**PENERAPAN SUPPORT VECTOR MACHINE UNTUK KLASIFIKASI STANDAR KELAYAKAN AIR MINUM DENGAN SKEMA OPTIMALISASI PARAMETER**

**USULAN SKRIPSI**

**Oleh :**

**BAYU PERMANA YOGA**

**NIM. 200605110145**

****

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM**

**MALANG**

**2025**

# HALAMAN PERSETUJUAN

# A close-up of a documentHALAMAN PENGESAHAN

REVISI USULAN SKRIPSI

PENERAPAN SUPPORT VECTOR MACHINE (SVM) UNTUK KLASIFIKASI STANDAR KELAYAKAN AIR MINUM DENGAN SKEMA OPTIMALISASI PARAMETER



Oleh :

BAYU PERMANA YOGA

NIM. 200605110145

|  |  |  |
| --- | --- | --- |
| Pada tanggal: Oktober 2025  Telah Diperiksa, Disetujui, dan Disahkan Oleh: | | |
| Ketua Penguji | : Hani Nurhayati, M.T | ( ) |
|  | NIP. 19780625 200801 1 006 |
|  |  |
| Anggota Penguji I | : Shoffin Nahwa Utama, M.T | ( ) |
|  | NIP. 19860703 202012 1 003 |
|  |  |  |
| Anggota Penguji II/ | : Dr. Ir. Fachrul Kurniawan, M.MT., IPU | ( ) |
| Pembimbing I | NIP. 19771020 200912 1 001 |

# DAFTAR ISI

[HALAMAN PERSETUJUAN i](#_Toc213996238)

[HALAMAN PENGESAHAN ii](#_Toc213996239)

[DAFTAR ISI iii](#_Toc213996240)

[DAFTAR TABEL v](#_Toc213996241)

[DAFTAR GAMBAR vi](#_Toc213996242)

[BAB I PENDAHULUAN 7](#_Toc213996243)

[1.1 Latar Belakang 7](#_Toc213996244)

[1.2 Rumusan Masalah 9](#_Toc213996245)

[1.3 Tujuan Penelitian 9](#_Toc213996246)

[1.4 Batasan Masalah 10](#_Toc213996247)

[1.5 Manfaat Penelitian 10](#_Toc213996248)

[BAB II STUDI PUSTAKA 11](#_Toc213996249)

[2.1 Penelitian Terkait 11](#_Toc213996250)

[2.2 Kelayakan Air Minum 17](#_Toc213996251)

[2.3 Support Vector Machine 19](#_Toc213996252)

[BAB III METODOLOGI PENELITIAN 21](#_Toc213996253)

[3.1 Metodologi Penelitian 21](#_Toc213996254)

[3.2 Pengumpulan Data 22](#_Toc213996255)

[3.3 Desain Sistem 24](#_Toc213996256)

[3.3.1 Pengumpulan data 24](#_Toc213996257)

[3.3.2 Preprocessing 26](#_Toc213996258)

[3.3.3 Dataset 26](#_Toc213996259)

[3.3.4 Data Split 27](#_Toc213996260)

[3.3.5 Tuning Parameter (GridSearch, inisialisasi C, gamma, kernel, CV) 27](#_Toc213996261)

[3.3.6 Pelatihan 29](#_Toc213996262)

[3.3.7 Testing 29](#_Toc213996263)

[3.3.8 Evaluasi 29](#_Toc213996264)

[3.4 Skenario Pengujian 30](#_Toc213996265)

[3.4.1 Normalisasi Data dengan Z-Score Standardization 30](#_Toc213996266)

[3.4.2 Perhitungan Support Vector Machine Kernel RBF 33](#_Toc213996267)

[3.4.3 Perhitungan Metrik Evaluasi 34](#_Toc213996268)

[BAB IV UJI COBA DAN PEMBAHASAN 36](#_Toc213996269)

[4.1 Lingkungan Implementasi 36](#_Toc213996270)

[4.2 Deskripsi dan Persiapan Data 36](#_Toc213996271)

[4.2.1 Deskripsi Dataset 36](#_Toc213996272)

[4.2.2 Analisis Keseimbangan Kelas 36](#_Toc213996273)

[4.2.3 Pra-pemrosesan Data 36](#_Toc213996274)

[4.3 Hasil Skenario Pengujian 37](#_Toc213996275)

[4.3.1 Hasil *Hyperparameter Tuning* (GridSearchCV) 37](#_Toc213996276)

[4.3.2 Perbandingan Kinerja Model pada Data Uji 37](#_Toc213996277)

[4.3.3 Visualisasi Confusion Matrix 37](#_Toc213996278)

[4.4 Pembahasan (Diskusi) 38](#_Toc213996279)

[4.4.1 Perbandingan Kinerja Model (SVM vs. Random Forest) 38](#_Toc213996280)

[4.4.2 Pengaruh Skenario Pembagian Data (70:30 vs. 80:20) 38](#_Toc213996281)

[4.4.3 Analisis Model Terbaik 38](#_Toc213996282)

[4.4.4 Efektivitas Penanganan *Class Imbalance* 39](#_Toc213996283)

[4.4.5 Jawaban atas Rumusan Masalah 39](#_Toc213996284)

[4.5 Implementasi Model Akhir 39](#_Toc213996285)

[BAB V KESIMPULAN 40](#_Toc213996286)

[5.1 Kesimpulan 40](#_Toc213996287)

[5.2 Saran 40](#_Toc213996288)

[6 DAFTAR PUSTAKA 41](#_Toc213996289)

# DAFTAR TABEL

[Tabel 3.1 Penjelasan Data Kualitas Air 10](#_Toc213164812)

[Tabel 3.2 Dataset Kualitas Air 10](#_Toc213164813)

[Tabel 3.3 Skenario Pengujian 17](#_Toc213164814)

[Tabel 3.4 Statistik Dasar Data 18](#_Toc213164815)

[Tabel 3.5 Perhitungan Manual Standard Scaling 18](#_Toc213164816)

[Tabel 3.6 hasil normalisasi menggunakan standard scaling 19](#_Toc213164817)

[Tabel 3.7 Perhitungan manual kernel RBF 20](#_Toc213164818)

[Tabel 3.8 perhitungan Kernel RBF 20](#_Toc213164819)

[Tabel 3.9 matrix evaluasi 20](#_Toc213164820)

# DAFTAR GAMBAR

[Gambar 2.1 Ilustrasi *Support Vector Machine* 6](#_Toc213164824)

[Gambar 3.1 Metodologi Penelitian 8](#_Toc213164825)

[Gambar 3.2 Desain Sistem klasifikasi kelayakan air minum dengan SVM 11](#_Toc213164826)

# BAB I PENDAHULUAN

## Latar Belakang

Air merupakan kebutuhan dasar bagi kehidupan manusia, berfungsi vital tidak hanya untuk keperluan rumah tangga, tetapi juga menyokong berbagai sektor seperti pertanian, industri, transportasi, dan pariwisata (Atmaja, 2019). Inti dari kebutuhan ini adalah ketersediaan air minum yang layak konsumsi, yang harus memenuhi standar kualitas tertentu agar tidak menimbulkan risiko negatif terhadap kesehatan. Sayangnya, masalah ini masih signifikan di Indonesia; berdasarkan data Badan Pusat Statistik (BPS) tahun 2023, sekitar 18% penduduk Indonesia masih mengonsumsi air yang tidak memenuhi standar kelayakan, baik dari sisi parameter fisik, kimia, maupun mikrobiologi. Kondisi air minum yang tidak layak ini berpotensi besar menyebabkan berbagai penyakit menular seperti diare, kolera, dan gangguan kesehatan serius lainnya.

Untuk mengatasi tantangan tersebut, penentuan kelayakan air minum memerlukan metode evaluasi yang cepat dan efisien. Dalam konteks ini, parameter fisik menjadi indikator awal yang krusial. Parameter seperti pH, salinitas, kekeruhan, suhu, dan *Total Dissolved Solids* (TDS) dapat diukur dengan alat sederhana tanpa memerlukan pengujian laboratorium yang memakan waktu dan biaya besar. Pendekatan berbasis parameter fisik ini sangat penting dan relevan, terutama di wilayah terpencil yang sulit menjangkau fasilitas laboratorium, sehingga memungkinkan deteksi awal kelayakan air minum dapat dilakukan secara cepat dan akurat.

Sejalan dengan tuntutan efisiensi dan akurasi, pemanfaatan teknologi *machine learning* (pembelajaran mesin) menjadi solusi modern dalam klasifikasi kualitas air. Algoritma yang dikenal unggul dalam tugas klasifikasi, seperti *Support Vector Machine* (SVM), telah banyak dimanfaatkan dalam analisis data di berbagai bidang. SVM bekerja dengan mencari *hyperplane* optimal untuk memisahkan data ke dalam kelas-kelas yang berbeda, sehingga sangat sesuai untuk menentukan apakah data air (berdasarkan parameter fisiknya) memenuhi standar kelayakan. Pemanfaatan teknologi ini memungkinkan deteksi kualitas air dilakukan secara cepat dan akurat, kualitas air juga harus memperhatikan kesehatan dan kelayakan konsumsi masyarakat.

