *C=1.0, gamma=1.0,* dan perbandingan *datasets* 80:20 hasil akurasinya 56%, jika menggunakan *scaling* data pada jumlah data 1000 memiliki akurasi yang rendah. Sedangkan jumlah data 2000 dan 3276 memiliki performa akurasi yang rendah, akurasi tertinggi pada jumlah data 2000 adalah 47%, untuk jumlah data 3276 adalah 45%.

Pada metode SVM dengan *Backward Elimination* akurasi tertinggi terdapat pada jumlah data 1000 menggunakan *scaling* data untuk 1 *K_Feature* dengan nilai *C=1.0, gamma=1.0,* dan perbandingan *datasets* 80:20 dengan tingkat akurasi 59%. Sedangkan jika tidak menggunakan *scaling* data pada jumlah data 1000 nilai akurasinya tidak menggalami peningkatan performa, hal ini terjadi karena fitur *Backward Elimination* tidak memiliki pengaruh pada jumlah data 1000.

Pada jumlah data 2000 pada metode SVM dengan *Backward Elimination*, nilai *C=1.0, gamma=1.0,* perbandingan *datasets* 80:20 dan jika tidak menggunakan *scaling* data hasil akurasi menggalami peningkatan pada *K_Feature* ke 7 yaitu 54%, akan tetapi turun nilai akurasinya 53% pada *K_Feature* ke 5. Sedangkan jika menggunakan *scaling* data akurasi tertinggi 58% pada *K_Feature* ke 1. Untuk jumlah data 3276 jika menggunakan *scaling* data hasil akurasi tertinggi 52% pada *K_Feature* ke 5 dan *K_Feature* ke 2, jika tidak menggunakan *scaling* data hasil akurasinya tidak menggalami peningkatan, hal ini terjadi karena *K_Feature* tidak berpenggaruh pada jumlah data 3276.

Selanjutnya jumlah data dan standar data atau *scaling* data berpengaruh pada metode SVM maupun Metode SVM dengan *Backward* Elimination, dengan jumlah data 1000 menghasilkan nilai akurasi yang lebih tinggi dari jumlah data 2000 dan 3276 data yang di gunakan. Sedangkan proses *scaling* tidak berpengaruh pada Metode SVM dilihat dari rendahnya akurasi pada jumlah data 1000, 2000, dan 3276 yang digunakan. untuk Metode SVM dengan *Backward Elimination* jika menggunakan *scaling* data memiliki nilai akurasi yang lebih tinggi jika di bandingkan dengan tidak menggunakan *scaling* data, dilihat dari *K_Feature* ke 1 pada jumlah data 1000. Ketika nilai C dan nilai *gamma* diubah hasil akurasinya pada data 1000 akurasinya tetap 59%.

Tabel 10. Perbandingan akurasi nilai C dan *gamma* pada 1000 data

Jumlah data		**C = 1.0**	**C = 5.0**	**C = 10.0**	**C = 100.0**
1000 data	***gamma=1***	*59%*	*59%*	*59%*	*59%*
	gamma=5	59%	59%	59%	59%
	gamma=10	59%	59%	59%	59%
	gamma=100	59%	59%	59%	59%

Dari perbandingan diatas nilai C dan *gamma* tidak berpengaruh terhadap nilai akurasi dari model yang dibuat, dimana jika nilai C dan gamma bertambah hasil akurasinya tetap 59%. Jika menggunakan perbandingan untuk pemisahan data *training* dan *testing* yang lain seperti perbandingan 90:10, 70:30, 60:40 atau 50:50, diharapkan nilai akurasinya meningkat pada jumlah data 1000, 2000, dan 3276. Atau memakai model validasi/evaluasi *K-Fold Cross*, yaitu salah satu teknik validasi silang, tujuanya untuk menghilangkan bias pada data. Ada

beberapa *feature selection* untuk menyeleksi parameter seperti *forward selection,* dan *Backward Elimination*, yang digunakan dalam penelitan ini adalah *Backward* Elimination, *Backward Elimination* bekerja dengan cara kerja memasukan semua parameter lalu mengeliminasi satu persatu sehingga meyisahkan parameter yang paling berpengaruh terhadap model, dengan begitu akan menaikan nilai akurasi dari metode SVM.

Metode SVM dengan *Backward Elimination* masih memiliki nilai akurasi yang rendah dengan akurasi 59% pada 1000 data. Pada penelitian sebelumnya yang dilakukan oleh [4] pada metode SVM memiliki nilai akurasi yang lebih rendah yakni 54,37% sedangkan pada peneltian yang sama hasil akurasi pada metode yang lain akurasinya 72,81%. Sehingga dapat dikatakan pada pemelitia ini metode SVM dengan *Backward Elimination* menghasilkan nilai akurasi yang lebih tinggi dari pada hanya mengunakan SVM saja.