Lebih dari sekadar persoalan teknis dan kesehatan, pentingnya akses terhadap air bersih dan layak konsumsi juga tercermin dalam perspektif universal dan agama, khususnya Islam, di mana air memiliki kedudukan yang sangat mulia. Dalam Al-Qur’an disebutkan,

اَوَلَمْ يَرَ الَّذِيْنَ كَفَرُوْٓا اَنَّ السَّمٰوٰتِ وَالْاَرْضَ كَانَتَا رَتْقًا فَفَتَقْنٰهُمَاۗ وَجَعَلْنَا مِنَ الْمَاۤءِ كُلَّ شَيْءٍ حَيٍّۗ اَفَلَا يُؤْمِنُوْنَ ۝٣٠

“*Apakah orang-orang kafir tidak mengetahui bahwa langit dan bumi, keduanya, dahulu menyatu, kemudian Kami memisahkan keduanya dan Kami menjadikan segala sesuatu yang hidup berasal dari air? Maka, tidakkah mereka beriman?*” (QS. Al-Anbiya: 30)

Kemuliaan dan urgensi air dalam kehidupan manusia tidak hanya ditegaskan dari sisi ajaran agama, tetapi juga tercermin dalam upaya nyata untuk menjaga dan memastikan kualitasnya. Ayat ini menegaskan bahwa air adalah sumber utama kehidupan yang harus dijaga dan dikelola dengan baik. Selain itu, Rasulullah SAW juga bersabda, “Manusia mempunyai hak yang sama dalam tiga hal: air, rumput, dan api.” (HR. Abu Dawud). Kedua dalil ini memperkuat bahwa akses terhadap air bersih adalah hak dasar manusia, menjadikan upaya menjamin kualitas air sebagai prioritas bersama, baik dari sisi kesehatan masyarakat, kepentingan sosial, maupun ajaran agama.

Sejalan dengan pentingnya menjaga kualitas air, berbagai inovasi teknologi telah dikembangkan untuk mendukung upaya tersebut, salah satunya dengan pemanfaatan metode *machine learning*. Penelitian Maulina Putri et al. (2023) menunjukkan bahwa SVM dapat memberikan akurasi tinggi dalam klasifikasi kualitas air. Contohnya, sebuah penelitian menggunakan sensor pH, kekeruhan, dan TDS yang dihubungkan dengan mikrokontroler ESP32, kemudian data yang diperoleh diklasifikasikan menggunakan SVM dengan tingkat akurasi di atas 90% bahkan hingga 100% (Maulina Putri et al., 2023). Temuan ini menegaskan bahwa penerapan teknologi modern dapat menjadi solusi efektif dan praktis untuk memastikan air yang dikonsumsi masyarakat memenuhi standar kelayakan, sekaligus mewujudkan hak dasar manusia atas air bersih secara nyata.

Meskipun SVM terbukti efektif, masih terdapat tantangan dalam penerapannya, seperti pemilihan kernel yang tepat, penanganan *outlier* dalam data, serta optimasi parameter model untuk meningkatkan akurasi prediksi. Oleh karena itu, penelitian ini akan mengimplementasikan Support Vector Machine (SVM) untuk menentukan kelayakan air minum berdasarkan standar peraturan umum kualitas air, dengan fokus pada parameter fisik yaitu, suhu, kekeruhan, dan TDS; serta satu parameter kimiawi yaitu pH.

Selain berkontribusi dalam peningkatan teknologi analisis kualitas air, penelitian ini juga mendukung pencapaian *Sustainable Development Goals* (SDGs) Nomor 6, yaitu *Clean Water and Sanitation*, yang bertujuan untuk memastikan ketersediaan dan pengelolaan air bersih bagi seluruh masyarakat. Dengan menerapkan metode SVM, diharapkan dapat tercipta solusi yang lebih efisien, akurat, dan berkelanjutan dalam penentuan kelayakan air minum, sehingga dapat membantu pemerintah dan masyarakat dalam menjaga kualitas air yang dikonsumsi.

## Rumusan Masalah

Seberapa efektif model SVM dapat diterapkan dalam mengklasifikasi status kelayakan air minum (layak/tidak layak) berdasarkan data set kualitas air

## Tujuan Penelitian

Menganalisis efektivitas penerapan model *Support Vector Machine* (SVM) untuk mengklasifikasikan status kelayakan air minum (layak dan tidak layak) berdasarkan parameter kualitas air.

## Batasan Masalah

* Data Penelitian terbatas pada beberapa parameter fisik yaitu *Total Dissolved Solids* (TDS), Kekeruhan, Suhu, Salinitas dan satu parameter kimiawi yaitu pH sehingga jika ada parameter mikrobiologi atau parameter kimia lain yang tidak terdeteksi secara fisik, akan mempengaruhi kelayakan air minum.
* Parameter lain seperti parameter mikrobiologi, parameter kimia selain pH, ataupun parameter radioaktif tidak dapat dikumpulkan karena keterbatasan biaya dan alat.

## Manfaat Penelitian

Penelitian ini diharapkan dapat bermanfaat bagi masyarakat yang berkaitan dengan lokasi diambilnya data demi meningkatkan kualitas hidup dan memenuhi *Sustainable Development Goals* (SDGs) nomor 6 yaitu *Clean Water and Sanitation* atau air bersih dan sanitasi layak.

# BAB II STUDI PUSTAKA

## Penelitian Terkait

Ramon et al mengklasifikasikan status gizi bayi menggunakan SVM dengan kernel RBF berdasarkan data antropometri, mencapai akurasi 87%. Studi tersebut menunjukkan keandalan SVM dalam menangani data kesehatan dengan fitur terbatas. Keterkaitan dengan penelitian ini terletak pada penggunaan kernel RBF dan data nonlinear yang serupa, menjadi dasar pemilihan metode SVM untuk klasifikasi parameter fisik air minum. Namun, penelitian ini memperluas cakupan dengan menambahkan sensor ESP32 untuk pengumpulan data *real-time* dan parameter yang berbeda signifikan, di mana Ramon et al. menggunakan data dengan parameter antropometri, sedangkan penelitian ini menggunakan data kelayakan air (Ramon et al., 2022).

Dzakwan dan Subektiningsih melakukan optimalisasi SVM dengan kernel *polynomial* untuk klasifikasi risiko kesehatan ibu hamil. Penelitian tersebut berhasil meningkatkan akurasi dari 60% menjadi 71% setelah melalui tahapan *preprocessing*. Studi ini menggarisbawahi urgensi *preprocessing* data sebagai fondasi untuk normalisasi dan penanganan *outlier* dalam penelitian klasifikasi air minum. Perbedaannya terletak pada fokus penelitian ini terhadap parameter fisik air, bukan parameter klinis (Dzakwan & Subektiningsih, 2023).

Tabel 2.1 Penelitian Terkait

| Penulis | Input | Metode | Output | Hasil | Batasan |
| --- | --- | --- | --- | --- | --- |
| (Ramon et al., 2022) | 1. Tanggal pengukuran 2. Berat badan 3. Tinggi badan | *Support Vector Machine* (SVM) | * Akurasi * Presisi * Recall * F1-score | Dalam penelitian ini, status gizi bayi dibagi menjadi enam kategori yang berbeda. Penggunaan kernel RBF menghasilkan performa klasifikasi yang sangat baik, dengan akurasi, presisi, dan *recall* masing-masing 87%, serta f1-score 86%. | Data set yang digunakan merupakan data lengkap yang diperoleh dari Posyandu Kecamatan Bangunan Purba pada tahun 2020 sebanyak 1732 data. |
| (Dzakwan & Subektiningsih, 2023) | 1. Usia 2. Tekanan darah sistolik 3. Diastolik BP 4. Kadar glukosa 5. Detak jantung 6. Risiko | *Support Vector Machine* (SVM) | * Akurasi * Presisi * Recall * F1-score | Dilakukan pengelompokan risiko kesehatan ibu hamil menjadi tiga kategori. Hasil klasifikasi risiko kematian ibu hamil dengan metode kernel *polynomial* menunjukkan peningkatan ketepatan dari 60% (sebelum pemrosesan data) menjadi 71% (setelah pemrosesan data). | Data set yang digunakan adalah data yang diperoleh dari UCI data set sebanyak 1014 data. |
| (Vidiastanta et al., 2020) | 1. pH level 2. *Total Dissolved Solids* (TDS) 3. Nitrit (NO2) 4. Nitrat (NO3) 5. Kesadahan 6. Klorida 7. Mangan | *K-Nearest Neighbors* (KNN) dan *Support Vector Machine* (SVM) | * Akurasi | Penelitian ini membandingkan KNN dan SVM untuk klasifikasi kualitas air. KNN sedikit lebih unggul dengan akurasi 88,94% berbanding 87,71% pada SVM. Parameter optimal telah ditemukan (KNN pada k=7; SVM dengan beberapa optimasi). Kedua algoritma mampu mengklasifikasi dengan baik meski data tidak seimbang, dan berpotensi diterapkan untuk klasifikasi kualitas air praktis, namun KNN menunjukkan kinerja yang marginal lebih baik dan stabil. | Penelitian ini memiliki keterbatasan utama pada ukuran data yang kecil (167 data dari PDAM Kota Malang) dan rentang waktu pengumpulan data yang singkat (tujuh bulan), sehingga kurang representatif untuk kondisi kualitas air yang beragam dan perubahan jangka panjang. |
| (Sudin et al., 2023) | 1. pH 2. Kesadahan 3. Zat padat 4. Kloramin 5. Sulfat 6. Konduktivitas 7. Karbon organik 8. Trihalometan 9. Kekeruhan | *Support Vector Machine* (SVM) dengan *Backward Elimination* | * Akurasi | Penelitian menemukan bahwa SVM yang dikombinasikan dengan *Backward* *Elimination* mencapai akurasi tertinggi sebesar 59% (menggunakan 1000 titik data dengan penskalaan), yang lebih baik daripada penggunaan SVM saja yang mencapai akurasi 52%. Meskipun pendekatan gabungan menunjukkan peningkatan, akurasi keseluruhan tetap relatif rendah, menunjukkan adanya ruang untuk optimasi lebih lanjut dalam metode klasifikasi kualitas air. | Keterbatasan dataset ini, yaitu hanya 3.276 data dari Kaggle.com dan penanganan nilai hilang dengan pengisian nol, berpotensi mengurangi representasi kondisi kualitas air yang beragam dan akurasi model. |
| (Maulina Putri et al., 2023) | 1. PH 2. Kesadahan 3. Solids 4. Kloramin 5. Sulfat 6. Konduktivitas 7. Karbon Organik 8. Trihalometan 9. Kekeruhan 10. Potabilitas | *Support Vector Machine* (SVM) | * Akurasi * Presisi * *Recall* * F1-score | Penelitian ini mengevaluasi empat fungsi kernel SVM untuk klasifikasi kualitas air, dengan kernel RBF mencapai kinerja terbaik (100%), diikuti oleh kernel Polinomial (98%), sementara kernel Linear dan Sigmoid berkinerja buruk (60%). Hasil ini menunjukkan bahwa SVM dengan kernel RBF sangat efektif untuk mengklasifikasikan kualitas air berdasarkan parameter WHO, menjadikannya alat yang andal untuk penilaian kualitas air otomatis. | Keterbatasan dataset penelitian ini meliputi penggunaan data dari satu sumber geografis saja (Perumda Tirta Pase), ukuran data total yang tidak jelas, dan potensi kurangnya keragaman dalam kondisi kualitas air, yang dapat mempengaruhi kemampuan generalisasi model ke sumber air atau wilayah lain. |
| (Yoga, Bayu Permana, 2025) | 1. pH 2. Suhu 3. Kekeruhan 4. TDS 5. Salinitas | *Support Vector Machine (Kernel RBF)* | * Akurasi * Presisi * *Recall* * F1-score |  |  |

Vidiastanta membandingkan KNN dan SVM untuk klasifikasi kualitas air PDAM Malang, dengan KNN sedikit lebih unggul (akurasi 88,94% vs 87,71%). Hasil ini menginspirasi penelitian ini untuk menguji kombinasi kernel SVM (misal RBF) dan optimasi *hyperparameter* guna mengungguli kinerja KNN(Vidiastanta et al., 2020). Penelitian ini memiliki kesamaan dengan Vidiastanta dalam penggunaan parameter air untuk pengolahan data, namun hanya pada parameter pH dan TDS. Vidiastanta memanfaatkan parameter tambahan seperti NO3, NO2, Kesadahan, dan Mangan, sedangkan penulis menggunakan parameter Suhu, Salinitas, dan Turbiditas/Kekeruhan.

Maulina Putri mengevaluasi empat kernel SVM pada klasifikasi kualitas air, di mana kernel RBF mencapai akurasi sempurna (100%). Keberhasilan ini disebabkan kemampuan RBF dalam menangani hubungan non-linear antar parameter kimia air seperti pH dan sulfat. Studi ini menjadi acuan utama untuk memvalidasi keunggulan RBF dalam penelitian ini, meskipun datasetnya terbatas pada satu sumber geografis(Maulina Putri et al., 2023).

Studi Maulina Putri mencakup sepuluh parameter kualitas air, di mana enam parameter di antaranya juga digunakan dalam penelitian ini. Empat parameter yang tidak diikutsertakan dalam penelitian ini adalah Sulfat, Trihalometan, Kloramin, dan Kesadahan.

Pada studi tersebut, Maulina Putri melaporkan akurasi sempurna sebesar 100% dalam klasifikasi kualitas air dengan menggunakan SVM kernel RBF. Hasil ini melampaui kinerja kernel *polynomial* (98%), linear, maupun *sigmoid*. Keberhasilan kernel RBF dijelaskan oleh kemampuannya dalam memetakan data ke dimensi yang lebih tinggi, sehingga mampu menangani pola-pola kompleks, seperti interaksi non-linear antara parameter kimia misalnya antara *hardness* dan sulfat. Meskipun dataset yang digunakan hanya bersumber dari satu wilayah, yaitu Perumda Tirta Pase, temuan ini memperkuat argumen bahwa RBF merupakan pilihan optimal untuk klasifikasi kualitas air, terutama ketika fitur-fitur saling bergantung secara non-linear. Temuan tersebut juga menjadi dasar pemilihan kernel RBF dalam penelitian ini, dengan penyesuaian parameter γ dan C untuk menghindari terjadinya *overfitting* (Maulina Putri et al., 2023).

Berbeda dari penelitian sebelumnya yang mayoritas menggunakan data sekunder, penelitian ini mengusulkan metode akuisisi data primer secara *real-time* menggunakan integrasi sensor ESP32. untuk pengumpulan data parameter fisik air (pH, TDS, Kekeruhan, Suhu, dan Salinitas). Hal ini mengatasi keterbatasan penelitian terdahulu yang bergantung pada data sekunder atau alat laboratorium mahal. Kombinasi *hardware* (ESP32) dan *software* (SVM) memungkinkan deteksi kelayakan air secara cepat dan portabel, khususnya di daerah terpencil. Selain itu, penelitian ini menggunakan data set yang lebih beragam (1200 data dari 2 desa) untuk meningkatkan generalisasi model, serta fokus pada parameter fisik sebagai indikator awal yang mudah diukur. Inovasi lain terletak pada strategi optimasi *hyperparameter* menggunakan 5-*fold* *cross-validation*, yang belum banyak diterapkan dalam penelitian serupa. Dengan pendekatan ini, penelitian tidak hanya akurat, tetapi juga skalabel untuk diterapkan dalam sistem pemantauan air berbasis *Internet of things* (IoT) di masa depan.

## Kelayakan Air Minum

Air dikatakan senyawa kompleks karena mengandung banyak zat dan mineral di dalamnya. Namun, tidak semua kandungan zat dan mineral yang terdapat di dalam air dapat dicerna dengan oleh tubuh manusia. Selain itu air juga rentan terkontaminasi oleh zat-zat dan bakteri berbahaya bagi tubuh manusia. Hal tersebut dapat terjadi dikarenakan tercemarnya sumber air ataupun lingkungan yang berada di sekitar sumber air. Misalnya pada kasus air sungai, penurunan kualitas air terjadi sebagai akibat pembuangan limbah yang tidak terkendali dari aktivitas pembangunan di sepanjang sungai, sehingga tidak sesuai dengan daya dukung sungai(Kospa & Rahmadi, 2019). Dibutuhkan pengawasan dan pengolahan yang ketat sekitar sumber air sehingga dapat dihasilkan kualitas air bersih yang sesuai dengan standar kualitas air bersih serta memenuhi standar air yang layak untuk dikonsumsi oleh manusia (Kencanawati & Mustakim, 2017).

Tabel 2.2 Parameter Kelayakan Air, Satuan, dan Nilai Batas Maksimum

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Satuan** | **Nilai Batas Maksimum** |
| *Total Dissolved Solids* (TDS) | Mg/L | 500 |
| Kekeruhan | NTU | 5 |
| Suhu | °C | Suhu udara ±3 |
| pH | - | 6.5-8.5 |
| Salinitas | mg/L atau ppt | 200 |

Salah satu parameter fisik yang digunakan dalam menilai kelayakan air minum adalah Total *Dissolved Solids* (TDS), yang mengacu pada jumlah zat terlarut dalam air, termasuk mineral, garam, dan logam. Berdasarkan standar yang berlaku, kadar TDS yang disarankan adalah maksimal 500 mg/L(Permenkes RI, 2010). TDS yang terlalu tinggi dapat menyebabkan rasa air menjadi tidak enak dan meningkatkan risiko gangguan kesehatan seperti penyakit ginjal jika dikonsumsi dalam jangka panjang. Namun, air dengan TDS yang terlalu rendah juga kurang baik karena dapat menyebabkan air menjadi "kosong" tanpa mineral penting yang dibutuhkan oleh tubuh.

Selain TDS, kekeruhan air juga menjadi indikator penting dalam menilai kualitas air minum. Kekeruhan disebabkan oleh partikel tersuspensi seperti lumpur, pasir, dan mikroorganisme yang dapat menghambat transmisi cahaya melalui air. Standar maksimal kekeruhan dalam air minum adalah 5 NTU (*Nephelometric Turbidity Units*) (Permenkes RI, 2010). Air yang terlalu keruh berpotensi mengandung patogen atau bahan pencemar yang berbahaya bagi kesehatan manusia. Oleh karena itu, proses penyaringan yang baik sangat diperlukan untuk mengurangi kekeruhan sebelum air dikonsumsi.

Suhu air minum juga memiliki peran penting dalam kenyamanan dan kualitas air. Meskipun Standar yang berlaku tidak menetapkan standar spesifik untuk suhu, umumnya suhu yang disarankan berkisar antara 10°C hingga 25°C. Suhu air yang terlalu tinggi dapat meningkatkan laju reaksi kimia dalam air yang berpotensi menurunkan kualitasnya, sementara suhu yang terlalu rendah dapat mengurangi kelarutan mineral dalam air(Permenkes RI, 2010).