4. KESIMPULAN

Berdasarkan hasil pengujian dan evaluasi pada metode *Support Vektor Machine* dan metode *Support Vaktor Machine* dengan *Backward Elimination,* maka dapat diambil kesimpulan antara lain:

1. Metode *Support Vaktor Machine* dengan *Backward Elimination* menggunakan nilai *C=1.0, gamma=1.0, datasets* 80:20, dan diterapkan tanpa *scaling* dan menggunakan *scaling*, menggunakan *scaling* data berhasil mengklasifikasikan kelayakan air minum dengan menghasilkan nilai akurasi tertinggi sebesar 59%, sedangkan tidak menggunakan *scaling* data hasil akurasi tertingginya 56%.
2. Metode *Support Vektor Machine* dengan *Backward Elimination* berhasil mengklasifikasikan kelayakan air minum dengan menggunakan jumlah data 1000, 2000, 3276, dengan rasio 80:20, jika menggunakan 1000 data hasil klasifikasi kelas 0 yaitu 136 dan kelas 1 adalah 64, jika menggunakan 2000 data hasil klasifikasi kelas 0 yaitu 269 dan kelas 1 adalah 131, sedangkan menggunakan 3276 data hasil klasifikasi kelas 0 yaitu 399 dan kelas 1 adalah 257.
3. Metode SVM dengan *Backward Elimination* hasil akurasinya lebih tinggi dari pada hanya menggunakan metode SVM saja, dengan nilai akurasi 59% untuk SVM dengan *Backward Elimination*, sedangkan SVM 52%.
4. Metode SVM dengan *Backward Elimination* jumlah data 1000 yang tertinggi dari jumlah data 2000 dan 3276, dimana jumlah data 1000 nilai akurasinya 59%, jumlah data 2000 nilai akurasinya 58%, dan jumlah data 3276 nilai akurasinya 52%.

5. DAFTAR PUSTAKA

[1] R. Resmiati, "Klasifikasi Pasien Kanker Payudara Menggunakan Metode *Support Vector Machin*e dengan *Backward Elimination*," *SISTEMASI*, vol. 10, pp. 381–393, 2021, [Online]. Available: http://sistemasi.ftik.unisi.ac.id.

[2] F. Muhamad, "Kualitas Air Pada Sumber Mata Air Di Pura Taman Desa Sanggalangit Sebagai Sumber Air Minum Berbasis Metode Storet," *J. Pendidik. Geogr. Undiksha*, vol. 7, no. 2, pp. 74–84, 2019, doi: 10.23887/jjpg.v7i2.20691.

[3] U. Sri, "Ketersediaan Air Bersih Untuk Kesehatan : Kasus Dalam Pencegahan Diare Pada Anak," *Optim. Peran Sains dan Teknol. untuk Mewujudkan Smart City*, no. June, pp. 211–236, 2017, [Online]. Available: https://www.researchgate.net/publication/326057942%0AKETERSEDIAAN.

[4] P. A. Riyantoko, "Analisis Sederhana Pada Kualitas Air Minum Berdasarkan Akurasi Model Klasifikasi Dengan Menggunakan *Lucifer Machine Learning*," *Siminar Nas. Sains Data (SANADA 2021)*, vol. 2021, no. Senada, pp. 12–18, 2021, [Online]. Available: https://senada.upnjatim.ac.id/index.php/senada/article/view/20.

[5] W. S. Dharmawan, "Komparasi Algoritma Klasifikasi SVM-PSO dan C4.5-PSO Dalam Prediksi Penyakit Jantung," *J. Inform. Manaj. dan Komput.*, vol. 13, no. 2, pp. 31–41, 2021, doi: http://dx.doi.org/10.36723/juri.v13i2.301.

[6] A. M. Puspitasari, "Klasifikasi Penyakit Gigi Dan Mulut Menggunakan Metode *Support Vector Machine*," *Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. March, pp. 802–810, 2018, [Online]. Available: http://j-ptiik.ub.ac.id.

[7] F. S. Jumeilah, "Penerapan *Support Vector Machine* (SVM) untuk Pengkategorian Penelitian," *J. RESTI* (Rekayasa Sist. dan Teknol. Informasi), vol. 1, no. 1, pp. 19–25, 2017, doi: 10.29207/resti.v1i1.11.

[8] D. Kurniawan, "Optimasi Algoritma *Support Vector Machine* (Svm) Menggunakan Adaboost," *Teknol. Inf.*, vol. 9, p. 13, 2013, [Online]. Available: http://research.pps.dinus.ac.id.

[9] D. Kurniawan, "Optimasi Algoritma *Support*

Vector Machine (Svm) Menggunakan Adaboost Untuk Penilaian Risiko Kredit," *J. Teknol. Inf.*, vol. 9, no. 1, pp. 1–13, 2013.

[10] Farizul Ma'arif, "Optimasi Fitur Menggunakan *Backward Elimination* Dan Algoritma SVM Untuk Klasifikasi Kanker Payudara," *J. Inform.*, vol. 4, no. 1, pp. 46–53, 2017, doi: https://doi.org/10.31294/ji.v4i1.1548.

[11] S. A. D. Ghani, "Algoritma *k-Nearest Neighbor* Berbasis *Backward Elimination* Pada Client Telemarketing," *Pros. Semin. Ilm. Sist. Inf. DAN Teknol. INFORMAS*, vol. VIII, no. 2, pp. 141–150, 2019, [Online]. Available: http://ejurnal.dipanegara.ac.id/index.php/sisiti/article/view/610.

[12] V. R. Joseph, "*Optimal Ratio for Data Splitting,"* pp. 1–16, 2021.