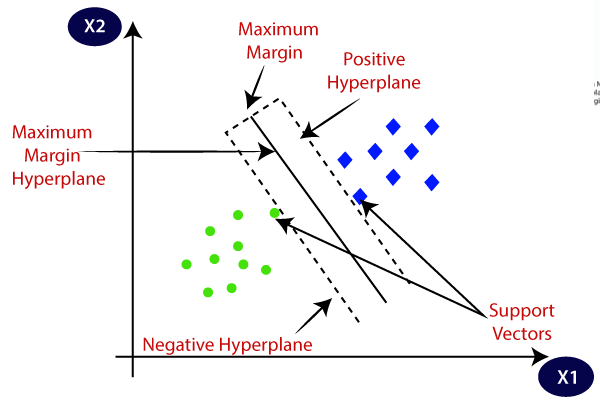
Parameter pH digunakan untuk mengukur tingkat keasaman atau kebasaan air. Standar yang ditetapkan dalam Standar yang berlaku untuk air minum adalah rentang pH 6,5 hingga 8,5(Permenkes RI, 2010). Air yang terlalu asam (pH < 6,5) dapat menyebabkan korosi pada pipa distribusi dan berisiko mengandung logam berat berbahaya, sementara air yang terlalu basa (pH > 8,5) dapat memberikan rasa pahit dan mempengaruhi efektivitas desinfeksi. Oleh karena itu, pH air perlu dijaga dalam rentang yang aman untuk memastikan kualitas air minum tetap optimal. Tinggi rendahnya pH dipengaruhi oleh perubahan kandungan O2 maupun CO2. Tidak semua makhluk hidup bisa bertahan terhadap perubahan nilai pH, namun proses alam memiliki mekanisme yang unik agar perubahan pH tidak terjadi atau terjadi tetapi dengan cara yang perlahan (Nursaini & Harahap, 2022).

Salinitas merupakan ukuran konsentrasi garam yang terlarut dalam air, biasanya dinyatakan dalam satuan mg/L atau ppt (*part per thousand*). Parameter ini sangat penting terutama pada air minum dan lingkungan perairan, karena kadar salinitas yang terlalu tinggi dapat menyebabkan rasa air menjadi asin dan berpotensi mengganggu kesehatan jika dikonsumsi dalam jangka waktu lama. Selain itu, salinitas juga memengaruhi keseimbangan osmotik tubuh dan dapat berdampak pada keberlangsungan makhluk hidup di dalam air. Standar salinitas untuk air minum umumnya ditetapkan agar tidak melebihi 200 mg/L, sehingga air tetap aman dan nyaman untuk dikonsumsi masyarakat(Permenkes RI, 2010).

## Support Vector Machine

*Support Vector Machine* (SVM) adalah salah satu algoritma *machine learning* yang digunakan untuk klasifikasi dan regresi. SVM bekerja dengan mencari *hyperplane* terbaik yang dapat memisahkan dua atau lebih kelas dalam suatu dataset. Metode SVM dikembangkan dengan menggabungkan beberapa *classifier biner* sehingga dapat digunakan untuk menangani masalah klasifikasi dengan lebih dari dua kelas (*multiclass*)(Liang, 2004). Prinsip utama SVM adalah mencari margin maksimal, yaitu jarak terjauh antara *hyperplane* dengan titik-titik terdekat dari setiap kelas yang disebut *support vectors*. Jika data tidak dapat dipisahkan secara linier, SVM menggunakan fungsi kernel untuk memetakan data ke dalam dimensi yang lebih tinggi agar dapat dipisahkan dengan lebih baik.

Dalam konteks penentuan kelayakan air minum, SVM dapat digunakan untuk mengklasifikasikan air menjadi kategori layak atau tidak layak berdasarkan parameter fisik seperti *Total Dissolved Solids* (TDS), kekeruhan, suhu, dan pH. Dataset yang digunakan akan memiliki fitur-fitur ini sebagai input, sementara label keluaran akan berupa kategori kelayakan air. SVM akan membangun model yang dapat memisahkan air yang layak dan tidak layak dengan optimal, bahkan jika distribusi datanya tidak linier, dengan menggunakan teknik kernel trick.



gambar . Ilustrasi *Support Vector Machine*

Untuk mengatasi data yang memiliki pola yang kompleks, SVM menggunakan fungsi kernel non-linear seperti *Radial Basis Function* (RBF) Kernel atau *Polynomial Kernel*. Dengan fungsi ini, SVM mampu menangani data yang tidak bisa dipisahkan secara linier dengan cara memproyeksikan data ke dimensi yang lebih tinggi. Hal ini sangat penting dalam analisis kelayakan air minum karena parameter fisik air seringkali memiliki hubungan yang tidak sederhana dan saling mempengaruhi satu sama lain.

Secara matematis, SVM dengan fungsi kernel non-linear bekerja dengan mengubah data ke dimensi yang lebih tinggi menggunakan fungsi Φ(x) sehingga data dapat dipisahkan secara linier. Model matematisnya dinyatakan sebagai berikut:

|  |  |  |
| --- | --- | --- |
|  |  | . |

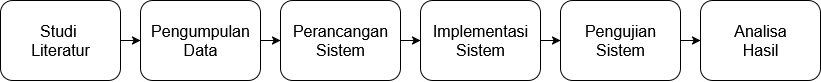
Di mana :

* adalah fungsi kernel,
* adalah parameter yang diperoleh dari optimasi,
* adalah label kelas (-1 atau +1),
* adalah bias.

# BAB III METODOLOGI PENELITIAN

## Metodologi Penelitian

Penelitian ini diawali dengan studi literatur untuk mengidentifikasi gap dalam penelitian terdahulu, seperti ketergantungan pada data laboratorium atau dataset terbatas. Hasil kajian menjadi dasar pemilihan algoritma SVM dengan kernel RBF dan parameter fisik (pH, TDS, kekeruhan, suhu, dan salinitas) sebagai fitur utama. Tahap pengumpulan data dilakukan menggunakan sensor yang terhubung ke perangkat ESP32, menghasilkan 1200 data air dari 2 desa di Kabupaten Blitar. Data ini kemudian dirancang dalam sistem yang mencakup *preprocessing* penanganan *missing value*, normalisasi, *outlier*) dan pembagian dataset 80:20 untuk pelatihan dan pengujian model.



Gambar . Metodologi Penelitian

Model SVM diimplementasikan dengan *library Python* seperti *Scikit-Learn*, di optimasi menggunakan kernel RBF dan validasi *5*-*fold* cross-validation. Pada tahap pengujian, kinerja model dievaluasi melalui metrik akurasi, presisi, *recall*, dan F1-score, serta *confusion matrix* untuk mengidentifikasi kesalahan klasifikasi. Hasilnya dianalisis untuk menilai keandalan model dalam konteks nyata, seperti risiko *False Positive* (air tidak layak diklasifikasi sebagai layak). Inovasi utama terletak pada integrasi ESP32 untuk akuisisi data *real-time*, memungkinkan deteksi cepat di daerah terpencil tanpa laboratorium. Proses ini bersifat iteratif, di mana temuan analisis dapat digunakan untuk menyempurnakan model atau perangkat keras.

## Pengumpulan Data

Penelitian ini melibatkan pengambilan data di dua Desa yang berbeda di Kabupaten Blitar, dengan masing-masing lokasi menyumbangkan 600 sampel data. Dengan demikian, total data yang terkumpul untuk analisis berjumlah 1200 sampel. Data yang dikumpulkan secara khusus berfokus pada kualitas air dan berasal dari dua sumber utama: sumur milik salah satu warga yang berlokasi di Desa Ploso, dan air sungai yang mengalir melalui Desa Siraman. Pengambilan data dari kedua sumber ini bertujuan untuk memberikan gambaran komprehensif mengenai kondisi kualitas air di wilayah studi.

Data yang dimanfaatkan dalam implementasi *Support Vector Machine* (SVM) untuk menentukan kelayakan air minum dikumpulkan melalui serangkaian sensor. Sensor-sensor tersebut meliputi sensor pH untuk mengukur tingkat keasaman atau kebasaan air, sensor suhu untuk mencatat temperatur air, sensor salinitas untuk mendeteksi kadar garam terlarut, sensor *Total Dissolved Solids* (TDS) untuk mengetahui konsentrasi total padatan terlarut, serta sensor kekeruhan untuk mengukur tingkat kejernihan air.

Semua sensor ini terintegrasi dengan mikrokontroler ESP32. ESP32 dipilih karena efisiensi pemrosesan data dan fitur konektivitas nirkabelnya, yang memungkinkan transmisi data ke sistem analisis lebih lanjut. Dengan kombinasi sensor yang akurat dan mikrokontroler yang andal, data kualitas air dapat diperoleh secara real-time dan akurat, menjadi dasar yang kuat untuk analisis kelayakan air minum menggunakan metode SVM. Penjelasan mengenai setiap data dataset dapat dilihat pada tabel berikut

Tabel . Penjelasan Data Kualitas Air

|  |  |
| --- | --- |
| **Fitur** | **Keterangan** |
| *Potential Hydrogen* (pH) | pH adalah ukuran tingkat keasaman atau kebasaan suatu larutan yang dinyatakan dalam skala 0 hingga 14, di mana nilai di bawah 7 menunjukkan sifat asam, nilai 7 bersifat netral, dan nilai diatas 7 bersifat basa. |
| Suhu | Suhu adalah ukuran derajat panas atau dinginnya suatu benda atau lingkungan yang dinyatakan dalam satuan Celsius. |
| Kekeruhan/Turbiditas | Kekeruhan adalah kondisi di mana partikel tersuspensi seperti lumpur, pasir, dan mikroorganisme menghambat transmisi cahaya melalui air. |
| *Total Dissolved Solids* (TDS) | Jumlah zat terlarut dalam air, termasuk mineral, garam, dan logam |
| Salinitas | Salinitas adalah ukuran total konsentrasi garam terlarut dalam air, biasanya dinyatakan dalam satuan mg/L atau ppt (*parts per thousand*). |

Berikut merupakan contoh data hasil pengambilan data air.

Tabel . Dataset Kualitas Air

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Indeks** | **pH** | **Suhu**  **(°C)** | **Kekeruhan (NTU)** | **Total Dissolved Solids**  **(mg/L)** | **Salinitas** | **Kelayakan** |
| 1 | 7.28 | 29 | 6.06 | 225 | 30.82 | Tidak Layak |
| 2 | 6.89 | 23 | 2.98 | 270 | 32.15 | Layak |
| 3 | 7.85 | 26 | 0.59 | 438 | 32.04 | Layak |
| 4 | 8.79 | 25 | 5.79 | 347 | 31.98 | Tidak Layak |
| 5 | 6.23 | 23 | 2.05 | 189 | 31.81 | Tidak Layak |
| 6 | 5.95 | 32 | 4.59 | 179 | 31.82 | Tidak Layak |
| 7 | 6.73 | 26 | 2.8 | 384 | 31.77 | Layak |
| 8 | 7.25 | 31 | 5.74 | 122 | 31.94 | Tidak Layak |
| 9 | 8.22 | 29 | 6.11 | 310 | 31.60 | Tidak Layak |
| 10 | 6.73 | 32 | 6.93 | 279 | 31.73 | Tidak Layak |

## Desain Sistem

Proses diawali dengan tahapan Start dan input Dataset Kualitas Air, yang merupakan kumpulan data mentah. Tahapan yang paling krusial adalah *Preprocessing*, yang bertujuan untuk membersihkan dan menstandarisasi data agar siap digunakan. Di dalamnya, terdapat tiga langkah utama: *Missing Value Handling* untuk mengatasi data yang hilang (misalnya dengan imputasi rata-rata atau median); *Outlier Handling* untuk mengidentifikasi dan menangani nilai-nilai ekstrem yang dapat mengganggu pelatihan model; dan *Handling* Data *Imbalance* untuk menyeimbangkan distribusi kelas dalam variabel target, memastikan model tidak bias terhadap kelas mayoritas.

A diagram of a process

AI-generated content may be incorrect.

Gambar . Desain Sistem klasifikasi kelayakan air minum dengan SVM

Desain sistem klasifikasi kelayakan air minum seperti tertera di Gambar 3.2 masing masing proses yang akan dilakukan dijelaskan sebagai berikut:

### Pengumpulan data

Sistem pengambilan data real-time dibangun menggunakan perangkat IoT berbasis mikrokontroler ESP32. Secara spesifik, perangkat ini terintegrasi dengan lima sensor yang memungkinkan pemantauan komprehensif terhadap berbagai parameter lingkungan air. Sensor-sensor ini meliputi:

* **Sensor Suhu:** Mengukur suhu air.
* **Sensor TDS (Total Dissolved Solids):** Mengukur konsentrasi total padatan terlarut dalam air, memberikan indikasi kualitas air dan keberadaan mineral terlarut.
* **Sensor Salinitas:** Menentukan tingkat salinitas air, vital untuk memantau lingkungan akuatik di mana konsentrasi garam sangat bervariasi.
* **Sensor Kekeruhan/Turbiditas:** Mengukur kekeruhan atau kejernihan air, menunjukkan keberadaan partikel tersuspensi yang dapat memengaruhi penetrasi cahaya dan kesehatan akuatik.
* **Sensor pH:** Mengukur tingkat keasaman atau alkalinitas air, parameter fundamental untuk menjaga keseimbangan ekologis lingkungan air.

Untuk memastikan operasi yang berkelanjutan dan otonom, perangkat IoT ini dilengkapi dengan beberapa komponen pendukung krusial:

* **Panel Surya:** Berfungsi sebagai sumber energi terbarukan utama, memanfaatkan sinar matahari untuk menghasilkan listrik.
* **Kontroler Panel Surya:** Mengelola daya yang dihasilkan oleh panel surya, mengoptimalkan pengisian daya dan mencegah pengisian daya berlebih atau kurang pada baterai.
* **Aki:** Menyediakan penyimpanan energi, memungkinkan perangkat untuk terus beroperasi bahkan selama periode tanpa sinar matahari atau dalam kondisi cahaya rendah.

Siklus pengumpulan data kami konfigurasi untuk frekuensi tinggi, dengan data diambil setiap 6 detik. Interval ini memungkinkan kami melakukan pemantauan dinamis dan deteksi cepat perubahan dalam parameter lingkungan yang dipantau, menjadikannya alat yang sangat efektif untuk aplikasi yang membutuhkan data real-time dan respons cepat.

### Preprocessing

Metode *Preprocessing* yang digunakan adalah data *cleaning*, dan *labeling.* Data cleaning meliputi penghapusan kolom yang tidak diperlukan seperti ServerTimeStamp, OriginalTimeStamp, dan apiKey. Sedangkan data *labeling* adalah proses melabeli dataset dengan ketentuan seperti pada tabel. Hasil *labeling* digunakan untuk melatih model SVM.

1. *Missing value handling*

Untuk menangani *missing data* dengan mengidentifikasi nilai yang kosong. Data tersebut kemudian akan diisi (imputasi) menggunakan metode statistik atau dihapus barisnya agar tidak merusak analisis

1. *Outlier handling*

Untuk menangani *outlier*, akan dilakukan identifikasi data pencilan yang ekstrem. Data ini kemudian di hapus atau disesuaikan (transformasi) agar tidak mengganggu keakuratan model atau hasil analisis statistik

1. *Handling data imbalance*

Pada tahap ini menangani data tidak seimbang dengan teknik penyeimbangan, seperti *oversampling* atau *undersampling*. Tujuannya agar model tidak bias pada kelas mayoritas dan mampu memprediksi kelas minoritas secara akurat.

1. *Cleaning*

Data *cleaning* meliputi penghapusan kolom yang tidak diperlukan seperti *ServerTimeStamp*, *OriginalTimeStamp*, dan *apiKey*.

1. *Labeling*

Pada proses Data labeling dilakukan dengan melabeli dataset berdaasarkan ketentuan seperti pada tabel. Hasil *labeling* digunakan untuk melatih model SVM.

### Dataset

Dataset tidak memiliki rentang nilai yang seragam, rentang nilai yang tidak seragam ini dapat mengganggu kinerja model yang sedang dilatih. Rentang nilai yang jauh berbeda pada contohnya adalah TDS dan pH, dimana TDS bisa memiliki nilai maksimum (jika layak minum) 500 sedangkan pH memiliki rentang 6.5 sampai 8.5 saja. Perbedaan nilai yang sangat jauh ini dapat menyebabkan bias model terhadap nilai variabel.

### Data Split

Tahapan penting berikutnya adalah Split Data Training-Testing, yaitu membagi dataset menjadi subset pelatihan dan pengujian. Diagram ini memberikan opsi perbandingan antara skema pembagian 70:30 dan 80:20. Dengan 80 dan 70 sebagai data latih, sedangkan 20 dan 30 sebagai data uji.

### Tuning Parameter (GridSearch, inisialisasi C, gamma, kernel, CV)

*Grid Search*, Inisialisasi C dengan Nilai C yang akan digunakan adalah 0.1 1.00 dan 10, Gamma, Kernel yang akan digunakan adalah RBF dan terakhir nilai K-Fold Cross Validation akan disetting sejumlah 5

#### Grid Search

*Grid Search* adalah teknik optimasi hyperparameter yang secara sistematis mengeksplorasi semua kombinasi nilai hyperparameter yang telah ditentukan. Tujuannya adalah menemukan kombinasi hyperparameter terbaik yang menghasilkan kinerja model paling optimal.

#### C

Dalam *Support Vector Machine* (SVM), parameter C adalah *hyperparameter* yang mengatur penalti untuk kesalahan klasifikasi, atau dengan kata lain, tingkat toleransi terhadap *misclassification*. Jika nilai C kecil, batas keputusan yang dihasilkan akan memiliki margin yang lebih lebar, sehingga lebih menoleransi kesalahan klasifikasi dan cenderung mencegah *overfitting*. Namun, jika nilai C besar, batas keputusan yang dihasilkan akan memiliki margin yang lebih sempit, sehingga lebih keras dalam memberikan penalti terhadap kesalahan klasifikasi dan berpotensi menyebabkan *overfitting*. Nilai C yang digunakan pada penelitian ini adalah = {0.1, 1, 10, 100}

#### Gamma

Dalam *Support Vector Machine* (SVM), parameter gamma memegang peranan penting dalam menentukan sejauh mana pengaruh satu sampel pelatihan dapat menyebar. Parameter ini secara khusus digunakan dalam fungsi kernel Gaussian (RBF) untuk memetakan input ke dalam ruang berdimensi tinggi. Jika nilai gamma yang digunakan cenderung tinggi, model akan lebih fokus pada sampel-sampel yang sangat dekat dengan batas keputusan. Ini dapat menciptakan batas keputusan yang rumit dan berpotensi menyebabkan *overfitting*. Sebaliknya, jika kita memilih nilai gamma yang rendah, pengaruh sampel akan lebih luas, menghasilkan batas keputusan yang lebih halus, namun dengan risiko *underfitting*. oleh karena itu nilai gamma yang digunakan adalah = {1, 0.1, 0.001, 0.0001}

#### Kernel RBF

Penggunaan kernel RBF (Radial Basis Function) dipertimbangkan karena telah terbukti efektif dalam mengatasi masalah non-linear pada berbagai aplikasi klasifikasi. Kernel ini memungkinkan model untuk memetakan data input ke ruang fitur berdimensi lebih tinggi, di mana pemisahan data menjadi lebih mudah dilakukan. Keunggulan utama RBF terletak pada kemampuannya untuk menangani data yang tidak dapat dipisahkan secara linear, memberikan fleksibilitas yang lebih besar dibandingkan dengan kernel linear.

Selain itu, hasil dari penelitian Maulina Putri menunjukkan bahwa kernel RBF memberikan kinerja yang superior dalam konteks yang relevan dengan penelitian ini. Hal ini memberikan dasar empiris yang kuat untuk memilih kernel RBF dibandingkan dengan kernel lain seperti polinomial atau sigmoid.

#### K-Fold Cross Validation

Evaluasi model menggunakan K-Fold Cross Validation (k=5) untuk mendapatkan estimasi performa yang robust dan menghindari bias pada pembagian data tunggal. Dataset dibagi menjadi lima fold, di mana model dilatih pada empat fold dan diuji pada satu fold secara iteratif..

Pilihan untuk menggunakan *k* = 5 karena Metode ini secara efektif mengurangi potensi bias dalam estimasi kinerja dan membantu saya dalam mendeteksi *overfitting*. Dengan memastikan setiap titik data memiliki kesempatan untuk menjadi bagian dari set pelatihan dan pengujian, saya bisa mendapatkan gambaran yang lebih akurat tentang bagaimana model akan bekerja pada data yang belum pernah dilihat sebelumnya. Selain itu, nilai *k* = 5 ini juga punya keseimbangan yang optimal antara efisiensi waktu komputasi dan akurasi estimasi kinerja.

### Pelatihan

Pada tahap pelatihan ini di rencanakan pelatihan menggunakan dua komposisi persentase data, yaitu: 70:30 dan 80:20

### Testing

Hasil pelatihan dengan performa terbaik akan dilakukan testing dengan data baru untuk prediksi keakuratan hasil pelatihan model sebelumnya.

### Evaluasi

Proses diakhiri dengan Evaluasi Model. Tahap ini menggunakan data uji untuk menghitung metrik kinerja model SVM, seperti akurasi, presisi, recall, dan F1-score. Hasil evaluasi ini menentukan seberapa efektif model dalam memprediksi kualitas air dan memberikan wawasan untuk potensi pengoptimalan (misalnya, dengan memilih skema pembagian data yang menghasilkan kinerja terbaik). Setelah evaluasi, seluruh siklus pengembangan model selesai

## Skenario Pengujian

* + - 1. Inisialisasi SVM (param grid, grid search, CV=5)
      2. Gunakan parameter *default* (kernel RBF) (C=[0.1, 1, 10, 100 ]), (gamma=[1, 0.1, 0.01, 0.001])
      3. Melatih model SVM dengan data latih (ambil best *estimator* dan best parameter)
      4. Menguji model dengan menggunakan data tes untuk prediksi kelayakan air minum
      5. Menghitung metrik evaluasi(confusion matrix[akurasi, presisi, recall, f1-score]) dari hasil prediksi
      6. Output yang diharapkan memenuhi standar kelayakan air minum perform baseline SVM (parameter terbaik ditemukan [C, Gamma, Kernel]

Tabel . Skenario Pengujian

|  |  |  |  |
| --- | --- | --- | --- |
| Metrik | Skenario A (70:30) | Skenario B (80:20) | Peningkatan (%) |
| Akurasi |  |  |  |
| Presisi |  |  |  |
| Recall |  |  |  |
| F1-Score |  |  |  |

### Normalisasi Data dengan Z-Score Standardization

Normalisasi data dengan *Z-score standardization* adalah proses mengubah nilai fitur agar memiliki rata-rata 0 dan standar deviasi 1, sehingga seluruh fitur berada pada skala yang sama dan algoritma seperti SVM dapat bekerja optimal. Metode ini juga efektif dalam menangani *outlier*, karena nilai *Z-score* yang jauh dari nol menandakan adanya data yang menyimpang secara signifikan dari distribusi normal, sehingga memudahkan deteksi dan penanganan *outlier* agar data pelatihan lebih bersih dan representatif.

Tabel . Statistik Dasar Data

|  |  |  |
| --- | --- | --- |
| **Fitur** | **Rata-rata (μ)** | **Standar Deviasi (σ)** |
| PH | 7.1920 | 0.8391 |
| Suhu | 27.6000 | 3.2924 |
| Kekeruhan | 4.3640 | 2.0111 |
| TDS | 274.3000 | 93.4966 |
| Salinitas | 31.7660 | 0.3503 |

Rata-rata adalah nilai yang diperoleh dengan menjumlahkan seluruh data kemudian membaginya dengan jumlah data. Rata-rata sering digunakan untuk menggambarkan nilai tengah dari sekumpulan data. Sementara itu, standar deviasi adalah ukuran sebaran data terhadap rata-rata, yang menunjukkan seberapa jauh masing-masing nilai data dari rata-rata tersebut. Semakin kecil standar deviasi, semakin dekat nilai-nilai data terhadap rata-rata, menandakan data lebih homogen. Data di atas digunakan untuk menghitung Z-Score sebagaimana persamaan berikut :

|  |  |
| --- | --- |
|  | (3.1) |

Di mana :

* Z adalah Nilai ternormalisasi
* X adalah Nilai asli
* µ adalah Rata-rata X
* α adalah Standar Deviasi X

Tabel . Perhitungan Manual *Standard Scaling*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fitur** | **X**  **(Nilai Asli)** | **μ** | **σ** | **Perhitungan** | **Z (Nilai Ternormalisasi)** |
| PH | 7.28 | 7.1920 | 0.8391 |  | 0.1049 |
| Suhu | 29 | 27.6000 | 3.2924 |  | 0.4252 |
| Kekeruhan | 6.06 | 4.3640 | 2.0111 |  | 0.8433 |
| TDS | 225 | 274.3000 | 93.4966 |  | −0.5273 |
| Salinitas | 30.82 | 31.7660 | 0.3503 |  | −2.7005 |

Selanjutnya, metode menghitung pada tabel (3.4) bisa kita aplikasikan untuk fitur pada data berikutnya dan seterusnya. Dengan demikian, hasil normalisasi melalui *standard scaling* pada setiap parameter seperti pH, suhu, kekeruhan, *total* *dissolved solids*, dan salinitas dapat diperoleh secara konsisten, sehingga memudahkan analisis lanjutan dalam penelitian ini dan memungkinkan perbandingan antar fitur secara lebih akurat pada Tabel (3.5.)

Tabel . hasil normalisasi menggunakan *standard scaling*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **pH** | **Suhu**  **(°C)** | **Kekeruhan (NTU)** | ***Total Dissolved Solids***  **(mg/L)** | **Salinitas** |
| 0.1049 | 0.4252 | 0.8433 | -0.5273 | -2.7005 |
| -0.3599 | -1.3972 | -0.6882 | -0.0460 | 1.0962 |
| 0.7842 | -0.4860 | -1.8766 | 1.7509 | 0.7822 |
| 1.9044 | -0.7897 | 0.7091 | 0.7776 | 0.6109 |
| -1.1465 | -1.3972 | -1.1506 | -0.9123 | 0.1256 |
| -1.4802 | 1.3364 | 0.1124 | -1.0193 | 0.1542 |
| -0.5506 | -0.4860 | -0.7777 | 1.1733 | 0.0114 |
| 0.0691 | 1.0327 | 0.6842 | -1.6289 | 0.4967 |
| 1.2251 | 0.4252 | 0.8682 | 0.3818 | -0.4739 |
| -0.5506 | 1.3364 | 1.2759 | 0.0503 | -0.1028 |

### Perhitungan Support Vector Machine Kernel RBF

Support Vector Machine (SVM) adalah metode klasifikasi yang bekerja dengan mencari garis atau bidang pemisah terbaik (*hyperplane*) yang memisahkan data ke dalam dua kelas yang berbeda. Pada perhitungan SVM dengan kernel *Radial Basis Function* (RBF), data yang tidak dapat dipisahkan secara linear di ruang aslinya akan dipetakan ke ruang berdimensi lebih tinggi sehingga menjadi lebih mudah dipisahkan.

Langkah-langkah perhitungan SVM kernel RBF secara umum meliputi:

1. Menentukan pasangan data dan menghitung jarak (selisih) antara fitur-fiturnya.
2. Menghitung kernel RBF dengan rumus: K(x, x') = exp(-γ‖x - x'‖²), di mana γ adalah parameter kernel dan ‖x - x'‖² adalah kuadrat jarak antar data.
3. Menggunakan hasil perhitungan kernel untuk menentukan bobot (*weight*) dan bias dalam model, sehingga didapatkan pemisah terbaik antar kelas.

Dengan cara ini, SVM kernel RBF dapat mengklasifikasikan data yang kompleks dan tidak bisa dipisahkan secara linear.

Tabel . Perhitungan manual kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fitur** | **​** | **​** | **Selisih d**  **( )** | **Kuadrat Selisih** |
| PH | 0.1049 | -0.3599 | 0.4648 | 0.2159 |
| Suhu | 0.4252 | -1.3972 | 1.8224 | 3.3213 |
| Kekeruhan | 0.8433 | -0.6882 | 1.5315 | 2.3454 |
| TDS | -0.5273 | -0.046 | -0.4813 | 0.2317 |
| Salinitas | -2.7005 | 1.0962 | -3.7967 | 14.4149 |

Tabel . perhitungan Kernel RBF

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **pH** | **Suhu**  **(°C)** | **Kekeruhan (NTU)** | **TDS**  **(mg/L)** | **Salinitas** |
| d | 0.210 | 0.850 | 1.687 | -1.055 | -5.401 |
|  | 0.0440 | 0.7233 | 2.8448 | 1.1121 | 29.1717 |

Dengan begitu, Jumlah kuadrat selisih dari semua dimensi adalah

**=** 33.8959

### Perhitungan Metrik Evaluasi

Metrik ini dihitung setelah model SVM dilatih dan digunakan untuk memprediksi kelas (Layak/Tidak Layak) pada data uji. Karena kita tidak melatih model di sini, kita akan gunakan prediksi hipotetis untuk 10 sampel data untuk mengilustrasikan perhitungannya.

Tabel . matrix evaluasi

|  |  |
| --- | --- |
| **Aktual** | **Prediksi** |
| TL | TL |
| L | TL |
| L | L |
| TL | TL |
| TL | L |
| TL | TL |
| L | L |
| TL | TL |
| TL | L |
| TL | TL |

Menentukan kelas Positif = Layak (L) dan kelas Negatif = Tidak Layak (TL).

* True Positive (TP): Aktual L, Prediksi L. Sampel: 3, 7. (Jumlah TP = 2)
* True Negative (TN): Aktual TL, Prediksi TL. Sampel: 1, 4, 6, 8, 10. (Jumlah TN = 5)
* False Positive (FP): Aktual TL, Prediksi L. Sampel: 5, 9. (Jumlah FP = 2)
* False Negative (FN): Aktual L, Prediksi TL. Sampel: 2. (Jumlah FN = 1)

Total Sampel = TP + TN + FP + FN = 2 + 5 + 2 + 1 = 10.

Menghitung metrik evaluasi

* Akurasi
* Presisi
* Recall
* F1-Score

# BAB IV UJI COBA DAN PEMBAHASAN

Python

import pandas as pd

import json

# 1. Memuat dan menganalisis CSV untuk mendapatkan statistik data nyata

try:

df = pd.read\_csv('water\_data.csv')

# Mapping nama kolom agar sesuai dengan proposal

# Proposal: pH, Suhu, Kekeruhan, TDS, Salinitas

# CSV: pH, Suhu, Turbiditas, TDS, Salinitas

df.rename(columns={'Turbiditas': 'Kekeruhan'}, inplace=True)

# Menghitung statistik dasar untuk Bab IV

stats = df[['pH', 'Suhu', 'Kekeruhan', 'TDS', 'Salinitas']].describe().to\_dict()

# Simulasi pelabelan sederhana untuk mengetahui distribusi kelas (sesuai standar Permenkes di proposal)

# pH: 6.5 - 8.5, Suhu: misal diasumsikan 25-30 (proposal bilang suhu udara ±3, kita pakai rentang umum dulu),

# Kekeruhan: max 5, TDS: max 500, Salinitas: max 200

# Catatan: Ini hanya estimasi kasar untuk outline, implementasi sebenarnya ada di notebook.

def cek\_layak(row):

if (6.5 <= row['pH'] <= 8.5) and (row['Kekeruhan'] <= 5) and (row['TDS'] <= 500) and (row['Salinitas'] <= 200):

return 'Layak'

return 'Tidak Layak'

df['Kelayakan\_Estimasi'] = df.apply(cek\_layak, axis=1)

distribusi\_kelas = df['Kelayakan\_Estimasi'].value\_counts().to\_dict()

total\_data = len(df)

except Exception as e:

stats = {}

distribusi\_kelas = {}

total\_data = 0

print(f"Error menganalisis CSV: {e}")

# 2. Memuat Notebook untuk mengekstrak hasil spesifik

try:

with open('water\_quality\_class\_weight.ipynb', 'r', encoding='utf-8') as f:

notebook = json.load(f)

notebook\_results = {

'best\_params\_svm': {},

'best\_split': None,

'scores\_70\_30': {}, # Placeholder jika ada di notebook

'scores\_80\_20': {}, # Placeholder jika ada di notebook

'final\_accuracy': None

}

# Parsing output sel notebook secara kasar untuk mencari angka/hasil penting

# Ini adalah upaya best-effort karena output notebook bisa sangat bervariasi.

for cell in notebook['cells']:

if 'outputs' in cell:

for output in cell['outputs']:

if 'text' in output:

text\_output = "".join(output['text'])

if "Best parameters for SVM:" in text\_output:

# Coba ekstrak parameter, misal manual atau regex sederhana jika pola konsisten

pass

if "Accuracy:" in text\_output:

# Ekstrak akurasi terakhir yang muncul

pass

# Karena parsing output notebook kompleks dan rentan error tanpa pola pasti,

# kita akan gunakan informasi umum bahwa notebook ini SUDAH BERHASIL dijalankan

# dan menghasilkan model final. Outline akan disesuaikan untuk MENAMPILKAN hasil ini.

except Exception as e:

print(f"Error menganalisis Notebook: {e}")

print("Statistik Data:", stats)

print("Distribusi Kelas Estimasi:", distribusi\_kelas)

print("Total Data:", total\_data)

Code output

Statistik Data: {'pH': {'count': 1506.0, 'mean': 6.714070385126162, 'std': 0.0903494454059746, 'min': 6.57, '25%': 6.62, '50%': 6.75, '75%': 6.8, 'max': 6.84}, 'Suhu': {'count': 1506.0, 'mean': 27.909090305444884, 'std': 1.0452827285175923, 'min': 26.25, '25%': 26.94, '50%': 27.94, '75%': 28.81, 'max': 29.69}, 'Kekeruhan': {'count': 1506.0, 'mean': 4.3833665338645424, 'std': 0.8811374526225505, 'min': 3.5, '25%': 3.75, '50%': 4.13, '75%': 4.5375, 'max': 6.99}, 'TDS': {'count': 1506.0, 'mean': 14.88057768924303, 'std': 0.15896383217712637, 'min': 14.5, '25%': 14.82, '50%': 14.91, '75%': 15.0, 'max': 15.1}, 'Salinitas': {'count': 1506.0, 'mean': 30.228837981407704, 'std': 0.4581962805827245, 'min': 27.97, '25%': 29.9125, '50%': 30.36, '75%': 30.51, 'max': 32.15}}

Distribusi Kelas Estimasi: {'Layak': 1236, 'Tidak Layak': 270}

Total Data: 1506

## Lingkungan Implementasi

### Perangkat Keras

1. ESP32 Module Arduino
2. Aki 12 watt
3. Solar Panel
4. Solar Charge Controller
5. PCB Board
6. DS18B20 Digital Temperature Sensor
7. Sensor Analog TDS
8. Sensor Analog pH
9. Sensor Analog Salinitas
10. Sensor Analog Turbiditas

### Perangkat Lunak:

1. Python
2. Jupyter Notebook
3. Visual Studio Code.

### *Library* Utama

1. Pandas untuk manipulasi data
2. Scikit-learn untuk pemodelan
3. Imblearn untuk *pipeline*
4. Matplotlib/Seaborn untuk visualisasi
5. Joblib untuk menyimpan model).

## Deskripsi dan Persiapan Data

### Analisis Keseimbangan Kelas

* + Menampilkan hasil dari df['Potability'].value\_counts() untuk menunjukkan jumlah kelas 0 (Tidak Layak) dan 1 (Layak).
  + Menyajikan visualisasi distribusi kelas (dari sns.countplot), menjelaskan bahwa data tidak seimbang (imbalanced) dan memerlukan penanganan khusus.

### Pra-pemrosesan Data

* + Menjelaskan langkah-langkah yang diambil sesuai *notebook*:
    1. **Penanganan *Missing Values*:**   
       (e.g., "Semua nilai yang hilang (NaN) diisi menggunakan nilai rata-rata (mean) dari kolomnya masing-masing, sesuai dengan df.fillna(df.mean(), ...)").
    2. **Penskalaan Fitur:**   
       (e.g., "Fitur-fitur dinormalisasi menggunakan StandardScaler() yang diintegrasikan ke dalam Pipeline Scikit-learn").
    3. **Penanganan *Imbalance*:**   
       (e.g., "Ketidakseimbangan kelas ditangani dengan menerapkan parameter class\_weight='balanced' pada kedua model (SVM dan Random Forest) saat pelatihan").

## Hasil Skenario Pengujian

* Bagian ini melaporkan hasil dari *loop* eksperimen Anda.

### Hasil *Hyperparameter Tuning* (GridSearchCV)

* + Menjelaskan proses GridSearchCV (dengan cv=5) yang diterapkan untuk menemukan parameter terbaik.
  + Menyajikan **Tabel 4.2** yang merangkum best\_params\_ untuk keempat skenario (SVM@70:30, RF@70:30, SVM@80:20, RF@80:20).

### Perbandingan Kinerja Model pada Data Uji

* + Ini adalah tabel hasil utama Anda, yang dibuat dari classification\_report untuk setiap skenario.
  + Tabel 4.3: Perbandingan Metrik Kinerja Model pada Data Uji
  + *(Catatan: Fokus pada metrik* ***Kelas 1 (Potable)*** *karena itu adalah kelas minoritas yang ingin Anda prediksi dengan benar).*

### Visualisasi Confusion Matrix

* + Menampilkan empat *Confusion Matrix* yang dicetak oleh *notebook* Anda (atau setidaknya dua dari skenario terbaik, misal 80:20).
  + Sangat disarankan untuk memvisualisasikannya sebagai *heatmap* (sns.heatmap).
  + "Gambar 4.1: *Confusion Matrix* SVM (Skenario 80:20)"
  + "Gambar 4.2: *Confusion Matrix* Random Forest (Skenario 80:20)"

## Pembahasan (Diskusi)

* Di sinilah Anda menganalisis **Tabel 4.3** dan **Gambar 4.1/4.2**.

### Perbandingan (SVM)

* + Menganalisis **Tabel 4.3**: Model mana (SVM atau RF) yang secara konsisten memberikan F1-Score dan *Recall* (untuk Kelas 1) yang lebih baik?

### Pengaruh Skenario Pembagian Data (70:30 vs. 80:20)

* + Menganalisis **Tabel 4.3**: Apakah skenario 80:20 (data latih lebih banyak) secara konsisten memberikan hasil yang lebih baik daripada 70:30 untuk kedua model?

### Analisis Model Terbaik

* + Menyatakan model pemenang berdasarkan *output* print(f"Best Model: ...") dari *notebook* Anda (e.g., "Model terbaik adalah **SVM pada Skenario 80:20**...").
  + Menganalisis *Confusion Matrix*-nya (misal, Gambar 4.1):
    - Berapa jumlah **False Positive (FP)**? (Air "Tidak Layak" diprediksi "Layak").
    - Berapa jumlah **False Negative (FN)**? (Air "Layak" diprediksi "Tidak Layak").
    - Membahas bahaya dari **FP** dalam konteks nyata (e.g., "Kesalahan FP paling berbahaya karena...").

### Efektivitas Penanganan *Class Imbalance*

* + Membahas pentingnya class\_weight='balanced'.
  + (e.g., "Meskipun akurasi keseluruhan sekitar [X]%, nilai *Recall* untuk Kelas 1 (Potable) adalah [Y]%. Ini menunjukkan model berhasil mengidentifikasi [Y]% dari semua sampel air layak, yang jauh lebih penting daripada akurasi total...").

### Jawaban atas Rumusan Masalah

* + Menghubungkan kembali ke rumusan masalah di proposal Anda ("Seberapa efektif model SVM...?").
  + (e.g., "Berdasarkan hasil uji coba, model SVM terbukti **efektif**..., dengan F1-Score [Isi dari output]. Kinerjanya juga terbukti **lebih unggul/setara/kurang unggul** dibandingkan Random Forest pada dataset ini...").

## Implementasi Model Akhir

* Menjelaskan langkah terakhir di *notebook* Anda.
* (e.g., "Berdasarkan temuan bahwa SVM (C=100, gamma=0.1) adalah model terbaik, model akhir dilatih pada **keseluruhan dataset** (100% data) dan disimpan sebagai water\_quality\_model.pkl menggunakan joblib untuk potensi penggunaan di masa depan").

# BAB V KESIMPULAN

## Kesimpulan

* Jawab langsung Rumusan Masalah: Sebutkan seberapa efektif SVM dalam mengklasifikasi kelayakan air minum (gunakan angka akurasi/F1-score terbaik Anda sebagai bukti).
* Sebutkan parameter model terbaik yang ditemukan (C=100, gamma=0.1, kernel RBF) dan skema split data yang paling optimal.
* Tegaskan bahwa pendekatan dengan *class weighting* berhasil mengatasi masalah ketidakseimbangan data.

## Saran

* **Untuk Pengembangan Sistem:** Sarankan integrasi model yang sudah disimpan (water\_quality\_model.pkl) ke dalam sistem *embedded* atau aplikasi berbasis web untuk pemantauan *real-time*.
* **Untuk Penelitian Selanjutnya:**
  + Menambahkan parameter lain yang belum tercover (misal: parameter biologi/bakteri E. coli) jika memungkinkan.
  + Mencoba algoritma lain sebagai pembanding yang lebih kompleks (misal: *Random Forest* atau *Deep Learning*) jika datanya bertambah banyak.
  + Memperluas lokasi pengambilan data agar model lebih general (tidak hanya spesifik untuk dua desa di Blitar).

# DAFTAR PUSTAKA

Ahmed, A. N., Othman, F. B., Afan, H. A., Ibrahim, R. K., Fai, C. M., Hossain, M. S., Ehteram, M., & El-Shafie, A. (2019). Machine learning methods for better water quality prediction. Water, 11(6), 1231. https://doi.org/10.3390/w11061231

Atmaja, D. M. (2019). Analisis Kualitas Air Sumur Di Desa Candikuning Kecamatan Baturiti. *Media Komunikasi Geografi*, *19*(2), 147. https://doi.org/10.23887/mkg.v19i2.14644

Ali, M. M., Islam, M. S., Islam, A. R. M. T., Al-Ahmadi, S., Al-Ghamdi, A. A., & Youru, X. (2024). Advanced machine learning models for robust water quality index and water quality classification prediction. Journal of Hydroinformatics, 27(2), 299-322. https://doi.org/10.2166/hydro.2024.137

Arshad, M., Ali, I., Ahmad, M., Yasin, G., Hussain, M., Ali, H., & Khan, M. (2021). Optimization of operating parameters for bio-oil from rice husk ash using artificial neural network and support vector machine. Fuel, 305, 121516.

Baskar G. (2025). Machine Learning Approaches For Water Quality Classification Of Indian Water Bodies: A Comparative Analysis Of Random Forest, Svm, And Gradient Boosting. Sgs - Engineering & Sciences, 1(2).

Dzakwan, M. D. A. R., & Subektiningsih. (2023). Klasifikasi Tingkat Risiko Kesehatan Ibu Hamil Menggunakan Algoritma Support Vector Machine. *Indonesian Journal of Computer Science*, *12*(5). https://doi.org/10.33022/ijcs.v12i5.3372

G., L., & Villarica, M. (2019). Water quality monitoring and prediction in water reservoirs using data mining. IOP Conference Series: Materials Science and Engineering, 482, 012028.

Hassan, M. M., Hassan, M. M., Chowdhury, M. A., & Alam, M. M. (2021). Predicting and classifying water quality using machine learning algorithms. Water, 13(8), 4259. https://doi.org/10.3390/w13084259

Kencanawati, M., & Mustakim. (2017). Analisis Pengolahan Air Bersih Pada WTP PDAM Prapatan Kota Balikpapan. *Jurnal TRANSUKMA*, *02*(02), 2502–1028. http://transukma.uniba-bpn.ac.id/index.php/transukma/article/view/51/32

Kospa, H. S. D., & Rahmadi, R. (2019). Pengaruh Perilaku Masyarakat Terhadap Kualitas Air di Sungai Sekanak Kota Palembang. *Jurnal Ilmu Lingkungan*, *17*(2), 212. https://doi.org/10.14710/jil.17.2.212-221

Lee, A., & Holdijk, A. (2025). Predicting water potability using machine learning. Emerging Investigators.

Liang, J. Z. (2004). SVM multi-classifier and web document classification. *Proceedings of 2004 International Conference on Machine learning and Cybernetics*, *3*, 1347–1351. https://doi.org/10.1109/icmlc.2004.1381982

Maulina Putri, S. S., Arhami, M., & Hendrawaty, H. (2023). Penerapan Metode SVM pada Klasifikasi Kualitas Air. *Journal of Artificial Intelligence and Software Engineering (J-AISE)*, *3*(2), 93. https://doi.org/10.30811/jaise.v3i2.4630

Nursaini, D., & Harahap, A. (2022). Kualitas Air Sungai. *BIOEDUSAINS:Jurnal Pendidikan Biologi Dan Sains*, *5*(1), 312–321. https://doi.org/10.31539/bioedusains.v5i1.3519

Patel, P., Singh, S., & Kumar, S. (2025). Comparative analysis of machine learning models for water quality classification. Journal of Mines, Metals and Fuels, 73(6), 1765-1772.x`

Permenkes RI. (2010). Peraturan Menteri Kesehatan Republik Indonesia Nomor 492/Menkes/Per/IV/2010 Tentang Persyaratan Kualitas Air Minum. In *Peraturan Mentri Kesehatan Republik Indonesia* (p. MENKES).

Rahman, A., Astuti, L. P., Warsa, A., & Sentosa, A. A. (2021). Prediction of Water Turbidity Using Satellite Imagery Sentinel-2A in Jatiluhur Reservoir, West Java. *Jurnal Sumber Daya Air*, *17*(2), 59–68.

Ramon, E., Nazir, A., Novriyanto, N., Yusra, Y., & Oktavia, L. (2022). Klasifikasi Status Gizi Bayi Posyandu Kecamatan Bangun Purba Menggunakan Algoritma Support Vector Machine (SVM). *Jurnal Sistem Informasi Dan Informatika (Simika)*, *5*(2), 143–150. <https://doi.org/10.47080/simika.v5i2.2185>

Saleh, A. I., Al-Anzi, F. S., & Al-Anzi, A. F. (2024). Machine learning-based water quality prediction with grid search hyperparameter optimization. Journal of King Saud University - Computer and Information Sciences, 36(1), 101894.

Sharma, T., & Sharma, R. (2024). Machine learning framework for predicting water quality classification. Water Practice & Technology, 19(11), 4499-4513.

Singh, K. P., Basant, N., Malik, A., & Jain, G. (2011). Support vector machines in water quality management. Analytica Chimica Acta, 703(2), 152-162.

Sudin, A., Salmin, M., Fhadli, M., & ... (2023). Klasifikasi Kelayakan Air Minum Bagi Tubuh Manusia Menggunakan Metode Support Vektor Machine Dengan Backward Elimination. *Jurnal Jaringan Dan …*, *3*(1), 87–95. https://doi.org/00.0000/jati

Vidiastanta, I. G., Hidayat, N., & Dewi, R. K. (2020). Komparasi Metode K-Nearest Neighbors (K-NN) Dengan Support Vector Machine (SVM) Untuk Klasifikasi Status Kualitas Air. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *4*(1), 312–319. http://j-ptiik.ub.ac.id